Interaction Control

- Manipulation requires interaction
  - object behavior affects control of force and motion
- Independent control of force *and* motion is not possible
  - object behavior relates force and motion
    - contact a rigid surface: *kinematic* constraint
    - move an object: *dynamic* constraint
- Accurate control of force *or* motion requires detailed models of
    - manipulator dynamics
    - object dynamics
  - object dynamics are usually known poorly, often not at all

Object Behavior

- Can object forces be treated as external (exogenous) disturbances?
  - the usual assumptions don't apply:
    - "disturbance" forces depend on manipulator state
    - forces often aren't small by any reasonable measure
- Can forces due to object behavior be treated as modeling uncertainties?
  - yes (to some extent) but the usual assumptions don't apply:
    - command and disturbance frequencies overlap
- Example: two people shaking hands
  - how each person moves influences the forces evoked
    - "disturbance" forces are state-dependent
  - each may exert comparable forces and move at comparable speeds
    - command & "disturbance" have comparable magnitude & frequency

Alternative: control *port behavior*

- Port behavior:
  - system properties and/or behaviors "seen" at an interaction port
- Interaction port:
  - characterized by conjugate variables that define power flow

$$\begin{cases} \text{power in} \qquad P = \mathbf{e}^{\mathbf{t}}\mathbf{f} \\ \mathbf{e} = \left[e_1 \cdots e_n\right]^t \quad \text{efforts (forces)} \\ \mathbf{f} = \left[f_1 \cdots f_n\right]^t \quad \text{flows (velocities)} \end{cases}$$

- Key point:

port behavior is unaffected by contact and interaction

Impedance & Admittance

- Impedance and admittance characterize interaction
  - a dynamic generalization of resistance and conductance
- Usually introduced for linear systems but generalizes to nonlinear systems
  - state-determined representation:
  - this form may be derived from or depicted as a network model

electrical capacitor $Z(s) = \dfrac{e(s)}{i(s)} = \dfrac{1}{Cs}$

electrical inductor $Z(s) = \dfrac{e(s)}{i(s)} = L(s)$

$$\begin{cases} \dot{z} = Z_s(z,V) & \text{State equations} \\ F = Z_o(z,V) & \text{Output equations} \\ P = F^t V & \text{Constraint on input \& output} \\ z \in \Re^n, F \in \Re^m, V \in \Re^m, P \in \Re \end{cases}$$

nonlinear 1D elastic element (spring)

$$\dot{x} = v$$
$$f = \Phi(x)$$

Impedance & Admittance (continued)

- Admittance is the causal dual of impedance
  - Admittance: flow out, effort in
  - Impedance: effort out, flow in
- Linear system: admittance is the inverse of impedance
- Nonlinear system:
  - causal dual is well-defined:
  - but may not correspond to any impedance
    - inverse may not exist

$$Y(s) = Z(s)^{-1}$$

electrical capacitor

$$Y(s) = \frac{i(s)}{e(s)} = Cs$$

$$\begin{cases} \dot{y} = Y_s(y, F) \\ V = Y_o(y, F) \\ P = F^t V \\ y \in \Re^n, F \in \Re^m, V \in \Re^m, P \in \Re \end{cases}$$

nonlinear 1D inertial element (mass)

$$\dot{p} = f$$

$$v = \Psi(p)$$

Impedance as dynamic stiffness

- Impedance is also loosely defined as a dynamic generalization of stiffness
  - effort out, displacement in
- Most useful for mechanical systems
  - displacement (or generalized position) plays a key role

$$\begin{cases} \dot{z} = Z_s(z, X) \\ F = Z_o(z, X) \\ dW = F^t dX \\ z \in \Re^n, F \in \Re^m, X \in \Re^m, P \in \Re \end{cases}$$

Interaction control: causal considerations

- What's the best input/output form for the manipulator?
- The set of objects likely to be manipulated includes
  - inertias
    - minimal model of most movable objects
  - kinematic constraints
    - simplest description of surface contact
- Causal considerations:
  - inertias *prefer* admittance causality
  - constraints *require* admittance causality
  - compatible manipulator behavior should be an impedance
- An ideal controller should make the manipulator behave as an impedance
- Hence impedance control
  - Hogan 1979, 1980, 1985, etc.

Robot Impedance Control

- Works well for interaction tasks:
  - Automotive assembly
    - (Case Western Reserve University, US)
  - Food packaging
    - (Technical University Delft, NL)
  - Hazardous material handling
    - (Oak Ridge National Labs, US)
  - Automated excavation
    - (University of Sydney, Australia)
  - … and many more

- Facilitates multi-robot / multi-limb coordination
    - Schneider et al., Stanford
- Enables physical cooperation of robots and humans
    - Kosuge et al., Japan
    - Hogan et al., MIT

OSCAR the robot

Photograph removed due to copyright restrictions.

E.D.Fasse & J.F.Broenink, U. Twente, NL

Network modeling perspective on interaction control

- Port concept
  - control interaction port behavior
  - port behavior is unaffected by contact and interaction
- Causal analysis
  - impedance and admittance characterize interaction
  - object is likely an admittance
  - control manipulator impedance
- Model structure
  - structure is important
  - power sources are commonly modeled as equivalent networks
    - Thévenin equivalent
    - Norton equivalent
- Can equivalent network structure be applied to interaction control?

Equivalent networks

- Initially applied to networks of static linear elements
    - Sources & linear resistors
  - Thévenin equivalent network
      - M. L. Thévenin, *Sur un nouveau théorème d'électricité dynamique.* Académie des Sciences, Comptes Rendus 1883, 97:159-161
    - Thévenin equivalent source—power supply or transfer
    - Thévenin equivalent impedance—interaction
    - Connection—series / common current / 1-junction
  - Norton equivalent network is the causal dual form
- Subsequently applied to networks of dynamic linear elements
    - Sources & (linear) resistors, capacitors, inductors

Nonlinear equivalent networks

- Can equivalent networks be defined for nonlinear systems?
  - Nonlinear impedance and admittance can be defined as above
  - Thévenin & Norton sources can also be defined
    - Hogan, N. (1985) *Impedance Control: An Approach to Manipulation.* ASME J. Dynamic Systems Measurement & Control, Vol. 107, pp. 1-24.
- However…
  - In general the junction structure cannot
- In other words:
  - separating the pieces is always possible
  - re-assembling them by superposition is not

Nonlinear equivalent network for interaction control

- One way to preserve the junction structure:
  - specify an equivalent network structure in the (desired) interaction behavior
  - provides key superposition properties
- Specifically:
  - *nodic* desired impedance
    - does not require inertial reference frame
  - "virtual" trajectory
    - "virtual" as it need not be a realizable trajectory

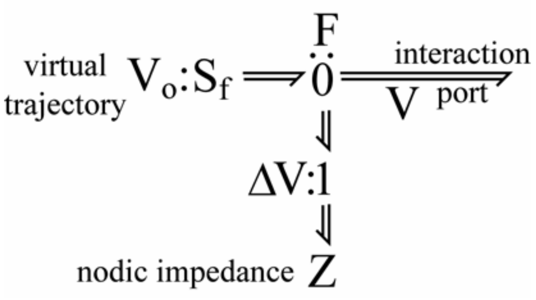
$\mathbf{V}_0 = \mathbf{V}_0 : \{c\}$ virtual trajectory

$\Delta\mathbf{V} = \mathbf{V}_0 - \mathbf{V}$

network junction structure (0 junction)

$$\left.\begin{aligned} \dot{\mathbf{z}} &= \mathbf{Z}_s(\mathbf{z}, \Delta\mathbf{V}) : \{c\} \\ \mathbf{F} &= \mathbf{Z}_o(\mathbf{z}, \Delta\mathbf{V}) : \{c\} \end{aligned}\right\}$$ nodic impedance

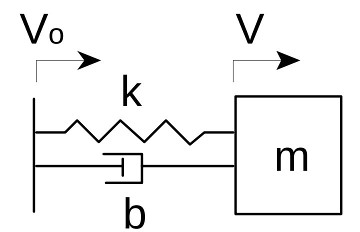
$: \{c\}$ denotes modulation by control inputs

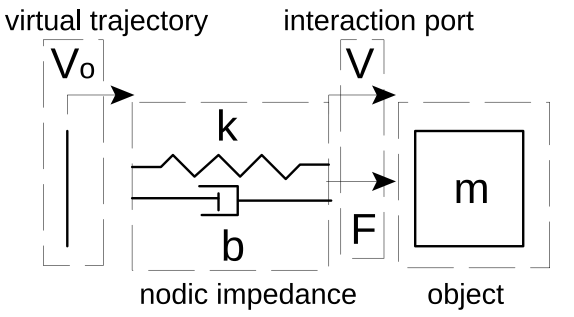


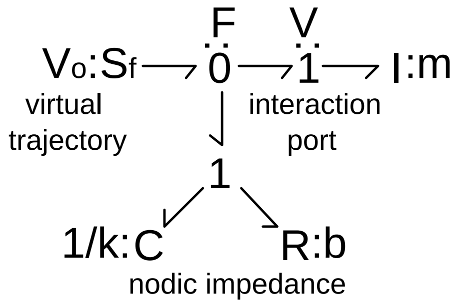
Norton equivalent network

Virtual trajectory

- Nodic impedance:
  – Defines desired interaction dynamics
  – Nodic because input velocity is defined relative to a “virtual” trajectory
- Virtual trajectory:
  – like a motion controller’s reference or nominal trajectory *but* no assumption that dynamics are fast compared to motion
  – “virtual” because it need not be realizable
    - e.g., need not be confined to manipulator’s workspace







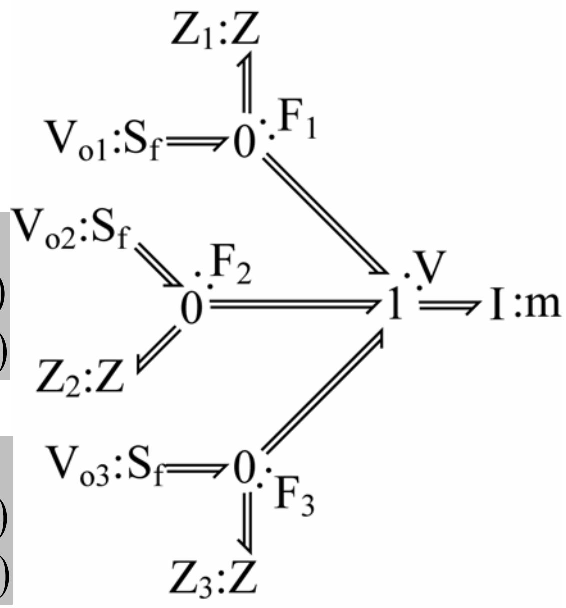
Superposition of “impedance forces”

- Minimal object model is an inertia
  - it responds to the sum of input forces
  - in network terms: it comes with an associated 1-junction
- This guarantees *linear* summation of component impedances…
- …even if the component impedances are *nonlinear*

$$\Delta\mathbf{V}_1 = \mathbf{V}_{o1} - \mathbf{V}$$
$$\dot{\mathbf{z}}_1 = \mathbf{Z}_{s1}(\mathbf{z}_1, \Delta\mathbf{V}_1)$$
$$\mathbf{F}_1 = \mathbf{Z}_{o1}(\mathbf{z}_1, \Delta\mathbf{V}_1)$$

$$\Delta\mathbf{V}_2 = \mathbf{V}_{o2} - \mathbf{V}$$
$$\dot{\mathbf{z}}_2 = \mathbf{Z}_{s2}(\mathbf{z}_2, \Delta\mathbf{V}_2)$$
$$\mathbf{F}_2 = \mathbf{Z}_{o2}(\mathbf{z}_2, \Delta\mathbf{V}_2)$$

$$\Delta\mathbf{V}_3 = \mathbf{V}_{o3} - \mathbf{V}$$
$$\dot{\mathbf{z}}_3 = \mathbf{Z}_{s3}(\mathbf{z}_3, \Delta\mathbf{V}_3)$$
$$\mathbf{F}_3 = \mathbf{Z}_{o3}(\mathbf{z}_3, \Delta\mathbf{V}_3)$$



$$\dot{\mathbf{V}} = \mathbf{m}^{-1}(\mathbf{F}_1 + \mathbf{F}_2 + \mathbf{F}_3)$$

One application: collision avoidance

- Impedance control also enables *non*-contact (virtual) interaction
  - Impedance component to acquire target:
    - Attractive force field (potential “valley”)
  - Impedance component to prevent unwanted collision:
    - Repulsive force-fields (potential “hills”)
    - One per object (or part thereof)
  - Total impedance is the sum of these components
    - Simultaneously acquires target while preventing collisions
  - Works for *moving* objects and targets
    - Update their location by feedback to the (nonlinear) controller
  - Computationally simple
    - Initial implementation used 8-bit Z80 processors
    - Andrews & Hogan, 1983

Andrews, J. R. and Hogan, N. (1983) *Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator*, pp. 243-251 in D. Hardt and W.J. Book, (eds.), Control of Manufacturing Processes and Robotic Systems, ASME.

High-speed collision avoidance

- Static protective (repulsive) fields must extend beyond object boundaries
  - may slow the robot unnecessarily
  - may occlude physically feasible paths
  - especially problematical if robot links are protected
- Solution: *time-varying* impedance components
  - protective (repulsive) fields grow as robot speeds up, shrink as it slows down
  - Fields shaped to yield maximum acceleration or deceleration
    - Newman & Hogan, 1987
    - See also extensive work by Khatib et al., Stanford

Newman, W. S. and Hogan, N. (1987) *High Speed Robot Control and Obstacle Avoidance Using Dynamic Potential Functions*, proc. IEEE Int. Conf. Robotics & Automation, Vol. 1, pp. 14-24.

Impedance Control Implementation

- Controlling robot impedance is an ideal
  - like most control system goals it may be difficult to attain
- How do you control impedance or admittance?
- One primitive but highly successful approach:
  - Design low-impedance hardware
    - Low-friction mechanism
      - Kinematic chain of rigid links
    - Torque-controlled actuators
      - e.g., permanent-magnet DC motors
      - high-bandwidth current-controlled amplifiers
  - Use feedback to increase output impedance
    - (Nonlinear) position and velocity feedback control
- “Simple” impedance control

Robot Model

- Effort-driven inertia

$$\mathbf{I}(\boldsymbol{\theta})\dot{\boldsymbol{\omega}} + \mathbf{C}(\boldsymbol{\theta},\boldsymbol{\omega}) + \mathbf{G}(\boldsymbol{\theta}) = \boldsymbol{\tau}_{motor} + \boldsymbol{\tau}_{interaction}$$

θ: generalized coordinates, joint angles, configuration variables

ω: generalized velocities, joint angular velocities

τ: generalized forces, joint torques

I: configuration-dependent inertia

C: inertial coupling (Coriolis & centrifugal accelerations)

G: potential forces (gravitational torques)

- Linkage kinematics transform interaction forces to interaction torques

$$\mathbf{X} = \mathbf{L}(\boldsymbol{\theta})$$

$$\mathbf{V} = \dot{\mathbf{X}} = (\partial\mathbf{L}/\partial\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega}$$

$$\boldsymbol{\tau}_{interaction} = \mathbf{J}(\boldsymbol{\theta})^t\,\mathbf{F}_{interaction}$$

X: interaction port (end-point) position

V: interaction port (end-point) velocity

$\mathbf{F}_{interaction}$: interaction port force

L: mechanism kinematic equations

J: mechanism Jacobian

Simple Impedance Control

- Target end-point behavior
  - Norton equivalent network with elastic and viscous impedance, possibly nonlinear
- Express as equivalent (joint-space) configuration-space behavior
  - use kinematic transformations
- This defines a position-and-velocity-feedback controller…
  - A (non-linear) variant of PD (proportional+derivative) control
- …that will implement the target behavior

$$\mathbf{F}_{impedance} = \mathbf{K}(\mathbf{X}_o - \mathbf{X}) + \mathbf{B}(\mathbf{V}_o - \mathbf{V})$$

$\mathbf{X}_o$: virtual position

$\mathbf{V}_o$: virtual velocity

K: displacement-dependent (elastic) force function

B: velocity-dependent force function

$$\boldsymbol{\tau}_{motor} = \mathbf{J}(\boldsymbol{\theta})^t \mathbf{F}_{impedance}$$

$$\boldsymbol{\tau}_{motor} = \mathbf{J}(\boldsymbol{\theta})^t \left(\mathbf{K}(\mathbf{X}_o - \mathbf{L}(\boldsymbol{\theta})) + \mathbf{B}(\mathbf{V}_o - \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega})\right)$$

Dynamics of controller impedance coupled to mechanism inertia with interaction port:

$$\mathbf{I}(\boldsymbol{\theta})\dot{\boldsymbol{\omega}} + \mathbf{C}(\boldsymbol{\theta},\boldsymbol{\omega}) + \mathbf{G}(\boldsymbol{\theta}) =$$
$$\mathbf{J}(\boldsymbol{\theta})^t \left(\mathbf{K}(\mathbf{X}_o - \mathbf{L}(\boldsymbol{\theta})) + \mathbf{B}(\mathbf{V}_o - \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega})\right)$$
$$+ \mathbf{J}(\boldsymbol{\theta})^t \mathbf{F}_{interaction}$$

Mechanism singularities

- Impedance control also facilitates interaction with the robot's own mechanics
  - Compare with motion control:
- Position control maps desired end-point trajectory onto configuration space (joint space)
  - Requires inverse kinematic equations
    - Ill-defined, no general algebraic solution exists
      - one end-point position usually corresponds to many configurations
      - some end-point positions may not be reachable
- Resolved-rate motion control uses inverse Jacobian
  - Locally linear approach, will find a solution if one exists
  - At some configurations Jacobian becomes singular
    - Motion is not possible in one or more directions
- A typical motion controller won't work at or near these singular configurations

$$\mathbf{X} = \mathbf{L}(\boldsymbol{\theta})$$

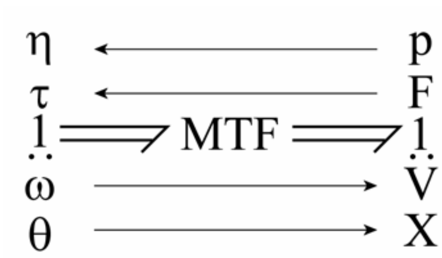
$$\boldsymbol{\theta}_{desired} = \mathbf{L}^{-1}(\mathbf{X}_{desired})$$

$$\mathbf{V} = \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega}$$

$$\boldsymbol{\omega}_{desired} = \mathbf{J}(\boldsymbol{\theta})^{-1}\mathbf{V}_{desired}$$

Mechanism junction structure

- Mechanism kinematics relate configuration space {**θ**} to workspace {**X**}
  - In network terms this defines a multiport modulated transformer
  - Hence power conjugate variables are well-defined in *opposite* directions



- Generalized coordinates uniquely define mechanism configuration
  - *By definition*
- Hence the following maps are *always* well-defined
  - generalized coordinates (configuration space) to end-point coordinates (workspace)
  - generalized velocities to workspace velocity
  - workspace force to generalized force
  - workspace momentum to generalized momentum

Control at mechanism singularities

- Simple impedance control law was derived by transforming desired behavior…
  - Norton equivalent network in workspace coordinates

  …from workspace to configuration (joint) space
- All of the required transformations are *guaranteed* well-defined at *all* configurations
  - $\mathbf{X} \Leftarrow \boldsymbol{\theta}$
  - $\mathbf{V} \Leftarrow \boldsymbol{\omega}$
  - $\boldsymbol{\tau} \Leftarrow \mathbf{F}$

$$\boldsymbol{\tau}_{motor} = \mathbf{J}(\boldsymbol{\theta})^t \left(\mathbf{K}\left(\mathbf{X}_o - \mathbf{L}(\boldsymbol{\theta})\right) + \mathbf{B}\left(\mathbf{V}_o - \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega}\right)\right)$$

- Hence the simple impedance controller can operate *near, at and through* mechanism singularities

Generalized coordinates

- Aside:
  - Identification of generalized coordinates requires care
    - Independently variable
    - Uniquely define mechanism configuration
    - Not themselves unique
  - Actuator coordinates are often suitable, but not always
    - Example: Stewart platform
  - Identification of generalized forces also requires care
    - Power conjugates to generalized velocities
    - $P = \boldsymbol{\tau}^t \boldsymbol{\omega}$ or $\mathrm{d}W = \boldsymbol{\tau}^t \mathrm{d}\boldsymbol{\theta}$
  - Actuator forces are often suitable, not always

Inverse kinematics

- Generally a tough computational problem
- Modeling & simulation afford simple, effective solutions
  - Assume a simple impedance controller
  - Apply it to a simulated mechanism with simplified dynamics
  - Guaranteed convergence properties
    - Hogan 1984
    - Slotine &Yoerger 1987

Hogan, N. (1984) *Some Computational Problems Simplified by Impedance Control*, proc. ASME Conf. on Computers in Engineering, pp. 203-209.

Slotine, J.-J.E., Yoerger, D.R. (1987) *A Rule-Based Inverse Kinematics Algorithm for Redundant Manipulators* Int. J. Robotics & Automation 2(2):86-89

- Same approach works for redundant mechanisms
  - Redundant: more generalized coordinates than workspace coordinates
  - Inverse kinematics is fundamentally "ill-posed"
  - Rate control based on Moore-Penrose pseudo-inverse suffers "drift"
  - Proper analysis of effective stiffness eliminates drift
    - Mussa-Ivaldi & Hogan 1991

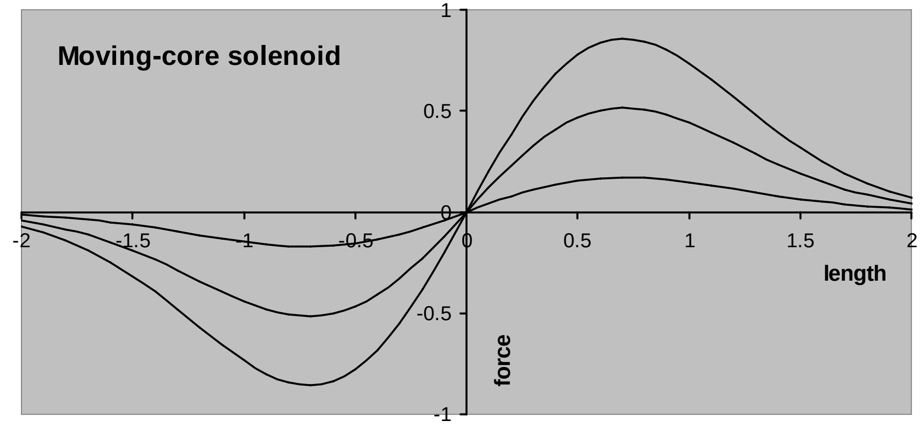
Mussa-Ivaldi, F. A. and Hogan, N. (1991) *Integrable Solutions of Kinematic Redundancy via Impedance Control.* Int. J. Robotics Research, 10(5):481-491

Intrinsically variable impedance

- Feedback control of impedance suffers inevitable imperfections
  - "parasitic" sensor & actuator dynamics
  - communication & computation delays
- Alternative: control impedance using intrinsic properties of the actuators and/or mechanism
  - Stiffness
  - Damping
  - Inertia

Intrinsically variable stiffness

- Engineering approaches
  - Moving-core solenoid
  - Separately-excited DC machine
    - Fasse et al. 1994
  - Variable-pressure air cylinder
  - Pneumatic tension actuator
    - McKibben "muscle"
  - …and many more
- Mammalian muscle
  - antagonist co-contraction increases stiffness & damping
  - complex underlying physics
    - see 2.183
  - increased stiffness requires increased force



Fasse, E. D., Hogan, N., Gomez, S. R., and Mehta, N. R. (1994) *A Novel Variable Mechanical-Impedance Electromechanical Actuator*. Proc. Symp. Haptic Interfaces for Virtual Environment and Teleoperator Systems, ASME DSC-Vol. 55-1, pp. 311-318.

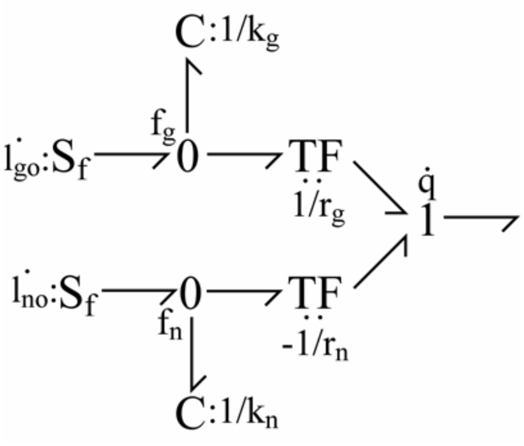
Opposing actuators at a joint

- Assume
  - constant moment arms
  - linear force-length relation
    - (grossly) simplified model of antagonist muscles about a joint

f: force; l: length; k: actuator stiffness

q: joint angle; t: torque; K: joint stiffness

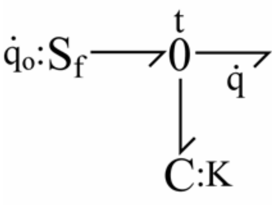
subscripts: g: agonist; n: antagonist, o: virtual



$$l_g = l_{go} - r_g q$$
$$l_n = l_{no} + r_n q$$
$$f_g = k_g l_g$$
$$f_n = k_n l_n$$
$$t = r_g f_g - r_n f_n = r_g k_g (l_{go} - r_g q) - r_n k_n (l_{no} + r_n q)$$
$$t = (r_g k_g l_{go} - r_n k_n l_{no}) - (r_g^2 k_g + r_n^2 k_n) q$$

- Equivalent behavior:
- Opposing torques subtract
- Opposing impedances add
  - Joint stiffness positive if actuator stiffness positive

$$t = K(q_o - q)$$
$$q_o = (r_g k_g l_{go} - r_n k_n l_{no})$$
$$K = (r_g^2 k_g + r_n^2 k_n)$$

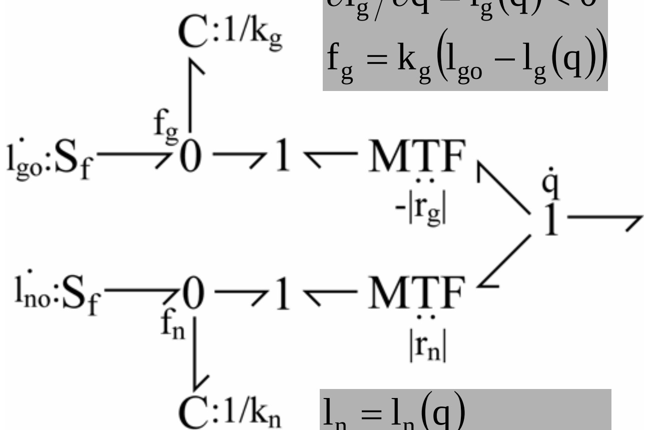


Configuration-dependent moment arms

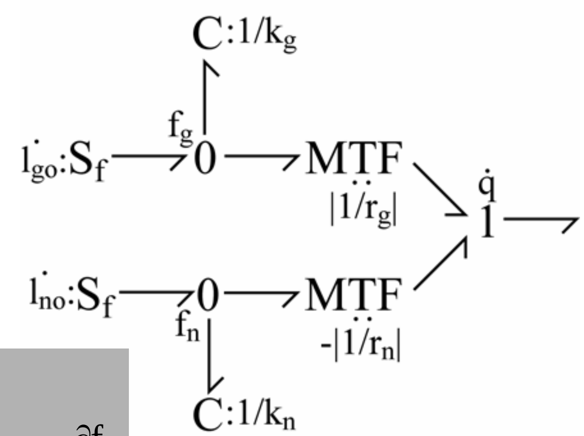
- Connection of linear actuators usually makes moment arm vary with configuration
- Joint stiffness, K:
  - Second term always positive
  - First term may be *negative*

$$l_g = l_g(q)$$
$$\partial l_g / \partial q = r_g(q) < 0$$
$$f_g = k_g\big(l_{go} - l_g(q)\big)$$

$$K = -\left(\frac{\partial r_g}{\partial q} f_g + \frac{\partial r_n}{\partial q} f_n\right) + \left(r_g^2 k_g + r_n^2 k_n\right)$$



More typical: change signs on the transformers

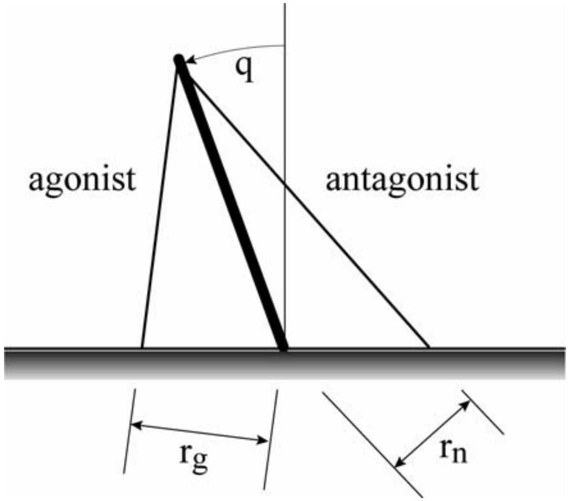


$$l_n = l_n(q)$$
$$\partial l_n / \partial q = r_n(q) > 0$$
$$f_n = k_n\big(l_{no} - l_n(q)\big)$$

$$t = -r_g(q) f_g(q) - r_n(q) f_n(q)$$
$$\frac{\partial t}{\partial q} = -\frac{\partial r_g}{\partial q} f_g - r_g \frac{\partial f_g}{\partial q} - \frac{\partial r_n}{\partial q} f_n - r_n \frac{\partial f_n}{\partial q}$$

This is the “tent-pole” effect

- Consequences of configuration-dependent moment arms:
- Opposing “ideal” (zero-impedance) tension actuators
  - agonist moment grows with angle, antagonist moment declines
  - *always* unstable
- Constant-stiffness actuators
  - stable only for limited tension
- Mammalian muscle:
- stiffness is proportional to tension
  - good approximation of complex behavior
  - can be stable for all tension



- Take-home messages:
- Kinematics matters
  - “Kinematic” stiffness may dominate
- Impedance matters
  - Zero output impedance may be highly undesirable

Intrinsically variable inertia

- Inertia is difficult to modulate via feedback but mechanism inertia is a strong function of configuration
- Use excess degrees of freedom to modulate inertia
  - e.g., compare contact with the fist or the fingertips
- Consider the apparent (translational) inertia at the tip of a 3-link open-chain planar mechanism
  - Use mechanism transformation properties
- Translational inertia is usually characterized by $\mathbf{p} = \mathbf{M}\mathbf{v}$
- Generalized (configuration space) inertia is $\boldsymbol{\eta} = \mathbf{I}(\boldsymbol{\theta})\boldsymbol{\omega}$
  - Jacobian: $\mathbf{v} = \mathbf{J}(\boldsymbol{\theta})\boldsymbol{\omega}$

    $\boldsymbol{\eta} = \mathbf{J}(\boldsymbol{\theta})^{t}\mathbf{p}$
  - Corresponding tip (workspace) inertia:

$$\mathbf{p} = \mathbf{J}(\boldsymbol{\theta})^{-t}\mathbf{I}(\boldsymbol{\theta})\mathbf{J}(\boldsymbol{\theta})^{-1}\mathbf{v}$$

$$\mathbf{M}_{tip} = \mathbf{J}(\boldsymbol{\theta})^{-t}\mathbf{I}(\boldsymbol{\theta})\mathbf{J}(\boldsymbol{\theta})^{-1}$$

- Snag: $\mathbf{J}(\theta)$ is not square—inverse $\mathbf{J}(\theta)^{-1}$ does not exist

Causal analysis

- Inertia is an admittance
  - prefers integral causality

$$\mathbf{v} = \mathbf{M}^{-1}\mathbf{p}$$
$$\boldsymbol{\omega} = \mathbf{I}(\boldsymbol{\theta})^{-1}\boldsymbol{\eta}$$

- Transform inverse configuration-space inertia
  - Corresponding tip (workspace) inertia
  - This transformation is *always* well-defined

$$\mathbf{v} = \mathbf{J}(\boldsymbol{\theta})\mathbf{I}(\boldsymbol{\theta})^{-1}\mathbf{J}(\boldsymbol{\theta})^{t}\mathbf{p}$$
$$\mathbf{M}_{\text{tip}}^{-1} = \mathbf{J}(\boldsymbol{\theta})\mathbf{I}(\boldsymbol{\theta})^{-1}\mathbf{J}(\boldsymbol{\theta})^{t}$$

- Does $\mathbf{I}(\theta)^{-1}$ always exist?
  - consider how we constructed $\mathbf{I}(\boldsymbol{\theta})$ from individual link inertias
  - $\mathbf{I}(\boldsymbol{\theta})$ must be symmetric positive definite, hence its inverse exists
- Does $\mathbf{M}_{\text{tip}}^{-1}$ always exist?
  - yes, but sometimes it loses rank
    - inverse mass goes to zero in some directions—can't move that way
  - causal argument: input force can always be applied
    - mechanism will "figure out" whether & how to move