

Solving Mechanical Equations with Constraints

Contact-aware Control, Lecture 5

by Sergei Savin

Fall 2020

- Solving first order ODEs
- Solving second order ODEs
- Mechanical equations
- Mechanical equations with constraints
- Constraint drift
 - ▶ Part 1: right-hand-side
 - ▶ Part 2-3: velocities
 - ▶ Part 4-6: positions
- Read more
- Homework

Solving first order ODEs

General form first order ODE has the form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t) \quad (1)$$

Solving it, means finding such $\mathbf{x}(t)$ that eq. (1) becomes equality. A typical strategy for solving it is a "forward" scheme, for example:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \mathbf{f}(\mathbf{x}_i, t)\Delta t \quad (2)$$

which is called forward Euler scheme. Backward (implicit) scheme would only be different in the function \mathbf{f} being evaluated at the point \mathbf{x}_{i+1} : $\mathbf{f}(\mathbf{x}_{i+1}, t)$ Other famous schemes are Runge-Kutta, variable step schemes, etc.

Solving second order ODEs

Second order ODE have the form:

$$\ddot{\mathbf{x}} = \mathbf{g}(\dot{\mathbf{x}}, \mathbf{x}, t) \quad (3)$$

Observation 1

Every second order differential equation expressed in the normal form can be re-expressed as a system of first order ODEs.

But we don't have to go this rout. Instead we can, for example, use Taylor expansion to solve the equations:

$$\begin{cases} \dot{\mathbf{x}}_{i+1} = \dot{\mathbf{x}}_i + \mathbf{g}(\dot{\mathbf{x}}_i, \mathbf{x}_i, t)\Delta t \\ \mathbf{x}_{i+1} = \mathbf{x}_i + \dot{\mathbf{x}}_i\Delta t + 0.5\mathbf{g}(\dot{\mathbf{x}}_i, \mathbf{x}_i, t)\Delta t^2 \end{cases} \quad (4)$$

Mechanical equations

Consider this special form of second order differential equations:

$$\mathbf{H}\ddot{\mathbf{q}} + \mathbf{c} = \boldsymbol{\tau} \quad (5)$$

First thing you can do is to rewrite it in the form where higher order derivatives are collected on the left hand side:

$$\ddot{\mathbf{q}} = \mathbf{H}^{-1}(\boldsymbol{\tau} - \mathbf{c}) \quad (6)$$

Notice that for mechanical systems \mathbf{H} is always invertible. Prove it, using the fact \mathbf{H} is the quadratic form matrix of the kinetic energy: $T = 0.5\dot{\mathbf{q}}^\top \mathbf{H}\dot{\mathbf{q}}$

Next we either turn it into a system of first order ODEs and solve it using methods for those, or leave it a second order ODE and solve it, for example, using (4).

Mechanical equations with constraints

Part 1

Now, how do we solve this system of equations?

$$\begin{cases} \mathbf{H}\ddot{\mathbf{q}} + \mathbf{c} = \boldsymbol{\tau} + \mathbf{F}^\top \lambda \\ \mathbf{g}(\mathbf{q}) = 0 \end{cases} \quad (7)$$

Notice there are two unknown: \mathbf{q} and λ . One has a second order derivative, another one is an algebraic variable. Also, notice that the first vector equation in the case includes derivatives, while the other does not. Equations like this are called DAE - *differential algebraic equations*.

There are a number of methods of solving DAE, but when one of the equation is algebraic, the rule of thumb is to differentiate it first, so we can use it in finding higher order derivatives.

Mechanical equations with constraints

Part 2

After differentiating $\mathbf{g}(\mathbf{q}) = 0$ twice, and defining $\mathbf{F} = \frac{\partial \mathbf{g}}{\partial \mathbf{q}}$, we get:

$$\begin{cases} \mathbf{H}\ddot{\mathbf{q}} + \mathbf{c} = \tau + \mathbf{F}^\top \lambda \\ \mathbf{F}\ddot{\mathbf{q}} + \dot{\mathbf{F}}\dot{\mathbf{q}} = 0 \end{cases} \quad (8)$$

Now we can treat both $\ddot{\mathbf{q}}$ and λ as unknown algebraic variables, and solve for them. This leads to a system:

$$\begin{bmatrix} \mathbf{H} & -\mathbf{F}^\top \\ \mathbf{F} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \tau - \mathbf{c} \\ -\dot{\mathbf{F}}\dot{\mathbf{q}} \end{bmatrix} \quad (9)$$

Matrix on the left hand side is invertible, as long as the reaction forces are uniquely defined (in other words, as long as \mathbf{F}^\top has a trivial null space).

Mechanical equations with constraints

Part 3

Let us define $\mathbf{M} = \begin{bmatrix} \mathbf{H} & -\mathbf{F}^\top \\ \mathbf{F} & \mathbf{0} \end{bmatrix}$. Assuming that you found left inverse of \mathbf{M} - let us call it matrix \mathbf{L} , $\mathbf{LM} = \mathbf{I}$ - you write expressions for both $\ddot{\mathbf{q}}$ and λ in terms of its components:

$$\begin{cases} \ddot{\mathbf{q}} = \mathbf{L}_{11}(\tau - \mathbf{c}) - \mathbf{L}_{12}\dot{\mathbf{F}}\dot{\mathbf{q}} \\ \lambda = \mathbf{L}_{21}(\tau - \mathbf{c}) - \mathbf{L}_{22}\dot{\mathbf{F}}\dot{\mathbf{q}} \end{cases} \quad (10)$$

This is useful both for simulation and for getting an insight into the dynamical system behaviour.

Constraint drift

Part 1

If we simply integrate the (10), the error will gradually accumulate and values of $\dot{\mathbf{q}}$ and \mathbf{q} will drift, even as $\ddot{\mathbf{q}}$ is calculated precisely.

There are a number of techniques for fixing the constraint drift. Here we will see how to use projection onto the constraint manifold to minimize the drift in the direction, violating constraints. Constraint manifold here is:

$$\begin{cases} \mathbf{g}(\mathbf{q}) = 0 \\ \mathbf{F}\dot{\mathbf{q}} = 0 \end{cases} \quad (11)$$

Constraint drift

Part 2 - velocities

Let us consider the velocity constraint equation $\mathbf{F}\dot{\mathbf{q}} = 0$. All solutions to this equation are found in the form:

$$\dot{\mathbf{q}} = \mathbf{N}\mathbf{z} \quad (12)$$

where $\mathbf{N} = \text{null}(\mathbf{F})$ - null space of the matrix \mathbf{F} , and \mathbf{z} are parameters in the minimal coordinate representation (encoding) of the null space solution for this linear system.

Constraint drift

Part 3 - velocities

We need to solve the following optimization:

$$\begin{aligned} & \underset{\mathbf{z}, \dot{\mathbf{q}}}{\text{minimize}} && ||\dot{\mathbf{q}}_e - \dot{\mathbf{q}}|| \\ & \text{subject to} && \dot{\mathbf{q}} = \mathbf{N}\mathbf{z} \end{aligned} \tag{13}$$

where $\dot{\mathbf{q}}_e$ is the estimate of the generalized velocity, possibly containing the drift.

Substituting, we acquire a simple least squares problem of minimizing $||\dot{\mathbf{q}}_e - \mathbf{N}\mathbf{z}||$, whose solution is $\mathbf{z} = \mathbf{N}^+ \dot{\mathbf{q}}_e$. Final solution is:

$$\dot{\mathbf{q}} = \mathbf{N}\mathbf{N}^+ \dot{\mathbf{q}}_e \tag{14}$$

We can recognize it as a simple projection of the estimate $\dot{\mathbf{q}}_e$ onto the column space of \mathbf{N} .

Constraint drift

Part 4 - positions

Now we can consider position. Equation $\mathbf{g}(\mathbf{q}) = 0$ is non-linear, so we can use Taylor expansion on it:

$$\mathbf{g}(\mathbf{q}) \sim \mathbf{g}_0 + \mathbf{F}(\mathbf{q} - \mathbf{q}_0) \quad (15)$$

where \mathbf{q}_0 is the linearization point; you can choose the current position estimate \mathbf{g}_e or the current desired position or anything else. Be aware that your choice will affect your solution.

General solution to the equation $\mathbf{g}_0 + \mathbf{F}(\mathbf{q} - \mathbf{q}_0) = 0$ is:

$$\mathbf{q} = \mathbf{q}_0 - \mathbf{F}^+ \mathbf{g}_0 + \mathbf{N}\zeta \quad (16)$$

where ζ serves the same role as \mathbf{z} did previously, except for positions.

Constraint drift

Part 5 - positions

We need to solve the following optimization to fix position drift:

$$\begin{aligned} & \underset{\zeta, \mathbf{q}}{\text{minimize}} && ||\mathbf{q}_e - \mathbf{q}|| \\ & \text{subject to} && \mathbf{q} = \mathbf{q}_0 - \mathbf{F}^+ \mathbf{g}_0 + \mathbf{N}\zeta \end{aligned} \tag{17}$$

Same as before, we get a least-squares problem of minimizing $||\mathbf{q}_e - \mathbf{q}_0 + \mathbf{F}^+ \mathbf{g}_0 - \mathbf{N}\zeta||$, with the solution $\zeta = \mathbf{N}^+(\mathbf{q}_e - \mathbf{q}_0 + \mathbf{F}^+ \mathbf{g}_0)$. Finally, the expression for the fixed position is:

$$\mathbf{q} = \mathbf{q}_0 - \mathbf{F}^+ \mathbf{g}_0 + \mathbf{N}\mathbf{N}^+(\mathbf{q}_e - \mathbf{q}_0 + \mathbf{F}^+ \mathbf{g}_0) \tag{18}$$

and after collecting the terms:

$$\mathbf{q} = (\mathbf{I} - \mathbf{N}\mathbf{N}^+)(\mathbf{q}_0 - \mathbf{F}^+ \mathbf{g}_0) + \mathbf{N}\mathbf{N}^+ \mathbf{q}_e \tag{19}$$

Constraint drift

Part 6 - positions

We can recognize that this expression:

$$\mathbf{q} = (\mathbf{I} - \mathbf{N}\mathbf{N}^+)(\mathbf{q}_0 - \mathbf{F}^+\mathbf{g}_0) + \mathbf{N}\mathbf{N}^+\mathbf{q}_e \quad (20)$$

is a projection of the particular solution $\mathbf{q}_0 - \mathbf{F}^+\mathbf{g}_0$ onto the left null space of \mathbf{N} and of the initial estimate \mathbf{q}_e onto the column space of \mathbf{N} .

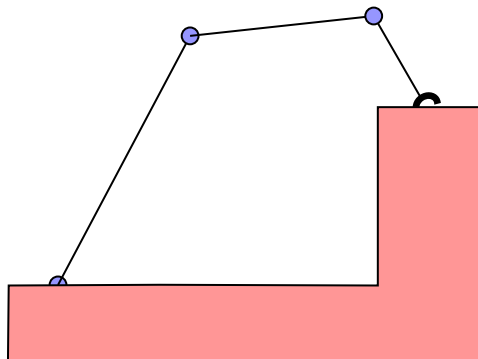
All these projections can be achieved with a single SVD decomposition of the matrix \mathbf{F} to find \mathbf{N} , since \mathbf{N} is orthonormal and its pseudoinverse is equivalent to its transpose.

You can read more at:

- Führer, C. and Leimkuhler, B.J., 1991. Numerical solution of differential-algebraic equations for constrained mechanical motion. *Numerische Mathematik*, 59(1), pp.55-69. - read to get the feeling of the field.

Homework

Simulate the motion of this robot forward (description of the robot is given in the previous lectures).



Lecture slides are available via Moodle.

You can help improve these slides at:

github.com/SergeiSa/Contact-Aware-Control-Slides-Fall-2020

Check Moodle for additional links, videos, textbook suggestions.