Torque and Force Control

Topics

- Hybrid Position-Force Control
- Force Control (J-Transpose and Stiffness Control)
- Non-linear Control (Computer Torque Control, Impedance Control)

Complementary Reading: J.J. Craig, Introduction to Robotics, Chapter 9,10,11.

Overview

Cover a variety of methods for controlling position and force using linear and nonlinear control.

Control Schemes	Workspace	Measured Variables	Applied Situations	Control Objectives
Position/Velocity Control	Cartesian Space	Position, Velocity	Free motion	Desired position/velocity
Force/Torque Control	Cartesian Space	Contact Force	Constrained Motion	Desired force/torque
Hybrid Control	Position and Force Subspaces	Position and Contact Force	All situations	Desired position and force
Impedance Control	Joint or Cartesian Space	Position and Contact Force	All situations	Impedance

Force / Torque Control

Force / Torque Control refers to controlling the forces and torques between *the robot and its environment.*

- Not to be confused by computed-torque control which follows kinematic trajectories.

Force Control using Jacobian Transpose

Assume:

- Quasi-static cases (i.e. ignoring inertial effects)
- "Full Transparency": there is nothing between the motor output and the member acting on the environment that dissipates the transmitted torque.

This turns out to be quite simple:

$$\tau = J^\top F$$

Proof:

$$\dot{x}^\top F = \dot{q}^\top \tau$$

$$(J\dot{q})^\top F = \dot{q}^\top \tau$$

$$\dot{q}^\top J^\top F = \dot{q}^\top \tau$$

$$\dot{q}^\top (J^\top F - \tau) = 0$$

Note: τ is added with gravity compensation $G(q)$ to keep the arms up while controlling force. Thus the applied torque to the joints is $\tau = J^\top F + G(q)$

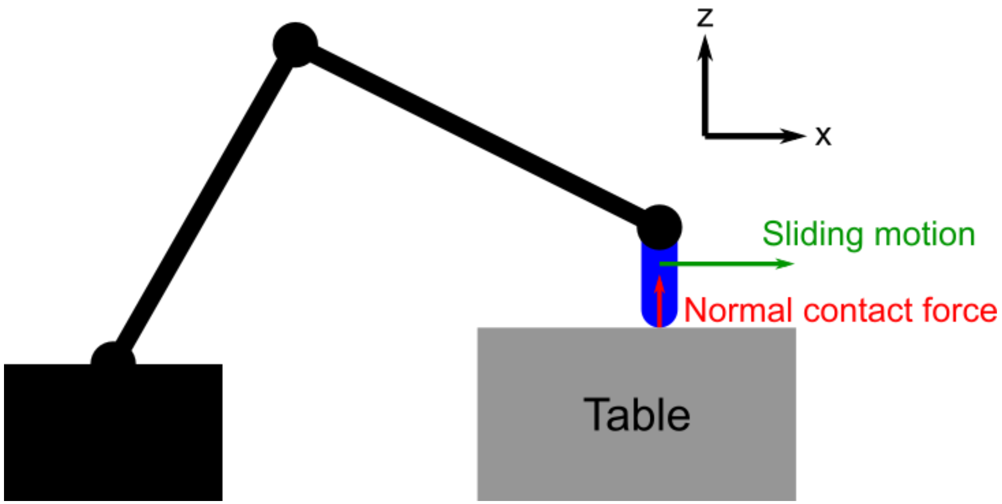
Position/Force Control: **Hybrid Control**

Basic concept:

- Control position, but if you do touch something regulate the forces on it

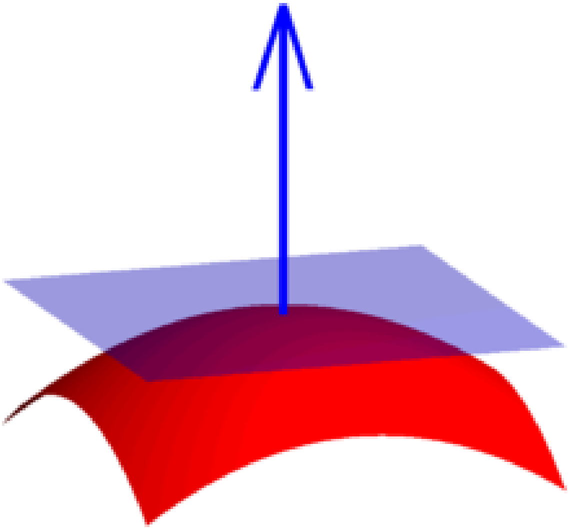
Mathematically:

- Regulate positions in free space
- If being constrained, regulate forces in directions normal to an object's surface, and regulate positions in all other directions.



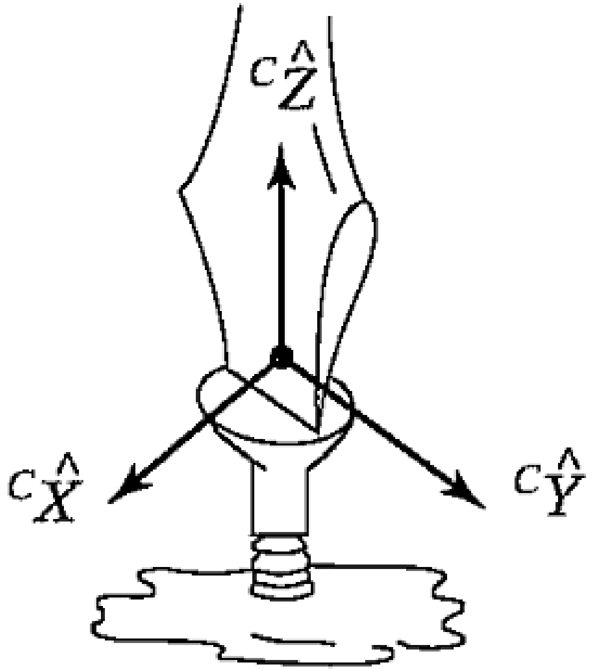
Rigorous Definition of Constraints

1. For every natural position constraint, there is an artificial force constraint. (e.g. if $v_x = 0$, then we get to control $f_x = \alpha$)
2. For every natural force constraint, there is an artificial position constraint. (e.g. if $f_x = 0$, then we get to control $v_x = \alpha$)



Examples

(b) Turning screwdriver



Natural constraints

$v_x = 0$ $\quad$ $f_y = 0$

$\omega_x = 0$ $\quad$ $n_z = 0$

$\omega_y = 0$

$v_z = 0$

Artificial constraints

$v_y = 0$ $\quad$ $f_x = 0$

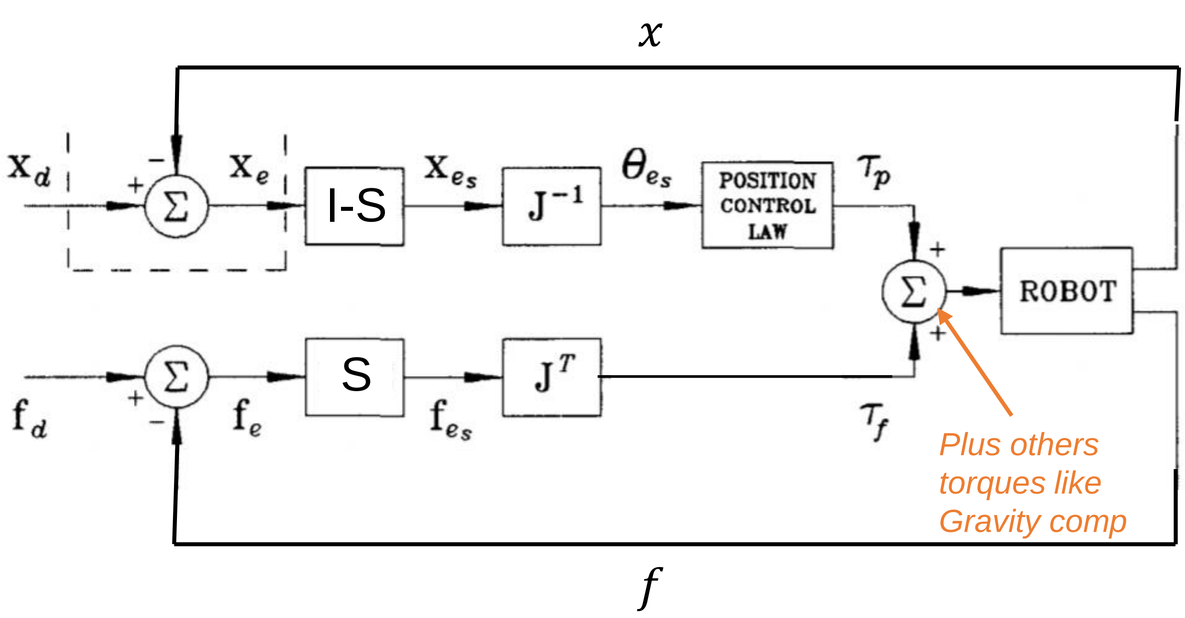
$\omega_z = \alpha_2$ $\quad$ $n_x = 0$

$n_y = 0$

$f_z = \alpha_3$

Hybrid Position/Force Control

- Define an Projection Matrix S:
  - $S = u_F u_F^\top$ is a projection matrix for a vector into a contact normal u_F
  - $S' = I - S$ projects remaining components to the surface tangent plane
- Apply S and S' as masks to position and force trajectories, *in the coordinate frame of the end-effector*, so that they may be regulated in orthogonal directions.



Stiffness Control

- If your robot is backdriveable, you can control stiffness $f = k(x_{ref} - x)$ rather than the position x or force f itself.
  - Beneficial if you want to make the robot feel like a spring
  - Doesn't work for robots that cannot be backdriven (i.e. they are much stiffer than the environment, or they will break if you push them too hard)
- Assume a 6-DOF spring force in Cartesian coordinates (x,y,z,r,p,y)

$$F = K_x(x_{ref} - x) = K_x \Delta x$$

- Since $\tau = J^\top F$ maps end-effector forces to torques on the motors, then

$$\tau \coloneqq J^\top K_x(x_{ref} - x)$$

$$\coloneqq J^\top K_x\left(x_{ref} - f(q)\right) \quad \text{or}$$

$$\approx J^\top K_x J(q_{ref} - q) \qquad \text{(if } \Delta x \text{ is small)}$$

Impedance Control

- Desire our robot to behave like a **mass-spring-damped system**
  - Safe to the robot and the environment
  - Stable
- Compute a nonlinear model-based control law that "cancels" the nonlinearities of the system to be controlled.
- Still considered a “position-based” control, though accounting for dynamics.
- Reduce the system to the appearance of a linear dynamical system

$$\dot{x} = Ax + Bu$$

- Can then be controlled with the simple linear control law

Example

- Consider a particle system that moves under

$$m\ddot{x} + b\dot{x} + cx = f$$

- Construct an applied force as a function of $\ddot{x}, \dot{x}, x$ such that

$$f \coloneqq \widehat{m}\left(\ddot{x}_{ref} + k_d \dot{e}_x + k_p e_x\right) + \hat{b}\dot{x} + \hat{c}x$$

- Apply controlled force to particle system:

$$m\ddot{x} + b\dot{x} + cx = \widehat{m}\left(\ddot{x}_{ref} + k_d \dot{e}_x + k_p e_x\right) + \hat{b}\dot{x} + \hat{c}x$$

- If $\hat{b} \approx b$, $\hat{c} \approx c$, $\widehat{m} \approx m$, then

$$\ddot{x} = \ddot{x}_{ref} + k_d \dot{e}_x + k_p e_x$$

- Or, written another way,

$$\ddot{e}_x + k_d \dot{e}_x + k_p e_x = 0$$

The solution to the ODE (via Laplace Transforms):

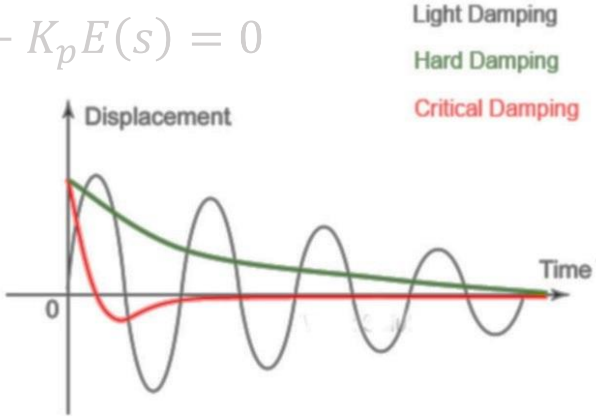
[let $e_x(0) = C_1$, $\dot{e}_x(0) = C_2$]

$$\left(s^2 E(s) - s e_x(0) - \dot{e}_x(0)\right) + K_d\left(sE(s) - e_x(0)\right) + K_p E(s) = 0$$

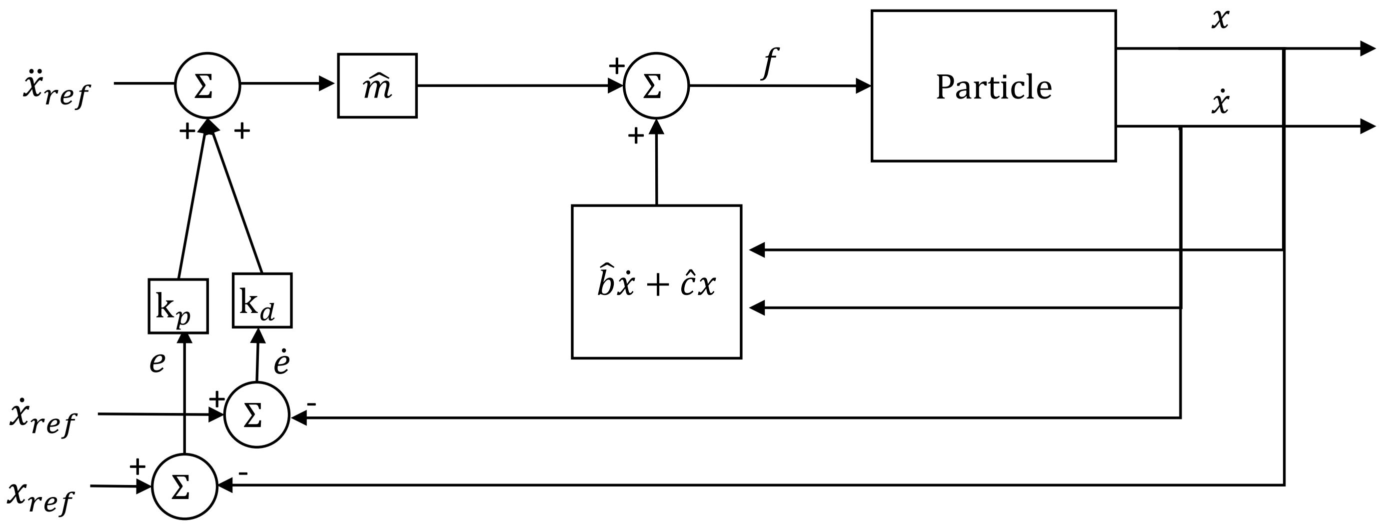
$$s^2 E(s) + K_d s E(s) + K_p E(s) = C_1 s + C_2 + K_p C_1$$

$$\left(s^2 + K_d s + K_p\right) E(s) = C_1 s + C_2 + K_p C_1$$

$$E(s) = \frac{C_1 s + (C_2 + K_p C_1)}{s^2 + K_d s + K_p}$$



- The higher the k_p, k_d the faster the system will accelerate to try to make $e_x, \dot{e}_x$ go to zero, then $\ddot{e}_x = 0$ and the system will be exactly matching your reference.



Computed Torque Control for a Multi-body Robot

For $D(q)\ddot{q} + C(q,\dot{q}) + G(q) = U$

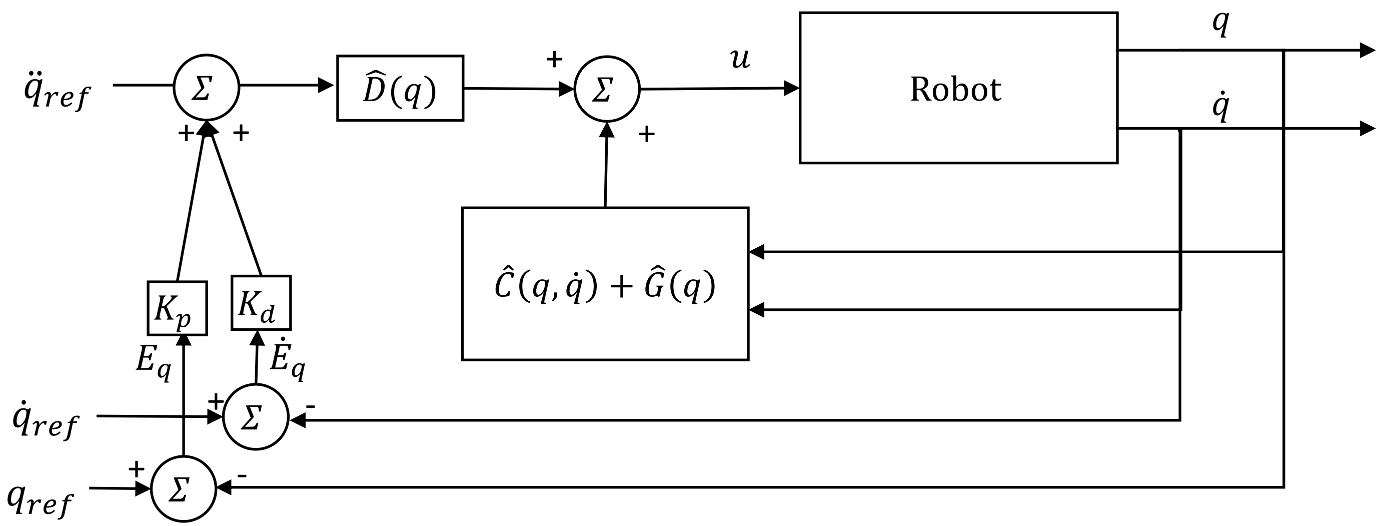
Then choose U:

$$U = \hat{D}(q)\left(\ddot{q}_{ref} + K_d\dot{E}_q + K_pE_q\right) + \hat{C}(q,\dot{q}) + \hat{G}(q)$$

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = \hat{D}(q)\left(\ddot{q}_{ref} + K_d\dot{E}_q + K_pE_q\right) + \hat{C}(q,\dot{q}) + \hat{G}(q)$$

Then

$$\ddot{E}_q + K_d\dot{E}_q + K_pE_q = 0$$



- If there is an end-effector force and/or losses,

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = U + J(q)^{\top}F_{ee} + F_{loss}$$

then

$$U = \widehat{D}(q)\left(\ddot{q}_{ref} + K_d\dot{E}_q + K_pE_q\right) + \hat{C}(q,\dot{q}) + \widehat{G}(q) - J(q)^{\top}F_{ee} - F_{loss}$$

to ensure that $\ddot{E}_q + K_d\dot{E}_q + K_pE_q = 0$

Remarks:

$$\ddot{E}_q + K_d\dot{E}_q + K_pE_q = 0$$

Describes a series of separable mass spring dampers if K_d, K_p are diagonal.

Impedance Control

As a generalization of **computed torque control**, we often hear about **impedance control**, which makes the robot behave like a mass-spring-damper in *Cartesian Space*.

$$M_d\ddot{x} + B(\dot{x}_d - x) + K(x_d - x) = 0$$

- Step 1: Describe robot dynamics equations, $D(q)\ddot{q} + \tilde{C}(q,\dot{q}) + G(q) = U$, in Cartesian coordinates, i.e., replace all q's with x's and group like terms.

$$\dot{x} = J\dot{q} \qquad \rightarrow \qquad \dot{q} = J^{-1}\dot{x}$$

$$\ddot{x} = \dot{J}\dot{q} + J\ddot{q} \qquad \rightarrow \qquad \ddot{q} = J^{-1}\ddot{x} - J^{-1}\dot{J}J^{-1}\dot{x}$$

Step 2: like computed torque formulation, find U that to match equation above.

Resulting in $M_x\ddot{x} + S_x\dot{x} + G_x = J^{-\top}U$

Then by substitution and rearrangement,

$$D\left(J^{-1}\ddot{x} - J^{-1}\dot{J}J^{-1}\dot{x}\right) + \tilde{C}J^{-1}\dot{x} + G = u_{act} + J^{\top}F_{ee} + F_{loss}$$

$$J^{-\top}DJ^{-1}\ddot{x} - J^{-\top}DJ^{-1}\dot{J}J^{-1}\dot{x} + J^{-\top}\tilde{C}J^{-1}\dot{x} + J^{-\top}G = J^{-\top}u_{act} + F_{ee} + F_{loss}$$

$$M_x\ddot{x} + S_x\dot{x} + G_x = J^{-\top}u_{act} + F_{ee} + F_{loss}$$

Projected Robot Equations of Motion

where

$$M_x(q) = J^{-\top}DJ^{-1}$$

$$S_x(q,\dot{q}) = -J^{-\top}DJ^{-1}\dot{J}J^{-1} + J^{-\top}\tilde{C}J^{-1}$$

$$G_x(q) = J^{-\top}G$$

Then, by choosing a control law:

$$u_{act} = J^{\top}\left(M_x\left(M_d^{-1}\left(B_x(\dot{x}_d - \dot{x}) + K_x(x_d - x)\right)\right) + S_x\dot{x}_d + G_x - F_{ee} - F_{loss}\right)$$

we get

$$M_d\ddot{x} + B_x(\dot{x}_d - \dot{x}) + K_x(x_d - x) = 0$$