Last Time:

- Contact Intro
- Discrete Mechanics with Impacts

Today:

- Coulomb Friction
- Maximimum Dissipation Principle
- LCP Methods

Coulomb Friction

- Last time we talked about non-smooth impacts

- We formulated this as an optimization problem

$$\min_{q(t)} \int_{t_0}^{t_f} L(q, \dot{q})\, dt \quad \leftarrow \text{least action}$$

$$\text{s.t.} \quad \phi(q) \geq 0 \quad \forall t \quad \leftarrow \text{interpenetration constraint}$$

- Discretized KKT Conditions:

$$D_2 L_d(q_{n-1}, q_n) + D_1 L_d(q_n, q_{n+1}) + h n_n^T D_1 \phi(q_n) = 0$$

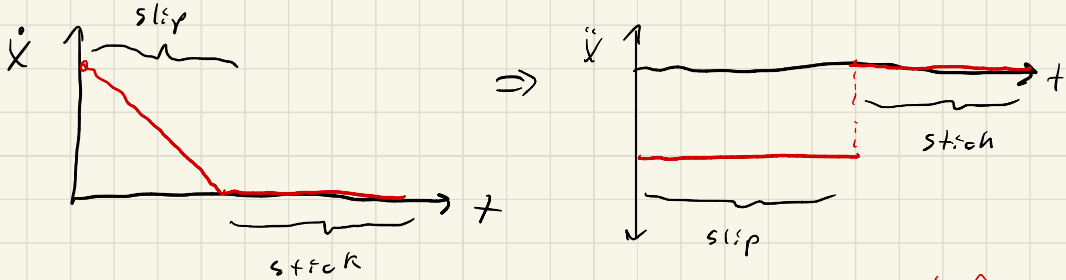
$$\phi(q_{n+1}) \geq 0$$

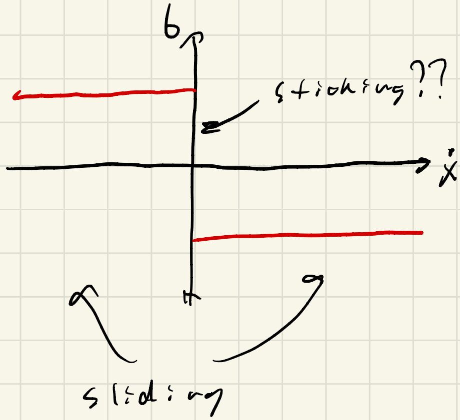
Lagrange multiplier (normal force) →

$$n_n \geq 0$$

$$n_n \phi(q_{n+1}) = 0$$

- Coulomb friction also produces non-smooth dynamics:





normal force

$$b = -\mu n \,\text{sign}(\dot{x})$$

friction coefficient

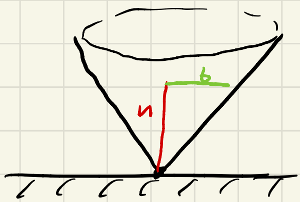
Maximum Dissipation Principle:

- Smooth contact models always have "creep"
- To model "true" Coulomb friction we solve another optimization problem:

$$\min_{b} \; \dot{T}$$ ← "dissipation" = (negative) time derivative of kinetic energy

$$\text{s.t.} \quad \|b\| \leq \mu n$$

- The constraint is called the friction cone:



- We can easily calculate $\dot{T}$:

$$T = \frac{1}{2}\dot{q}^T M(q)\dot{q} \Rightarrow \dot{T} = \underbrace{\frac{\partial T}{\partial \dot{q}}\ddot{q}}_{\dot{q}^T M(q)\ddot{q}} + \cancel{\frac{\partial T}{\partial q}}\dot{q}$$

no b dependence

$$M(q)\ddot{q} + C(q,\dot{q}) + G(q) = J_{(q)}^T b + \dots$$

$$\Rightarrow \ddot{q} = M^{-1}J^T b + \dots$$

maps tangent forces at contact point into generalized forces

- Plugging this back in:

$$\min_b \ \dot{q}^T J_{(q)}^T b$$

linear objective (in b)

$$\text{s.t. } \|b\| \le \mu n$$

cone constraint

$\Rightarrow$ This is a convex optimization problem called a second-order cone program (SOCP)

- We can easily discretize this:

$$\min_{b_n} \left(\frac{q_{n+1}-q_n}{h}\right)^T J^T b_n$$

$$\text{s.t.} \quad \|b_n\| \leq \mu n_n$$

- KKT Conditions:

$$J\left(\frac{q_{n+1}-q_n}{h}\right) + \lambda \frac{b_n}{\|b_n\|} = 0$$

$$\mu n_n - \|b_n\| \geq 0$$

lagrange multiplier $\longrightarrow$ $\lambda \geq 0$

$$\lambda(\mu n_n - \|b_n\|) = 0$$

- We can see that b_n is always in the opposite direction of tangential velocity and λ takes on the magnitude of the tangential velocity.

- Note that these KKT conditions have issues near $\|b\| = 0$ as written. This isn't an issue with primal-dual interior point methods.

- A simple hack that we'll use for now is to smooth the 2-norm:

$$\|b\| \approx \sqrt{b^T b + \varepsilon^2} - \varepsilon, \quad \varepsilon \approx 10^{-6}$$

- This allows IPOPT to converge to $tol \approx \varepsilon$

- Use same complementarity relaxation trick as last time.

- Now we just stack impact + max. disp. KKT conditions and jointly solve for $q_{n+1}, \lambda_n, \beta_n$

* Example

LCP Methods:

- The methods we derived are nice but computationally expensive.

- Most current simulators (Bullet, Dart, etc.) make additional approximations.

- First, use Euler integration:

$$M(q_n)\left(\frac{V_{n+1}-V_n}{h}\right) + C(q_n, v_n) + G(q_n) = \underbrace{J^T(q_n)\,F}_{\text{contact forces}} + B(q_n)\,u$$

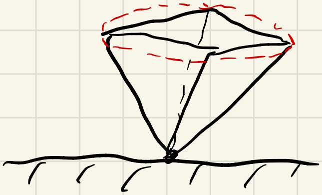
$$q_{n+1} = q_n + hV_{n+1}$$

- Not that this is linear in q_{n+1}, V_{n+1}

- Next, linearize the interpenetration constraint

$$\phi(q_{n+1}) \simeq \phi(q_n) + \frac{\partial\phi}{\partial q_n}(h\,V_{n+1})$$

- Last, linearize friction cone by making it a pyramid:



$$\|b\|_2 \le \mu n$$

$$\Downarrow$$

$$\|b\|_1 \le \mu n$$

- We can write this as:

$$b = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix}}_{d}$$

$$d \ge 0$$

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}}_{e^T} d \le \mu n$$

- Now all of the pieces of the problem are linear except the complementarity conditions

- This can be put into the standard form of a linear complementarity problem (LCP):

$$Ax + y = z$$

$$x \ge 0$$

$$z \ge 0$$

$$x \odot z = 0$$

- LCPs are closely related to QPs and there are nice/fast standard solvers.

- Most current time-stepping simulators (e.g. Bullet, Dart) use this LCP formulation.