Control for systems with explicit constraints

Control Theory, Lecture 13

by Sergei Savin

Spring 2021

Content

- Explicit and no constraints
- Explicit and implicit constraints
- Examples of systems with constraints
- Typical reasons why explicit constraints arise
- Ways to control systems with explicit constraints

Explicit and no constraints

LTI systems we studied before have the form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state of the system. This form has no explicit constraints.

Let us introduce one of the forms of a linear dynamical system with explicit constraints:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{F}\lambda \\ \mathbf{G}\dot{\mathbf{x}} = \mathbf{0} \end{cases}$$

where $\mathbf{G}$ is a constraint matrix, $\lambda \in \mathbb{R}^k$ is the constraints reaction forces, and $\mathbf{F}$ is the reaction force Jacobian.

Consider a two mass system with a spring:

$$\begin{cases} \ddot{x}_1 + \mu\dot{x}_1 + k(x_1 - x_2) = 0 \\ \ddot{x}_2 + \mu\dot{x}_2 + k(x_2 - x_1) = 0 \end{cases}$$

We can add a constraint $x_2 = 10$. This implies that $\ddot{x}_2 = 0$. Corresponding system of equations is:

$$\begin{cases} \ddot{x}_1 + \mu\dot{x}_1 + k(x_1 - x_2) = 0 \\ \ddot{x}_2 + \mu\dot{x}_2 + k(x_2 - x_1) = \lambda \\ \ddot{x}_2 = 0 \end{cases}$$

But that is the same as:

$$\ddot{x}_1 + \mu\dot{x}_1 + k(x_1 - 10) = 0$$

Thus we transformed the system with *explicit constraints* into a system with *implicit constraints*

Figure 1: Walking robots

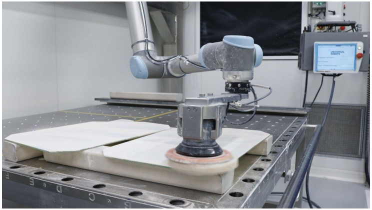

Figure 2: Polishing with industrial arms

Typical reasons why explicit constraints arise

Explicit constraints are usually not a necessity and not a physical property of the problem. However, they are often encountered in practice. Typical situations when they are encountered as as follows:

- Systems with contact interactions.
- Hybrid systems (two or more different dynamics which switch between one-another).
- Nonholonomic constraints in the dynamics (dynamics of a unicycle, bicycle, etc.).
- Dynamics is more clear and easy to work which when non-minimal representation is used.

Ways to control systems with explicit constraints

There are basic ways to deal with such systems:

- Reduce to a system with implicit constraints and control that system instead.
- Treat reaction forces as a yet another external force.
- Design control law based on the explicit representation of constraints.

Consider equations in the form:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{c} \tag{1}$$

where $\mathbf{A}$ is the state matrix, $\mathbf{B}$ is the control matrix and $\mathbf{c}$ is the affine term of the affine dynamics model.

For systems with constraints the same linearization takes form:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{S}\lambda + \mathbf{c} \\ \mathbf{G}\dot{\mathbf{x}} = 0 \end{cases} \tag{2}$$

where $\mathbf{S}$ is linearized constraint Jacobian and $\mathbf{G} = \begin{bmatrix} \mathbf{F} & \mathbf{0} \\ \dot{\mathbf{F}} & \mathbf{F} \end{bmatrix}$.

We can observe that constraint $\mathbf{G}\dot{\mathbf{x}} = 0$ implies that all feasible state velocities $\dot{\mathbf{x}}$ lie in the null space of $\mathbf{G}$. This means that we can introduce a new lower dimensional variable $\mathbf{z}$ to describe $\mathbf{x}$ (assuming initial value of $\mathbf{x}$ lies in the column space of $\mathbf{N}$):

$$\mathbf{N}\mathbf{z} = \mathbf{x} \tag{3}$$

where $\mathbf{N} = \text{null}(\mathbf{G})$ - orthonormal basis in the null space of $\mathbf{G}$.

Let us re-express dynamics (2) in terms of $\mathbf{z}$ by multiplying it by $\mathbf{N}^\top$ on the left:

$$\mathbf{N}^\top \dot{\mathbf{x}} = \mathbf{N}^\top \mathbf{A}\mathbf{x} + \mathbf{N}^\top \mathbf{B}\mathbf{u} + \mathbf{N}^\top \mathbf{S}\lambda + \mathbf{N}^\top \mathbf{c} \tag{4}$$

We can prove that $\mathbf{N}^\top \mathbf{S} = 0$ for all mechanical systems (for example, by observing that mechanical constrains do not do work) or check that our particular $\mathbf{S}$ lies in the row space of our $\mathbf{G}$.

Noting that $\dot{\mathbf{z}} = \mathbf{N}^\top \dot{\mathbf{x}}$ and $\mathbf{x} = \mathbf{N}\mathbf{z}$ we get:

$$\dot{\mathbf{z}} = \mathbf{N}^\top \mathbf{A}\mathbf{N}\mathbf{z} + \mathbf{N}^\top \mathbf{B}\mathbf{u} + \mathbf{N}^\top \mathbf{c} \tag{5}$$

Defining $\mathbf{A}_N = \mathbf{N}^\top \mathbf{A}\mathbf{N}$, $\mathbf{B}_N = \mathbf{N}^\top \mathbf{B}$ and $\mathbf{c}_N = \mathbf{N}^\top \mathbf{c}$ we get:

$$\dot{\mathbf{z}} = \mathbf{A}_N \mathbf{z} + \mathbf{B}_N \mathbf{u} + \mathbf{c}_N \tag{6}$$

Since we achieved that our constrained dynamics is written in the standard LTI form:

$$\dot{\mathbf{z}} = \mathbf{A}_N \mathbf{z} + \mathbf{B}_N \mathbf{u} + \mathbf{c}_N, \tag{7}$$

we can use standard LTI control methods on it, for example finding optimal feedback gains via pole placement or LQR:

$$\mathbf{K}_N = \mathrm{lqr}(\mathbf{A}_N, \mathbf{B}_N, \mathbf{Q}, \mathbf{R}) \tag{8}$$

where $\mathbf{Q}$ and $\mathbf{R}$ are matrices defining cost function for the LQR problem.

For any LTI system, including the LTI form of a constrained system we saw previously, inverse dynamics can be solved precisely by a pseudo-inverse, as long as there exist a solution. The following condition verifies it:

$$(\mathbf{I} - \mathbf{B}\mathbf{B}^{+})(\dot{\mathbf{x}} - \mathbf{A}\mathbf{x} - \mathbf{c}) = 0, \tag{9}$$

The condition checks if vector $(\dot{\mathbf{x}} - \mathbf{A}\mathbf{x} - \mathbf{c})$ lies in the column space of $\mathbf{B}$. If it holds, precise solution to inverse kinematics can be found as:

$$\mathbf{u}_{ID} = \mathbf{B}^{+}(\dot{\mathbf{x}} - \mathbf{A}\mathbf{x} - \mathbf{c}). \tag{10}$$

For a constrained mechanical system we can solve inverse dynamics without the need for linearization. Consider the following dynamics:

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{g} = \mathbf{T}\mathbf{u} + \mathbf{F}^\top \lambda \tag{11}$$

We can represent constraint Jacobian $\mathbf{F}^\top$ as its QR decomposition: $\mathbf{F}^\top = \mathbf{Q}\begin{bmatrix}\mathbf{R} \\ \mathbf{0}\end{bmatrix}$, where $\mathbf{Q}^\top\mathbf{Q} = \mathbf{Q}\mathbf{Q}^\top = \mathbf{I}$ and $\mathbf{R}$ is convertible.

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{g} = \mathbf{T}\mathbf{u} + \mathbf{Q}\begin{bmatrix}\mathbf{R} \\ \mathbf{0}\end{bmatrix}\lambda \tag{12}$$

Let us multiply the equation by $\mathbf{Q}^\top$:

$$\mathbf{Q}^\top(\mathbf{H}\ddot{\mathbf{q}} + \mathbf{Q}^\top\mathbf{C}\dot{\mathbf{q}} + \mathbf{g}) = \mathbf{Q}^\top\mathbf{T}\mathbf{u} + \begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}\lambda \tag{13}$$

Introducing switching variables (to divide upper and lower part of the equations) $\mathbf{S}_1 = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}$ and $\mathbf{S}_2 = \begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}$ and multiplying equations by one and the other we get two systems:

$$\begin{cases}\mathbf{S}_1\mathbf{Q}^\top(\mathbf{H}\ddot{\mathbf{q}} + \mathbf{Q}^\top\mathbf{C}\dot{\mathbf{q}} + \mathbf{g}) = \mathbf{S}_1\mathbf{Q}^\top\mathbf{T}\mathbf{u} + \mathbf{R}\lambda\\ \mathbf{S}_2\mathbf{Q}^\top(\mathbf{H}\ddot{\mathbf{q}} + \mathbf{Q}^\top\mathbf{C}\dot{\mathbf{q}} + \mathbf{g}) = \mathbf{S}_2\mathbf{Q}^\top\mathbf{T}\mathbf{u}\end{cases} \tag{14}$$

The main advantage we achieved is that now we can calculate both $\mathbf{u}$ and λ

Resulting expression for $\mathbf{u}$ is:

$$\mathbf{u} = (\mathbf{S}_2\mathbf{Q}^\top\mathbf{T})^+\mathbf{S}_2\mathbf{Q}^\top(\mathbf{H}\ddot{\mathbf{q}} + \mathbf{Q}^\top\mathbf{C}\dot{\mathbf{q}} + \mathbf{g}) \tag{15}$$

Expression for λ is:

$$\lambda = \mathbf{R}^{-1}\mathbf{S}_1\mathbf{Q}^\top(\mathbf{H}\ddot{\mathbf{q}} + \mathbf{Q}^\top\mathbf{C}\dot{\mathbf{q}} + \mathbf{g} - \mathbf{T}\mathbf{u}) \tag{16}$$

We can notice a pseudo-inverse, implying that the no-residual solution does not have to exist.

We can easily write inverse dynamics as a QP:

$$\begin{aligned} \underset{\mathbf{u},\lambda}{\text{minimize}} \quad & ||\mathbf{u}||, \\ \text{subject to} \quad & \begin{cases} \mathbf{H}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{g} = \mathbf{T}\mathbf{u} + \mathbf{F}^\top \lambda \\ \mathbf{F}\ddot{\mathbf{q}} + \dot{\mathbf{F}}\dot{\mathbf{q}} = 0 \end{cases} \end{aligned} \tag{17}$$

If there are some constraints or limits on the control input (torque limits, for instance) or the reaction forces are restricted (by friction cones, for instance), those can be directly added.

- Mason, S., Righetti, L. and Schaal, S., 2014, November. Full dynamics LQR control of a humanoid robot: An experimental study on balancing and squatting. In 2014 IEEE-RAS International Conference on Humanoid Robots (pp. 374-379). IEEE.
- Mason, S., Rotella, N., Schaal, S. and Righetti, L., 2016, November. Balancing and walking using full dynamics LQR control with contact constraints. In 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids) (pp. 63-68). IEEE. - arxiv.org/pdf/1701.08179
- Mistry, M., Buchli, J. and Schaal, S., 2010, May. Inverse dynamics control of floating base systems using orthogonal decomposition. In 2010 IEEE international conference on robotics and automation (pp. 3406-3412). IEEE. - citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.212.3601&rep=re

Thank you!

Lecture slides are available via Moodle.

You can help improve these slides at:
github.com/SergeiSa/Control-Theory-Slides-Spring-2021

Check Moodle for additional links, videos, textbook suggestions.