ME 599/699 Robot Modeling & Control
Robot Control

Instructor: Hasan Poonawala

Contents

1 Dynamical Systems

A general dynamical system has a state $x(t) \in X$ at time t, where X is the state space of the system. For mechanical system, the state often comprises both the configuration q and its derivative $\dot{q}$.

Many dynamical systems have an n-dimensional state space, and the evolution of the state with time is given by an ordinary differential equation:

$$\dot{x}(t) = f(x, t). \tag{1}$$

For many systems, f only depends on the current state, not on time. These systems are known as autonomous systems.

A controlled dynamical system typically also allows an additional signal u known as the control signal:

$$\dot{x}(t) = f(x, u, t). \tag{2}$$

Suppose f is time-invariant, and we choose a state-based feedback control $u = k(x)$. The dynamics becomes

$$\dot{x}(t) \ = f(x, k(x)) = f_{cl}(x), \tag{3}$$

which is again an autonomous system.

1.1 Solutions Of ODEs

A solution $x^*\colon [t_0, t_f] \mapsto X$ is a function that maps each time t in the interval $[t_0, t_f]$ to a unique state $x \in X$, denoted $x^*(t)$. The state at time t_0 is called the initial condition.

If one is given a map $\bar{x}\colon [t_1, t_2] \mapsto X$, then $\bar{(t)}$ is a solution to the ODE if for all $\bar{t} \in [t_1, t_2]$

$$\frac{d}{dt}\bar{x}(t)|_{\bar{t}} = f(\bar{x}(\bar{t}), \bar{t}) \tag{4}$$

1.2 Stability

The study of dynamical systems is often concerned with the existence of equilibria and the properties of these equilibria. For now, we focus on autonomous systems.

Definition 1 (Equilibrium)**.** An *equilibrium* point $x_e \in X$ is a point such that $f(x_e, t) = 0$ for all t.

The main property of equilibria is their stability.

Definition 2 (Stable)**.** An equilibrium point x_e is *stable* if for every $\epsilon > 0$, there exists $\delta > 0$ such that every solution $x(t)$ with initial condition $x(t_0) \in B_\delta(x_e)$ is such that $x(t) \in B_\epsilon(x_e)$.

In other words, solutions that start close stay close, no matter how small you define staying close to be.

Definition 3 (Asymptotically Stable)**.** An equilibrium point x_e is *asymptotically stable* if it is stable and for every solution $x(t)$ with initial condition $x(t_0) \in B_\delta(x_e)$ for some $\delta > 0$

$$\lim_{t \to \infty} x(t) = x_e \tag{5}$$

In other words, solutions not only stay close, they return back to x_e in a long enough time frame.

Definition 4 (Exponentially Stable)**.** An equilibrium point x_e is *asymptotically stable* if it is stable and for every solution $x(t)$ with initial condition $x(t_0) \in B_\delta(x_e)$ for some $\delta > 0$

$$\lim_{t \to \infty} x(t) = x_e \tag{6}$$

In other words, solutions not only stay close, they return back to x_e in a long enough time frame.

2 Linear Dynamical Systems

Consider the linear dynamical system

$$\dot{x}(t) = Ax(t) + Bu. \tag{7}$$

If $u \equiv 0$, then the dynamical system is stable if $Re(\lambda) \leq 0$ where λ is any eigenvalue of A. If $u \equiv 0$, then the dynamical system is **asymptotically** stable if $Re(\lambda) < 0$.

Suppose that there is an eigenvalue $\lambda^{us} \in \mathbb{C}$ of A such that $Re(\lambda^{us}) > 0$. The system is unstable when run in open loop ($u \equiv 0$). Suppose we choose $u = -Kx$. Then

$$\dot{x}(t) = (A - BK)x(t). \tag{8}$$

Naturally, we want to choose K so that all eigenvalues of $A - BK$ have non-negative real part.

2.1 Transfer Functions

The state-based representation focuses on the system state. Instead, we can define an output $y = Cx + Du$, and understand the system not through its (internal) state x, but merely through the relationship between its input u and output y. That is, given some input signal $u(t)$, what will the output signal $y(t)$ be?

As seen in introductory courses (ME 340), working in the frequency domain is a much easier way to answer this question, leading to a transfer function representation:

$$Y(s) = G(s)U(s), \tag{9}$$

where

$$G(s) = C(sI - A)^{-1}B + D. \tag{10}$$

For a single-input single-output system, the transfer function $G(s)$ is the quotient of two real polynomial functions of the complex variable $s = \sigma + j\omega$, i.e., $G(s) = n(s)/d(s)$. For multi-input multi-output systems, the transfer function $G(s)$ is a matrix of such SISO transfer functions.

The roots $s_i = \sigma_i + j\omega_i$ of the equation $d(s) = 0$ are known as the poles of the transfer function. Stability requires that all poles are in the closed left half plane, i.e., $\sigma_i \leq 0$ for all poles s_i.

3 Independent Joint Control

Using the machinery so far, we can take desired end effector positions and velocities and construct a trajectory for the link angles that achieves our task. For each link i, we construct a trajectory $q_i^d(t)$, implying $\dot{q}_i^d(t)$.

The problem then becomes how to make our link angles track those desired angles and rate of change of angles. Mathematically, we want

$$\lim_{t\to\infty} q_i(t) \to q_i^d(t).$$

To get there, we have to understand the physical implementation of the torques τ on the link.

3.1 Actuators

To control a joint i, corresponding to link angle θ_i, we typically rigidly attach links $i-1$ and i to the housing/stator and shaft/rotor of a rotary actuator respectively. For prismatic joints, these links $i-1$ and i are rigidly attached to the housing and piston of a linear actuator, respectively. The output of the motor becomes the force/torque τ, unless a gear-like mechanism is introduced, at which point the torque τ on the link is some multiple of the motor's output, say τ_m.

We consider permanent magnet DC motors. Other kinds include *AC* motors and brushless *DC* motors.

A model for the motor torque is $\tau = K_m i_a$, if the flux in the motor is constant. The current is generated by a voltage source, and has dynamics

$$L\frac{d}{dt}i_a + Ri_a = V - V_b, \tag{11}$$

where L is the motor inductance, R is the winding resistance, V_b is the back EMF and is proportional to ω_m, the motor speed.

3.2 Joint Model

This section focuses on the most naive approach: independent servo control for each motor attached to each joint. For slow motions, the SISO model we derive is adequate, especially when the gear reductions are large. The gear ratio is typically in the range of 20 to 200 or more.

We have the actuator inertia J_a and gear inertia J_g driven by the motor torque τ_m, with gear friction coefficient B_m. The gear reduces θ_m to θ_s by ratio r. The second shaft is connected to an intertia J_l and driven by load torques τ_l.

The dynamics governing θ_m are

$$\begin{aligned} J_m\ddot{\theta}_m + B_m\dot{\theta}_m &= \tau_m - \tau_l/r \\ &= K_m i_a - \tau_l/r \end{aligned} \tag{12}$$

We can rewrite (11) and (13b) as

$$(Ls + R)I_a(s) = V(s) - K_b \; s\Theta_m(s), \tag{13a}$$

$$(J_m s^2 + B_m s)\Theta_m(s) = K_m I_a(s) - \tau_l(s)/r \tag{13b}$$

We combine these equations to obtain

$$s\left((Ls + R)(J_m s + B_m) + K_b K_m\right)\Theta_m(s) = K_m V(s) - \frac{(Ls + R)}{r}\tau_l(s). \tag{14}$$

The electrical time constant L/R is much smaller than the mechanical time constant J_m/B_m. So, we can divide by R and set L/R to zero, obtaining.

$$s\left((J_m s + B_m) + \frac{K_b K_m}{R}\right)\Theta_m(s) = \frac{K_m}{R}V(s) - \frac{1}{r}\tau_l(s). \tag{15}$$

Setting $u = K_m V/R$ and $d = -\tau_l/r$, we obtain the motor equation as

$$J\ddot{\theta}_m + B\dot{\theta}_m = u(t) - d(t). \tag{16}$$

Alternatively,

$$(Js^2 + Bs)\Theta_m(s) = U(s) - D(s) \tag{17}$$

3.3 Routh Hurwitz Criterion

- The second-degree polynomial, $P(s) = s^2 + a_1 s + a_0$ has both roots in the open left half plane (and the system with characteristic equation $P(s) = 0$ is stable) if and only if both coefficients satisfy $a_i > 0$.
- The third-order polynomial $P(s) = s^3 + a_2 s^2 + a_1 s + a_0$ has all roots in the open left half plane if and only if a_2, a_0 are positive and $a_2 a_1 > a_0$.

3.4 P Control

The simplest control strategy is Proportional control:

$$u(t) = -k_p(\theta_m(t) - \theta_d(t)). \tag{18}$$

The closed-loop model becomes

$$(Js^2 + Bs)\Theta_m(s) = -k_p\Theta_m(s) + k_p\Theta_d(s) - D(s), \tag{19}$$

or

$$\Theta_m(s) = \frac{k_p}{Js^2 + Bs + k_p}\Theta_d(s) - \frac{1}{Js^2 + Bs + k_p}D(s). \tag{20}$$

The error, is therefore

$$E(s) = \Theta_d(s) - \Theta_m(s) = \frac{Js^2 + Bs}{Js^2 + Bs + k_p}\Theta_d(s) - \frac{1}{Js^2 + Bs + k_p}D(s). \tag{21}$$

As long as $k_p > 0$ and disturbances are bounded we get stability. For step disturbance $d(t) = D$ and reference $\theta_d(t) = \theta_d$, we apply the final value theorem to see that

$$e_{ss} = \lim_{s \to 0} sE(s) = -\frac{D}{K_p} \tag{22}$$

So, we can make the errors zero and the effect of disturbance small. What we can't do is shape the transient response, since we have limited control over the closed loop poles. To rectify this, we must use a PD control.

3.5 PD Control

The simplest control strategy is Proportional control:

$$u(t) = -k_p(\theta_m(t) - \theta_d(t)) - K_d\dot{\theta}_m(t). \tag{23}$$

The closed-loop model becomes

$$(Js^2 + Bs)\Theta_m(s) = -k_p\Theta_m(s) + k_p\Theta_d(s) + k_d s\theta_m(s) - D(s), \tag{24}$$

or

$$\Theta_m(s) = \frac{k_p}{Js^2 + (B + k_d)s + k_p}\Theta_d(s) - \frac{1}{Js^2 + (B + k_d)s + k_p}D(s). \tag{25}$$

As long as $k_p > 0$, $k_d > 0$, and disturbances are bounded, the closed loop system is stable. Moreover, we can move both poles arbitrarily.

In practice, we can't move the poles wherever due to actuator saturation. An easier way to at least get e_{ss} to be zero is to use integral action.

3.6 PID Control

$$u(t) = -k_p(\theta_m(t) - \theta_d(t)) - K_d\dot{\theta}_m(t) - k_I\int_0^t(\theta_m(\eta) - \theta_d(\eta))d\eta, \tag{26}$$

$$U(s) = \left(k_p + \frac{k_I}{s}\right)(\Theta_d(s) - \Theta_m(s)) - K_d\Theta_m(s). \tag{27}$$

$$\Theta_m(s) = \frac{k_p s + k_I}{Js^3 + (B + k_d)s^2 + k_p s + k_I}\Theta_d(s) - \frac{s}{Js^3 + (B + k_d)s^2 + k_p s + k_I}D(s). \tag{28}$$

$$E(s) = \frac{Js^3 + (B + k_d)s^2}{Js^3 + (B + k_d)s^2 + k_p s + k_I}\Theta_d(s) - \frac{s}{Js^3 + (B + k_d)s^2 + k_p s + k_I}D(s). \tag{29}$$

Same final value theorem test gives us that $e_{ss} = 0$ for all constant step reference and constant disturbances.

Applying Routh-Hurwitz criterion, we get that $k_p, k_d, k_I > 0$ and

$$k_I < \frac{k_p(B + k_d)}{J} \tag{30}$$

3.7 FeedForward Control

The control approaches so far worked for constant references and disturbances. What happens when the reference is time varying? One approach is to use a feedforward control input.

Let the plant be $G(s)$, the controller be $H(s)$, and feedforward transfer function be $F(S)$. Then,

$$U(s) = F(s)\Theta_d(s) + H(s)\Theta_d(s) \tag{31}$$

One can show that if $F(s) = 1/G(s)$ and $F(s)$ is stable, then $E(s) = \Theta_d(s) - Y(s) = 0$.

Let

$$G(s) = \frac{q(s)}{p(s)}, \qquad H(s) = \frac{c(s)}{d(s)}, \text{ and } \qquad F(s) = \frac{a(s)}{b(s)}. \tag{32}$$

Then

$$T(s) = \frac{\Theta_m(s)}{\Theta_d(s)} = \frac{q(s)\left(c(s)b(s) + a(s)d(s)\right)}{b(s)\left(p(s)d(s) + q(s)c(s)\right)}, \tag{33}$$

and

$$\frac{E(s)}{\Theta_d(s)} = \frac{d(s)\left(q(s)a(s) - b(s)p(s)\right)}{b(s)\left(p(s)d(s) + q(s)c(s)\right)}. \tag{34}$$

We can get $E(s) \equiv 0$ if $q(s)a(s) - b(s)p(s) = 0$, or

$$\frac{q(s)}{p(s)} = \frac{b(s)}{b(a)} \tag{35}$$

$$\implies F(s) = \frac{1}{G(s)} \tag{36}$$

The effect of this choice when $H(s)$ is a PD control is that

$$\left(Js^2 + (B + K_d)s + k_p\right) E(s) = -D(s). \tag{37}$$

The closed loop system can be modeled as

$$J\ddot{e}(t) + (B + K_d)\dot{e}(t) + k_p e(t) = -d(t). \tag{38}$$

If $d(t) = 0$ then $e(t) \to 0$ for $k_p > 0$, $k_d > 0$.

3.8 Advanced Joint Models

There are two issues that affect the performance of the independent joint control strategies derived so far. The first is the issue of actuator saturation, the second is the effect of flexibility in the actuator or link.

The effect of saturation is to create integrator wind-up, and reduce the rise time in response to step functions. Furthermore, ramps may be untrackable.

The effect of joint flexibility is to introduce oscillatory modes, which restricts the bandwidth of the control behavior so that these modes are not excited to resonance. Harmonic gear mechanisms have low backlash, high torque transmission, and compact size. However, they use a flexspline that introduces flexibility.

We model the flexibility as a single spring with spring constant $k[N/m^2]$. Consider such a flexible spring between the motor and the load. The equations are given by

$$J_l\ddot{\theta}_l + B_l\dot{\theta}_l + k(\theta_l - \theta_m) = 0, \text{ and} \tag{39}$$

$$J_m\ddot{\theta}_l + B_m\dot{\theta}_l + k(\theta_m - \theta_l) = u, \tag{40}$$

where u is input torque applied to the motor shaft.

$$p_l(s) = J_l s^2 + B_l s + k \tag{41}$$

$$p_m(s) = J_m s^2 + B_m s + k \tag{42}$$

$$\frac{\Theta_l(s)}{U(s)} = \frac{k}{p_l(s)p_k(s) - k^2} \tag{43}$$

$$\frac{\Theta_m(s)}{U(s)} = \frac{p_l(s)}{p_l(s)p_k(s) - k^2} \tag{44}$$

$$\tag{45}$$

The open-loop characteristic equation is

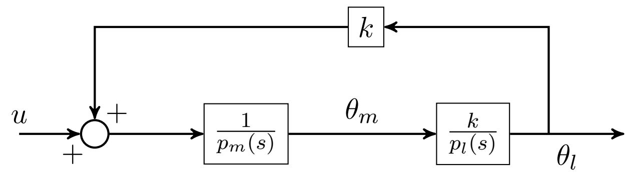
$$J_l J_m s^4 + (J_l B_m + J_m B_l)\, s^3 + (k(J_l + J_m) + B_l B_m)\, s^2 + k(B_l + B_m)s \tag{46}$$

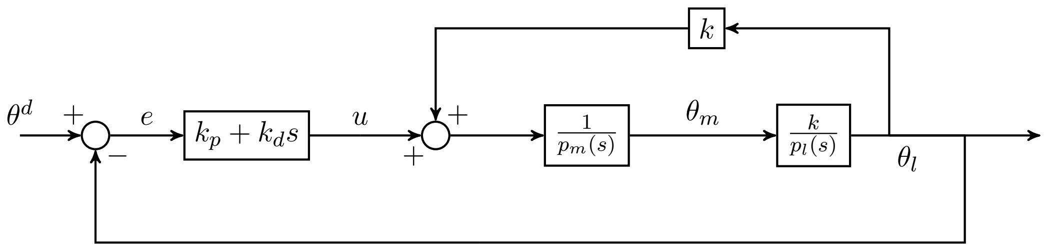
To understand this equation, consider the case where viscous friction is absent. The open-loop characteristic equation is $s^2(J_l J_m s^2 + k(J_l + J_m))$, which has a double pole at the origin and two complex conjugate poles of the imaginary axis. That's a neutrally stable system. The frequency of the imaginary poles are proportional to $\sqrt{(}k)$. For systems with small values of B_l and B_m and high stiffness k, we expect the poles to be close to this situation, indicating that this system is hard to control.

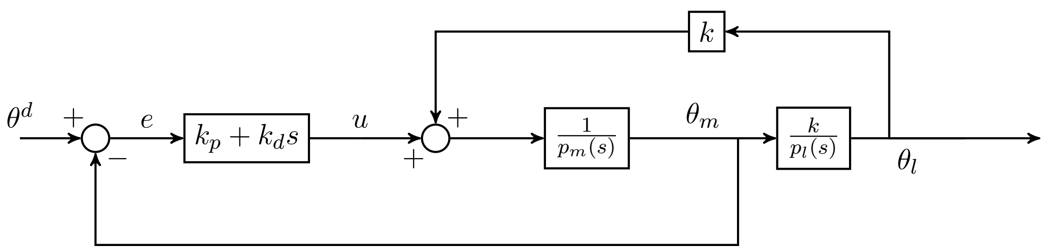
The effectiveness of a PD control will depend on whether the signal is from the motor or the load. The short message is:

1. If you use θ_m, you can control θ_m well but are then letting θ_l be driven by passive dynamics.

2. If you use θ_l you can control θ_l but you have to be less aggressive to not excite a resonant feedback due to the spring.







4 State Space Models

The flexible motor case shows that a PD control using a single measurement might not be suitable. If you use feedback from both θ_m and θ_l, you create four gains that are hard to design using typical frequency-domain techniques.

At this point, one moves to state space models. The flexible joint model can be given a state $x \in \mathbb{R}^4$ where

$$x_1 = \theta_l, \quad x_2 = \dot{\theta}_l, \quad x_3 = \theta_m, \quad x_4 = \dot{\theta}_m. \tag{47}$$

The state space model is

$$\dot{x}_1 = x_2 \tag{48a}$$

$$\dot{x}_2 = -\frac{k}{J_l}x_1 - \frac{B_l}{J_l}x_2 + \frac{k}{J_l}x_3 \tag{48b}$$

$$\dot{x}_3 = x_4 \tag{48c}$$

$$\dot{x}_4 = \frac{k}{J_m}x_1 - \frac{k}{J_m}x_3 - \frac{B_m}{J_m}x_4 + \frac{1}{J_m}u \tag{48d}$$

which we can represent compactly as the linear system

$$\dot{x} = Ax + Bu \tag{49}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{J_l} & -\frac{B_l}{J_l} & \frac{k}{J_l} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k}{J_m} & 0 & -\frac{k}{J_m} & -\frac{B_m}{J_m} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix}. \tag{50}$$

Note that the input u is a scalar. If we know all elements of x, we can choose a linear feedback

$$u = -k^T x + r \tag{51}$$

where r is the reference.

How useful is the choice of k?

If our system is controllable, we can choose k to assign the poles of $A - Bk^T$ however we place, as long as complex poles have their conjugates as poles.

Our system is!

The cost is that we need all of k.

What if we only measure θ_l? Or we only measure θ_m?

One trick to make good use of partial measurements of the state is to build state observers. One deduce the entire state from measurements if the system is observable.

Controllability and Observability

which we can represent compactly as the linear system

$$\dot{x} = Ax + Bu \tag{52}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k}{J_l} & -\frac{B_l}{J_l} & \frac{k}{J_l} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k}{J_m} & 0 & -\frac{k}{J_m} & -\frac{B_m}{J_m} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J_m} \end{bmatrix}. \tag{53}$$

Note that the input u is a scalar.

Controllability

If we know all elements of x, we can choose a linear feedback

$$u = -k^T x + r \tag{54}$$

where r is the reference.

How useful is the choice of k?

If our system is controllable, we can choose k to assign the poles of $A - Bk^T$ however we place, as long as complex poles have their conjugates as poles.

Our system is!

The cost is that we need all of k.

Observability

What if we only measure θ_l? Or we only measure θ_m?

One trick to make good use of partial measurements of the state is to build state observers. One can deduce the entire state from measurements if the system is observable.

Path And Trajectory Planning

I showed you guys a position controlled manipulator, and PID controls that worked. Had to manually tune gains to artificially 'approach' the object.

The trajectory planning problem can be cast as an optimization problem. Suppose we can measure the 'goodness' of a trajectory by a function J. We want to find a solution $q^*(t)$ of the problem:

$$\begin{aligned} \min_{q(t)} \quad & J(q(t)) \\ \text{subject to} \quad & \\ & \text{Robot doesn't destroy itself or things} \\ & \text{Other concerns} \end{aligned}$$

This version of the problem doesn't worry about control. Out pops q^*t and we try and use the trajectory tracking controllers.

Typically solved using heuristics. Why ?

Let's break this down:

Simplest J: all trajectories have value 0 (the same, not useless).

Important thing is to not hit things.

Start with finding a path. Then, attach time to the path to get a trajectory.

- Figure out the Obstacle-free configuration space (difficult, use over-approximations of robot and obstacles)
- Sample points in free space (easy)
- Connect points in free space (depends)

How to connect?

- Potential Field + random walk
- Probabilistic Road Maps
- Rapidly-exploring Random Trees

Attaching time. One approach: use polynomial function of sufficient degree to specify initial point, final point, initial velocity, final velocity, and add accelerations.

Finding coefficients given start and end times and values is a linear program.

5 Multivariable Control

We derived an independent joint model for the robot dynamics, encompassing link and actuators, and designed controllers based on this model. The full model combines the Euler-Lagrangian model for the robot link dynamics with the actuator models to obtain an equation that looks like

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + B\dot{q} + g(q) = u, \tag{55}$$

where u denotes the input due to the voltage, whereas τ was the torque acting on the link joint. In particular,

$$u_k = r_k \frac{K_{m_k}}{R_k} V_k,$$

where $\theta_{m_k} = r_k q_k$, and $M(q) = D(q) + J$, and J is diagonal with $r_k^2 J_{m_k}$ as k^{th} diagonal element.

5.1 PD Control Revisited

In the absence of gravity, a PD control for set-point tracking works for the full dynamics too! However, we show this using a Lyapunov function instead of the final value theorem. Let $u = -K_P(q - q_d) - K_D\dot{q}$. The Lyapunov function we use is

$$V = \frac{1}{2}\dot{q}^T M(q)\dot{q} + \frac{1}{2}(q - q_d)^T K_P (q - q_d) \tag{56}$$

$$
\begin{aligned}
\dot{V} &= \dot{q}^T M(q)\ddot{q} + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \dot{q}^T K_P(q - q_d) \\
&= \dot{q}^T \left(u - C(q,\dot{q}) + K_P(q - q_d)\right) + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} \\
&= \dot{q}^T \left(u + K_P(q - q_d)\right) \\
&= -\dot{q}^T K_D \dot{q}
\end{aligned}
\tag{57}
$$

Using La Salle's invariance principle, the fact that $\dot{V} \leq 0$ enables us to conclude that $\dot{V} \to 0$ so that $q \to q_d$ and $\dot{q} \to 0$.

La Salle's invariance principle states that for a candidate Lyapunov function $V(x)$ (continuous, positive definite) if $\dot{V} \leq 0$ then solutions $x(t)$ from all initial conditions will approach the largest set invariant set $M = \{x \in \mathbb{R}^n : V(t) \equiv 0\}$.

For (103) with a PD control feedback and Lyapunov function (56), $\dot{V} \equiv 0 \implies \dot{q} \equiv 0 \implies \ddot{q} \equiv 0 \implies u = -K_P(q - q_d) \equiv 0 \implies q \equiv q_d$.

5.2 Gravity Compensation

Since $g(q)$ depends on q and not $\dot{q}$, we may choose to include gravity compensation. Then, $u = -K_P(q - q_d) - K_D\dot{q} + g(q)$. It is straightforward to show that the PD control with gravity compensation will achieve zero errors in a set-point regulation task with no disturbances.

5.3 Inverse Dynamics Control

Consider

$$
M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = u, \tag{58}
$$

Let $u = M(q)a_q + C(q,\dot{q})\dot{q} + g(q)$. The closed loop simply becomes

$$
M(q)\ddot{q} = M(q)a_q \implies \ddot{q} = a_q \tag{59}
$$

We can choose a_q to be $a_q = \ddot{q}(t) + K_P(q_d(t) - q(t)) + K_D(\dot{q}_d(t) - \dot{q}(t))$. If $e(t) = q(t) - q_d(t)$, then $\ddot{e} + K_D\dot{e} + K_P e = 0$. It is straightforward to choose gains K_D and K_P so that $e(t) \to 0$ as $t \to \infty$.

5.4 Task Space Inverse Dynamics Control

Let X be the end-effector pose with orientation given by a minimal representation of $SO(3)$. Then,

$$
\ddot{X} = J_a(q)\ddot{q} + \dot{J}_a(q)\dot{q} \tag{60}
$$

If we choose

$$a_q = J_a(q)^{-1}\left(a_X - \dot{J}_a(q)\dot{q}\right) \tag{61}$$

then the joint space inverse dynamics control implies a task space dynamics of

$$\ddot{X} = a_X \tag{62}$$

and we can now track task space trajectories $X_d(t)$. The caveat is that $J_a(q)$ must be non-singular. If the task is not the full end-effector pose, but coordinates of smaller size, Jacobian pseudoinverses may be used.

5.5 Robust Inverse Dynamics Control

The issue with inverse dynamics control is that the guarantees assume perfect cancellation of nonlinear dynamics to obtain the linearized system. When the model is not perfectly known, we want our control performance to be robust to the errors, and perhaps adapt to them as well.

Consider the true system

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = u. \tag{63}$$

We design our control based on the assumed system Consider

$$\hat{M}(q)a_q + \hat{C}(q,\dot{q})\dot{q} + \hat{g}(q) = u. \tag{64}$$

We model the closed loop as

$$\ddot{q} = a_q + \eta(q,\dot{q},\ddot{q},a_q) \tag{65}$$

where

$$\begin{aligned} \eta(q,\ddot{q},a_q) &= M(q)^{-1}\left((M(q)-\hat{M}(q))a_q + (C(q,\dot{q})-\hat{C}(q,\dot{q}))\dot{q} + g(q) - g(q)\right) \\ &= M^{-1}\left(\tilde{M}a_q + \tilde{C}\dot{q} + \tilde{g}\right) \\ &= Ea_q + M^{-1}\left(\tilde{C}\dot{q} + \tilde{g}\right) \end{aligned} \tag{66}$$

Let $e = (\tilde{q}, \tilde{\dot{q}})$. Selecting $a_q = \ddot{q}_d(t) - K_0\tilde{q} - K_1\tilde{\dot{q}} + \delta a$, where the last term is to be designed, we get

$$\dot{e} = \begin{bmatrix} 0 & I \\ -K_0 & -K_1 \end{bmatrix} e + \begin{bmatrix} 0 \\ I \end{bmatrix}(\delta a + \eta) \tag{67}$$

Suppose we can bound η as $\|\eta\| \leq \rho(e,t)$, we can then design δa to guarantee ultimate boundedness of e.

Let $V = e^T P e$ where $A^T P + PA = -Q$. Since A can be made Hurwitz by choosing K_0 and K_1, we know that for each $Q > 0$ there exists $P > 0$ that satisfies the Lyapunov equation.

We have that

$$\begin{aligned} \dot{V} &= e^T PAe + e^T A^T Pe + 2e^T PB(\delta a + \eta) \\ &= -e^T Qe + 2e^T PB(\delta a + \eta) \end{aligned} \tag{68}$$

We choose

$$\delta a = \begin{cases} -\rho(e,t)\frac{B^T Pe}{\|B^T Pe\|} & , \quad \text{if } \|B^T Pe\| \neq 0 \\ 0 & , \quad \text{if } \|B^T Pe\| = 0 \end{cases} \tag{69}$$

Let $w = B^T Pe$. Then the second term in (68) is

$$\begin{aligned} w^T\left(-\rho\frac{w}{\|w\|} + \eta\right) &\leq -\rho\|w\| + \|w\|\|\eta\| && (w^T\eta \leq \|w\|\|\eta\|) \\ &\leq \|w\|(-\rho + \|\eta\|) \\ &\leq 0 && (\|\eta\| \leq \rho(e,t)) \end{aligned}$$

So, $\dot{V} \leq -e^T Qe < 0$.

All of this works if $\|\eta\| \leq \rho(e,t)$. To define such a bound, consider

$$\begin{aligned} \eta &= Ea_q + M^{-1}\left(\tilde{C}\dot{q} + \tilde{g}\right) \\ &= E\delta a + E(\ddot{q}_d(t) - K_0\tilde{q} - K_1\dot{\tilde{q}}) + M^{-1}\left(\tilde{C}\dot{q} + \tilde{g}\right) \\ &\implies \|\eta\| \leq \alpha\|\delta a\| + \gamma_1\|e\| + \gamma_2\|e\|^2 + \gamma_3 \end{aligned} \tag{70}$$

If we can find $\|E\| = \alpha < 1$, and constants γ_i above, we can define

$$\rho(e,t) = \frac{1}{1-\alpha}\left(\gamma_1\|e\| + \gamma_2\|e\|^2 + \gamma_3\right). \tag{71}$$

For $E = M^{-1}\tilde{M} = M^{-1}\hat{M} - 1$ we can always ensure $\alpha < 1$ by defining $\hat{M}$ as

$$\hat{M} = \frac{2}{\overline{M} + \underline{M}} I \tag{72}$$

where $\underline{M} \leq \|M^{-1}\| \leq \overline{M}$.

5.5.1 Continuous Robust Control

The robust controller (69) is discontinuous, but ensures that $e(t) \to 0$. We can use a continuous approximation at the cost of only being able to show that the errors are uniformly ultimately bounded (UUB). An open ball $B_r(y) \in \mathbb{R}^n$ is a set $\{x \in \mathbb{R}^n \colon \|x - y\| < r\}$.

A system is UUB with respect to ball $B_r(0)$ if for every initial condition the error $e(t)$ there exists $T < \infty$ such that $e(t) \in B_r(0)\ \forall t \geq T$. The trouble is this ultimate bound becomes large when $\rho(e,t)$ is large.

$$\delta a = \begin{cases} -\rho(e,t)\frac{B^T Pe}{\|B^T Pe\|} & , \quad \text{if } \|B^T Pe\| > \epsilon \\ -\frac{\rho(e,t)}{\epsilon} B^T Pe & , \quad \text{if } \|B^T Pe\| \leq \epsilon \end{cases} \tag{73}$$

So, $\dot{V} = -e^T Qe + 2w^T(\delta a + \eta)$. When $\|w\| \leq \epsilon$

$$\begin{aligned} \dot{V} &= -e^T Qe + 2w^T(\delta a + \eta) \\ &\leq -e^T Qe + 2w^T(-\frac{\rho}{\epsilon}w + \rho\frac{w}{\|w\|}) \\ &\leq -e^T Qe - 2\frac{\rho}{\epsilon}\|w\|^2 + 2\rho\|w\| \end{aligned} \tag{74}$$

which is clearly maximized at $\|w\| = \epsilon/2$

Thus

$$\dot{V} \leq -e^T Qe + \epsilon\frac{\rho}{2} \tag{75}$$

We want to find the smallest ball in error coordinates e outside of which $\dot{V} < 0$. Clearly, $\dot{V} < 0$ when $e^T Qe > \epsilon\rho/2$. Since $e^T Qe \geq \lambda_{min}(Q)\|e\|^2$, $\dot{V} \geq 0$ when $\lambda_{min}(Q)\|e\|^2 \geq \epsilon\rho/2$. So, $\dot{V} < 0$ outside of the set

$$\delta = \left(\frac{\epsilon\rho}{2\lambda_{min}(Q)}\right)^{1/2} \tag{76}$$

The UUB ball comes from the smallest ball containing the smallest level set that contains $B_\delta(0)$.

5.6 Adaptive Inverse Dynamics Control

The error in model estimate affects $\rho(\epsilon, t)$ which ruins the lowest achievable error. Ideally, we want smaller model errors to achieve lower error. Luckily, we can learn models on-the-fly using adaptive control theory.

The idea is to create a dynamical system whose state is the parameters we want to estimate. We feed it an input that makes the estimated parameters reach a set that permits the state errors to converge.

We want to control the system

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = Y(q,\dot{q},\ddot{q})\theta = u. \tag{77}$$

Choosing $u = Y(q,\dot{q},a_q)\hat{\theta}$, where $a_q = \ddot{q}_d(t) - K_0\tilde{q} - K_1\dot{\tilde{q}}$ we get

$$\ddot{\tilde{q}} + K_1\dot{\tilde{q}} + K_0\tilde{q} = M^{-1}Y(q,\dot{q},\ddot{q})\tilde{\theta} = \Phi\tilde{\theta}, \tag{78}$$

where $\tilde{\theta} = \hat{\theta} - \theta$.

We get the ODE

$$\dot{e} = Ae + B\Phi\tilde{\theta} \tag{79}$$

with same matrices as in the robust case.

How should we pick $\hat{\theta}$, given that we don't know θ? Consider a function of $e, \hat{\theta}, \theta$ given by

$$V = e^T Pe + \tilde{\theta}^T \Gamma \tilde{\theta}. \tag{80}$$

For $P > 0$ and $\Gamma > 0$, $V = 0$ when $e = 0$ and $\theta = \hat{\theta}$.

Let K_0 and K_1 be chosen so that A is Hurwitz. Implies there exists $Q > 0$ such that $A^T P + PA = -Q$. We have

$$\dot{V} = -e^T Qe + 2\tilde{\theta}^T \left(\Phi^T B^T Pe + \Gamma\dot{\hat{\theta}}\right) \tag{81}$$

If we knew θ we'd ensure that the second term was negative definite. However, since we don't, we set the second term to zero, by choosing

$$\dot{\hat{\theta}} = -\Gamma^{-1}\Phi^T B^T Pe \tag{82}$$

It's like a nonlinear integral control!

Analysis We have that $\dot{V}$ is non-positive and is the square of a term. Therefore $V(t) - V(t_0) \leq \int_0^T e^T(s)Qe(s)ds < \infty$. This makes $e(t)$ a square integrable function. Now, if we can show that its derivative $\dot{e}(t)$ is bounded, we can show that $e(t) \to 0$.

Lemma 1 (Barabalat)**.** *Suppose $f\colon \mathbb{R} \mapsto \mathbb{R}$ is a square integrable function and further suppose that its derivative $\dot{f}$ is bounded. Then $f(t) \to 0$ as $t \to \infty$.*

So, why is $\dot{e}$ bounded? Since $V(t)$ is bounded so are $e(t)$ and $\tilde{\theta}(t)$. Since $e(t)$ is bounded so are $\tilde{q}$ and $\dot{\tilde{q}}$. This implies that $\ddot{\tilde{q}}$ is bounded, so that $\dot{e}(t)$ is bounded. Requires bounded $\ddot{q}_d(t)$.

5.7 Passivity Based Control

We define a controller as

$$u = M(q)a + C(q, \dot{q})v + g(q) - Kr, \text{ where} \tag{83}$$

$$\begin{aligned} v &= \dot{q}_d - \Lambda\tilde{q} && (84)\\ a &= \dot{v} = \ddot{q}_d - \Lambda\dot{\tilde{q}} && (85)\\ r &= \dot{q} - v = \dot{\tilde{q}} + \Lambda\tilde{q}, && (86) \end{aligned}$$

with K, Λ diagonal matrices of positive gains.

The closed loop becomes

$$M(q)\dot{r} + C(q, \dot{q})r + Kr = 0 \tag{87}$$

$$V = \frac{1}{2}r^T M(q)r + \tilde{q}^T \Lambda K\tilde{q} \tag{88}$$

$$\begin{aligned} \dot{V} &= -r^T Kr + 2\tilde{q}^T \Lambda K\dot{\tilde{q}} + \frac{1}{2}r^T \left(\dot{M}(q) - 2C\right) r && (89)\\ &= -\tilde{q}^T \Lambda^T K\Lambda\tilde{q} - \dot{\tilde{q}}^T K\dot{\tilde{q}} \\ &= -e^T Qe \end{aligned}$$

If $M(q)$ is bounded then we can conclude that $e = 0 \iff V = 0$ so that the origin is GAS.

This seems to suggest passivity-based control has no advantage over inverse dynamics control. The advantage appears in the robust and adaptive control approaches, where the constraints on $M(q)$ are relaxed. Robust control required bounds on the error and mass matrix terms. Implementing the adaptive controller requires knowledge of acceleration (due to Φ) and invertible $M(q)$.

5.7.1 Robust Control

The control is

$$u = \hat{M}(q)a + \hat{C}(q, \dot{q})v + \hat{g}(q) - Kr = Y(q, \dot{q}, a, v)\hat{\theta} - Kr \tag{90}$$

The closed-loop becomes

$$M(q)r + C(q, \dot{q})r + Kr = Y(\hat{\theta} - \theta) \tag{91}$$

Let $\hat{\theta} = \theta_0 + \delta\theta$ where $\|\tilde{\theta}\| = \|\theta - \theta_0\| \leq \rho$. The same Lyapunov candidate as in for passivity gives

$$\dot{V} = -e^T Q e + r^T Y(\delta\theta + \tilde{\theta}) \tag{92}$$

The same UUB analysis goes through where $w = r^T Y$, $\delta\theta = \delta a$ and $\tilde{\theta} = \eta$. Note that the uncertainty characterization is simpler, and prior knowledge is easily baked in.

5.7.2 Adaptive Control

The closed-loop is again

$$M(q)r + C(q,\dot{q})r + Kr = Y(\hat{\theta} - \theta) \tag{93}$$

The Lyapunov function is

$$V = \frac{1}{2} r^T M(q) r + \tilde{q}^T \Lambda K \tilde{q} + \frac{1}{2}\tilde{\theta}^T \Gamma \tilde{\theta} \tag{94}$$

and the update law becomes

$$\dot{\hat{\theta}} = -\Gamma^{-1} Y(q,\dot{q},a,v) r \tag{95}$$

Again we see that

$$\dot{V} = -\tilde{q}^T \Lambda^T K \Lambda \tilde{q} - \dot{\tilde{q}}^T K \dot{\tilde{q}} - \tilde{\theta}^T (Y^T r + \Gamma \dot{\hat{\theta}}) \tag{96}$$

As in the previous adaptive control analysis, we use Barbalat's Lemma to conclude $e(t) \to 0$ and $\|\tilde{\theta}\|$ remains bounded.

6 Force Control

The robot control tasks we have focused on involve a desired, possibly time-varying, position of the end effector. This end-effector trajectory (task-space trajectory) dictates a trajectory $q(t)$ for the robot joint coordinates q. Many tasks are sufficiently characterized by the position of the end-effector.

In some cases, the task for the robot involves generating desired forces rather than just positions. Simple examples involve pushing delicate objects on a table-top, polishing or grinding, assembly tasks, throwing a ball.

Force Sensors There are typically three locations for placing sensors that measure the forces acting on the robot. They are the wrist, the joints, and the end-effector. The wrist sensor is usually a force-and torque sensor placed between the end-effector and the final robot link. A force sensor measures the torques about the actuator shaft. The end-effector sensors are often tactile sensors placed on the fingers of grippers.

6.1 Coordinate Frames and Constraints

The forces acting on a robot often come about due to contact with the environment. This contact occurs at specific surfaces and points, and represent position constraints. The forces acting on the robot then arise as reactions to the existence of these constraints. For example, your finger moving in free space would not sense any pressure at the finger tip, until it is pressed against a surface. Even when you sense a pressure due to a force, the magnitude depends on whether the surface is pushing on you, or whether you are pushing on the object, versus maintaining a light stationary contact.

6.2 Reciprocal Bases

In more formal descriptions of mechanics, the linear and angular velocity $\xi = (v, \omega)$ and force and moment $F = (f, n)$ are considered dual to each other. The quantities ξ and F are called twists and wrenches respectively, and are both six-dimensional.

Given a configuration space corresponding to a mechanical system, twists belong to the tangent space $\mathcal{M}$ and wrenches belong to the co-tangent space $\mathcal{F}$, and their product corresponds to power. The numerical value of the power depends on the bases we choose for $\mathcal{M}$ and $\mathcal{F}$. For consistency, the power we calculate must be the same for any pair of bases we choose, if these two pairs of bases are linearly related. This consistency is achieved by using reciprocal bases.

Definition 5 (Reciprocal Bases)**.** If $\{e_1, \dots, e_6\}$ is a basis for $\mathcal{M}$ and $\{f_1, \dots, f_6\}$ is a basis for $\mathcal{F}$, these two bases are *reciprocal* if

$$e_i^T f_j = \begin{cases} 1 & , \quad \text{if } i = j \\ 0 & , \quad \text{if } i \neq j. \end{cases} \tag{97}$$

Definition 6. A twist $\xi \in \mathcal{M}$ and wrench $\mathbb{F} \in \mathcal{F}$ are called $r$eciprocal if

$$\xi^T F = v^T f + \omega^T n = 0 \tag{98}$$

The advantage of using reciprocal bases for $\mathcal{M}$ and $\mathcal{F}$ is that the product $\xi^T F$ has the invariance we want. Therefore, the reciprocity condition (6) is invariant with respect to the choice of reciprocal bases.

6.3 Natural and Artificial Constraints

We discuss natural constraints which are defined using (6). The intuition is that the power consumed by a twist and wrench to satisfy a natural constraint is zero. A natural constraint typically comes from the environment the robot is interacting with. The total power consumed by such a twist and wrench may be non-zero, but none of that power consumption is due to the natural constraint.

Natural constraints in turn define artificial constraints, which arise due to specifying reference values for input motion and force control tasks. These are constraints we impose on the motion to complete a given task. The natural constraints together with the artificial constraint provide a complete reference for ξ and F.

Compliance frame A compliance frame $o_c x_c y_c z_c$ (also called a constraint frame) is a frame in which the task is easily described. For example, consider

1. inserting a peg into a hole, or
2. turning a crank.

Peg We choose an orthonormal bases for both $\mathcal{M}$ and $\mathcal{F}$, with respect to a compliant frame with origin at the bottom and center of peg, z parallel to peg axis.

7 Network Models and Impedance

Introduction The reciprocity condition $\xi^T F = 0$ mean that the forces of constraint do no work in directions of motion compatible with motion constraints. This property holds under ideal conditions of no friction and perfectly rigid robot and environment. In practice, compliance and friction alter the nature of constraints and constraint forces.

For example, when a robot pushes against a compliant surface, the contact experiences both non-zero normal reaction forces and non-zero motion, so that the work $\xi^T F$ is non-zero. If the stiffness of the surface is k, then the force is kx, and the total work done is

$$\begin{aligned} W &= \int_0^t \dot{x}(u) k x(u) du \\ &= \int_0^t \frac{d}{du} \frac{1}{2} k x(u)^2 du \\ &= \frac{1}{2} k (x(t)^2 - x(0)^2) \end{aligned} \tag{99}$$

The work done increases with k, as does the force required to induce a velocity $\dot{x}$. This impact of stiffness on the relationship between force, velocity and energy is captured by the more general notion of impedance.

7.1 One-Port Model

The robot and environment are modeled as one-port nodes in a network (See main text for details). Each node has two interaction ports and corresponding port variables **effort** F_i and **flow** V_i. The relationship between these port variables depend on the dynamics of the system.

The interaction between the robot and environment is then modeled as a network of such one-ports, where the connections between nodes occur at the interaction ports of the nodes, and flow and effort are transmitted between interacting nodes. For a network with a robot and environment, the flow and effort variables are V_r, V_e and F_r, F_e respectively. The power consumed or dissipated by the network is $V^T F$.

7.2 Impedance

The relationship between flow and effort for a system is given by the impedance operator of that system. For linear systems, we have the definition below

Definition 7. For a one-port network, the impedance $Z(s)$ is

$$Z(s) = \frac{F(s)}{V(s)}.$$

Example 1. For a S-M-D with dynamics $M\ddot{x} + B\dot{x} + Kx = F$, we have

$$Z(s) = Ms + B + \frac{K}{s}. \tag{100}$$

The task space inverse dynamics control approach allows us to think of impedance of a robot in terms of (100), even though the relationship between velocity and force for the full model (103) is quite complex.

7.2.1 Classification of Impedance Operators

1. Inertial: iff $|Z(0)| = 0$
2. Resistive: iff $0 < |Z(0)| < \infty$
3. Capacitive: iff $|Z(0)| = \infty$

See (100) for example of each kind (mass, damper, spring, respectively).

The impedance of the robot determines the force response of the end-effector to a velocity input at the end-effector. That is,

$$F(s) = Z(s)V(s). \tag{101}$$

Suppose we move the end-effector with a constant reference.

$$F(s) = MsV_r\frac{1}{s} \tag{102}$$

7.2.2 Admittance

The reciprocal relationship is often called admittance $Y(s)$

$$Y(s) = \frac{V(s)}{F(s)}.$$

7.2.3 Thevenin and Norton Equivalents

We can represent any one-port network consisting of multiple nodes as an equivalent network containing just one impedance and a source. In a Thevenin equivalent network, the impedance $Z(s)$ is placed in series with a source of effort F_s. In a Norton equivalent network, the impedance $Z(s)$ is placed in parallel to a source of flow V_s. These sources represent references (see artificial constraints) or external disturbances.

7.3 Task Space Dynamics

The interaction between manipulator and environment is easier to describe in the task space rather than the joint space. Given a system

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + B\dot{q} + g(q) + J^T(q)F_e = u, \tag{103}$$

we can impose the control

$$u = M(q)a_q + C(q,\dot{q})\dot{q} + B\dot{q} + g(q) + J^T(q)a_f, \tag{104}$$

to obtain the closed loop

$$\ddot{q} = a_q + M(q)^{-1}J^T(q)(F_e - a_f). \tag{105}$$

We can choose a_q, a desired acceleration in the joint space, as

$$a_q = J_a(q)^{-1}\left(a_X - \dot{J}_a(q)\dot{q}\right). \tag{106}$$

where a_X is the desired acceleration in the task space X e. Since

$$\ddot{X} = J(q)\ddot{q} + \dot{J}(q)\dot{q}, \tag{107}$$

the closed-loop task space dynamics are

$$\ddot{X} = a_X + W(q)(F_e - a_f), \tag{108}$$

where $W(q) = J(q)M^{-1}(q)J(q)$ is the mobility tensor.

If $W(q)$ is non-singular, then we set $a_f = F_e$, and achieve any force regulation task by modifying a_X appropriately. Therefore, we design impedance controllers for the system

$$\ddot{X} = a_X. \tag{109}$$

7.4 Impedance Control

When a robot interacts with physical objects, defining the interaction by impedance of the robot is often easier than specifying the exact motion of the robot end-effector relative to the environment.

Example 2 (Apparent Inertia)**.** we can control a system $M\ddot{x} = u - F$ so that it appears to us as a lighter mass $m' < M$ by using the control $u = -mF$, $m > 0$. The new apparent mass is $m' = \frac{M}{1+m}$.

In general, suppose we want the system (103) to behave like a system with inertia, damping, and stiffness in the **task space** given by

$$M_d\ddot{x} + B_d\dot{x} + K_d x = F. \tag{110}$$

We would need to implement a task space inverse dynamics control, and choose a_X as

$$a_x = \ddot{x} - M_d^{-1}(B_d\dot{x} + K_d x + F) \tag{111}$$

Note that the term F corresponds to the earlier point of setting $a_f = F_e$ and using a_X to implement any force feedback.

Our closed-loop model (110) is a spring-mass-damper model with desired parameters. When there is no external force ($F = 0$), clearly $x \to 0$ for $M_d, B_d, K_d > 0$. Instead, we can derive an impedance control relative to a trajectory $x^d(t)$, so that our impedance control achieves trajectory tracking, however the reaction of the system to disturbance forces F behaves like our target system.

7.5 Hybrid Impedance Control

When we fix the robot impedance, the velocity or force at the contact depends on what the environment does. Instead, we'd like to fix one of the robot's force or velocity to a reference value.

To achieve this goal, we need to exploit the environment's characteristics when possible.

The main achievement of impedance control is that we do not have to know anything about the environment beyond one simple thing: is $Z_e(0) = 0$ or is $Z_e(0) = \infty$? That knowledge is enough to design a controller that achieved our goals (position or force-tracking) no matter what the interaction force happens to be.