EECS 106B/206B

Robotic Manipulation and Interaction

Valmik Prabhu

Announcements

Check out the website: http://inst.eecs.berkeley.edu/~ee106b/sp20

Join Piazza and Gradescope (M45XDR)

Lab 0 in lab section today / tomorrow

HW 1 is due 1/31

Project 1a will be released this week and will be discussed during lab section

Goals of this lecture

Apply the geometric formalism we covered last lecture to rigid body transforms

- How does this connect to forward/inverse kinematics?
- How does this connect to Jacobians?

Review Lagrangian dynamics

- Why doesn’t our geometric formalism work here?

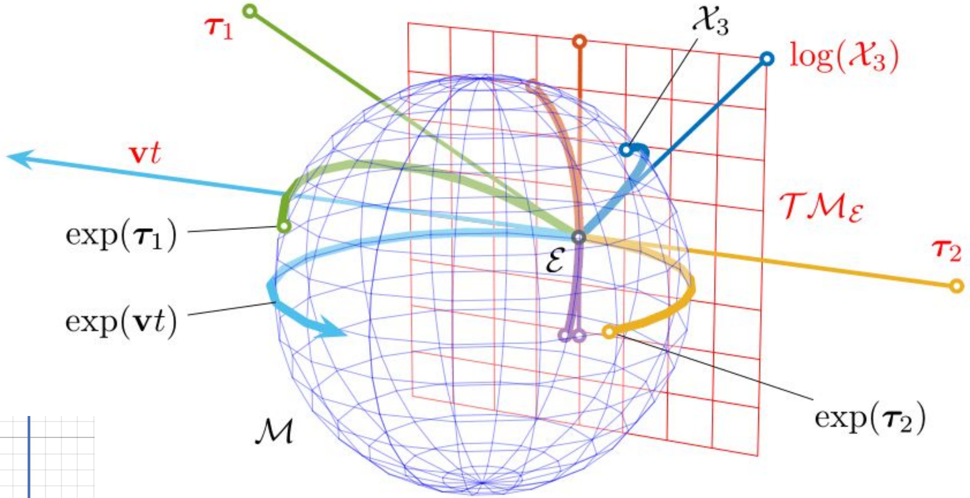
Review of Rigid Body Motion

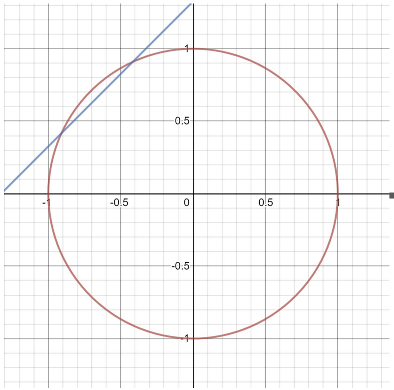
Content primarily from MLS Ch 2

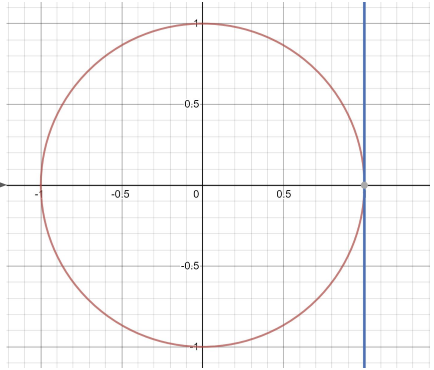
More advanced material may be found in MLS Appendix A.3

What We Covered Last Time

- Rotations are Lie Groups
- so(n) is its Lie Algebra
- Matrix Exponential
- Velocities to so(n)







Rigid Body Transformations

A rigid body transformation is an *affine* transformation consisting of a rotation followed by a translation

$$q_a = R_{ab} q_b + p_{ab}$$

Rigid body transformations have the following properties:

1. Length is preserved: $\|g(p) - g(q)\| = \|p - q\| \quad \forall p, q \in \mathbb{R}^3$
2. The orientation is preserved: $g(v \times w) = g(v) \times g(w) \quad \forall v, w \in \mathbb{R}^3$

Homogeneous Coordinates

Most of the mathematical tricks we have apply only to linear systems. Homogeneous coordinates are a clever bit of notation that make affine rigid body transforms into matrices (linear transforms)

$$q_a = R_{ab} q_b + p_{ab} \longrightarrow \begin{bmatrix} q_a \\ 1 \end{bmatrix} = \begin{bmatrix} R_a b & p_a b \\ 0 & 1 \end{bmatrix} \begin{bmatrix} q_b \\ 1 \end{bmatrix}$$

Point: $\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$ Vector: $\begin{bmatrix} x \\ y \\ z \\ 0 \end{bmatrix}$

Point - Point = Vector

Vector ± Vector = Vector

Point ± Vector = Point

Point + Point = n/a

Rigid Body Transforms as a Group

Rigid body transforms are also a Lie Group, the Special Euclidean Group SE(n)

$$SE(n) := \mathbb{R}^n \times SO(n)$$

Closure:

If $g_1, g_2 \in SE(n),\ \text{then } g_1 g_2 \in SE(n)$

Identity:

$I \in SE(n)$

Associativity:

Matrix multiplication is associative

Inverse:

$$g^{-1} = \begin{bmatrix} R^T & -R^T p \\ 0 & 1 \end{bmatrix} \in SE(n)$$

The Lie Algebra se(n)

The Lie Algebra to SE(n) is se(n)

$$\hat{\xi} = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix} \qquad \hat{\omega} \in so(n), v \in \mathbb{R}^n$$

While this is an nxn matrix, it can be defined by only n(n-1)/2 + n variables:

$$\xi = \begin{bmatrix} v \\ \omega \end{bmatrix}$$

This representation of se(n) is called a twist. The wedge* operator is used to transform between the two notations.

* While it is technically defined as the wedge operator, everyone calls it a hat

The Inverse Exponential: The Matrix Logarithm

The inverse of a matrix exponential is the matrix logarithm.

$$\theta = \arccos\left(\frac{tr(R^f(R^i)^{-1}) - 1}{2}\right)$$

$$\omega = \frac{1}{2\sin(T)}\begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{bmatrix}$$

$$\hat{\xi} = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix}$$

$$v = [(I - e^{\hat{\omega}T})\hat{\omega} + \omega\omega^T]^{-1}(p_f - R_f R_i^{-1} p_i)$$

Velocities of a Rigid Body

The velocities for rigid body transforms work exactly the same as the velocities for rotations!

Body Velocity:

$$\hat{\xi}^b_{AB} = g^{-1}_{AB}\dot{g}_{AB}$$

Spatial Velocity:

$$\hat{\xi}^s_{AB} = \dot{g}_{AB}g^{-1}_{AB}$$

Once again, the body velocity is defined as the Lie Algebra in the second frame (B), while the spatial velocity is defined as the Lie Algebra in the first (A).

Computing the Matrix Exponential

The matrix exponential is rather hard to compute for certain matrices, and computational solutions are prone to accumulating error. Thus we use the following formulas to analytically compute the matrix exponential:

SO(3): Rodrigues Formula

$$e^{\hat{\omega}\theta} = I + \hat{\omega}\sin(\theta) + \hat{\omega}^2(1 - \cos(\theta))$$

SE(3): Screw Theory

$$e^{\widehat{\xi}\theta} = \begin{bmatrix} e^{\widehat{\omega}\theta} & (I - e^{\widehat{\omega}\theta})(\omega \times v) + \omega\omega^T v\theta \\ 0 & 1 \end{bmatrix}$$

Forward Kinematics

We can derive Forward Kinematics from velocities by moving each joint one at a time, using the following differential equation:

$$\dot{g}_{ST} = \hat{\xi} g_{ST}$$

$$g_{ST}(T) = e^{\hat{\xi} T} g_{ST}(0)$$

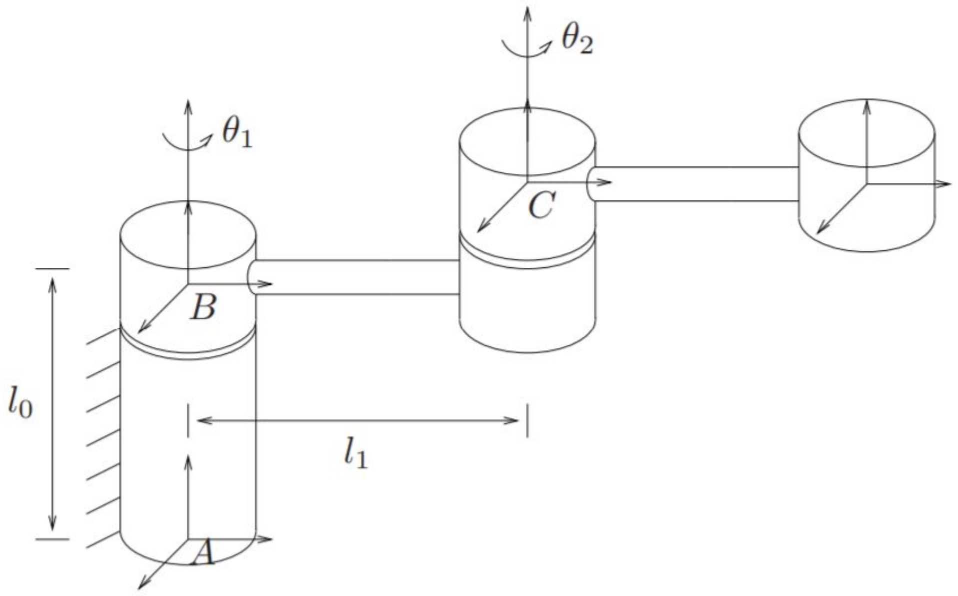


Figure from: MLS Chapter 2.4

Inverse Kinematics

Forward Kinematics is not an injective function; multiple sets of joint angles can result in the same end effector configuration. Thus, the inverse kinematics will have multiple solutions in general.

- An arbitrary 6-DOF manipulator has 16
- An elbow manipulator has 8
- A 7-DOF manipulator (like Baxter) has infinite solutions

There are two main problems with inverse kinematics:

1. How do you find a solution at all?
2. Which solution do you pick?

The Matrix Adjoint

We can define a change of basis on velocities since they're vectors

$$\hat{V}^s_{AB} = g_{AB}\hat{V}^b_{AB}g^{-1}_{AB}$$

But to change the basis of a twist, we define the *adjoint* matrix, which operates on vectors

$$V^s_{AB} = Ad_{g_{AB}}V^b_{AB}$$

$$Ad_{g_{AB}} = \begin{bmatrix} R_{AB} & \hat{p}_{AB}R_{AB} \\ 0 & R_{AB} \end{bmatrix} \quad Ad^{-1}_{g_{AB}} = \begin{bmatrix} R^T_{AB} & -R^T_{AB}\hat{p}_{AB} \\ 0 & R^T_{AB} \end{bmatrix} = Ad_{g^{-1}_{AB}}$$

Velocity of Forward Kinematics

We want to compute the following derivative

$$\frac{dg_{ST}(\theta(t))}{dt} = \frac{\partial g_{ST}(\theta)}{\partial \theta}\frac{d\theta}{dt} = \frac{\partial g_{ST}(\theta)}{\partial \theta}\dot{\theta}$$

How do we define this?

$$\frac{\partial g_{ST}(\theta)}{\partial \theta}\dot{\theta} = \frac{\partial g_{ST}(\theta)}{\partial \theta_1}\dot{\theta}_1 + \frac{\partial g_{ST}(\theta)}{\partial \theta_2}\dot{\theta}_2 + \cdots \frac{\partial g_{ST}(\theta)}{\partial \theta_n}\dot{\theta}_n$$

Deriving the Jacobian

$$\hat{V}_{ST}^{s} = \dot{g}_{ST} g_{ST}^{-1} = \sum_i \frac{\partial g_{ST}}{\partial \theta_i} \dot{\theta}_i g_{ST}^{-1} = \sum_i \left(\frac{\partial g_{ST}}{\partial \theta_i} g_{ST}^{-1} \right) \dot{\theta}_i$$

$$\frac{\partial g_{ST}}{\partial \theta_i} g_{ST}^{-1} \longrightarrow \left(\frac{\partial g_{st}}{\partial \theta_i} g_{st}^{-1} \right)^{\vee} = \mathrm{Ad}_{\left(e^{\widehat{\xi}_1 \theta_1} \cdots e^{\widehat{\xi}_{i-1} \theta_{i-1}}\right)} \xi_i$$

This is the velocity from joint i

The Jacobian

Using the vee operator, we can turn this summation into a matrix multiplication

$$J^s_{st}(\theta) = \begin{bmatrix} \xi_1 & \xi_2' & \cdots & \xi_n' \end{bmatrix}$$

$$J^b_{st}(\theta) = \begin{bmatrix} \xi_1^\dagger & \cdots & \xi_{n-1}^\dagger & \xi_n^\dagger \end{bmatrix}$$

$$\xi_i' = \mathrm{Ad}_{\left(e^{\widehat{\xi}_1\theta_1} \cdots e^{\widehat{\xi}_{i-1}\theta_{i-1}}\right)} \xi_i$$

$$\xi_i^\dagger = \mathrm{Ad}^{-1}_{\left(e^{\widehat{\xi}_i\theta_i} \cdots e^{\widehat{\xi}_n\theta_n} g_{st}(0)\right)} \xi_i$$

$$V^s_{ST} = J^s_{ST}(\theta)\dot{\theta}$$

$$V^b_{ST} = J^b_{ST}(\theta)\dot{\theta}$$

$$J^s_{st}(\theta) = \mathrm{Ad}_{g_{st}(\theta)} J^b_{st}(\theta).$$

The Pseudoinverse Jacobian

We have a way to generate body or spatial velocities of our end effector from the robot's joint velocities. Can we solve the inverse of this problem and generate joint velocities from a desired end effector velocity?

$$V = J(\theta)\dot{\theta} \implies \dot{\theta} = J^{-1}(\theta)V$$

If J is not a square matrix, its inverse is not defined. We have to use the Moore-Penrose pseudoinverse instead.

$$J^{\dagger} = J^{T}(JJ^{T})^{-1} \qquad \dot{\theta} = J^{\dagger}(\theta)V$$

This is essentially a Least-Squares solution to picking the joint velocities

Wrenches: The duals of twists

A wrench represents an instantaneous force

$$F = \begin{bmatrix} f \\ \tau \end{bmatrix} \begin{matrix} \text{linear component } f \in \mathbb{R}^3 \\ \text{angular component } \tau \in \mathbb{R}^3 \end{matrix}$$

The work caused by a wrench and a velocity is $W = \int V^T F dt$

Two wrenches are equivalent if they generate the same work for all possible V

To change coordinate frames, we use the transpose adjoint: $F_b = Ad^T_{g_{ab}} F_a$

To get joint torques from a wrench, we use J^T: $\tau = J^T(\theta) F$

Lagrangian Dynamics

Rather than basing dynamics on conservation of momentum, we instead use conservation of energy

Since constraint forces do no work, they don't contribute to the total system energy (and we can ignore them)

Thus, we can write our dynamics in terms of *generalized coordinates* (joint angles) instead of rigid body transforms

Dynamics: The Lagrangian

We define the lagrangian

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q),$$

T is kinetic energy, V potential energy

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \Upsilon$$

Υ is a vector of external forces

Energy of an Open-Chain Manipulator

We define coordinate frames for each link (the black dots).

We can define the inertia of each link in its body frame.

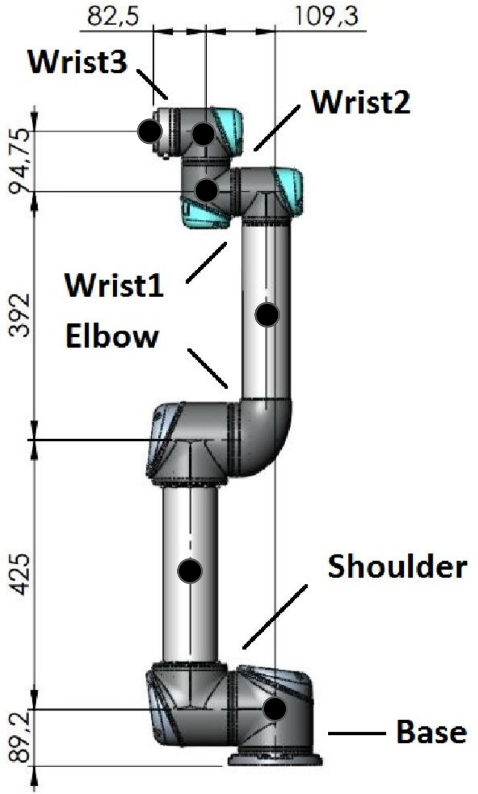
$$\mathcal{M}_i = \left[\begin{array}{ccc|ccc} m_i & & 0 & & & \\ & m_i & & & 0 & \\ 0 & & m_i & & & \\ \hline & & & I_{xi} & & 0 \\ & 0 & & & I_{yi} & \\ & & & 0 & & I_{zi} \end{array}\right]$$

$$V^b_{sl_i} = J^b_{sl_i}(\theta)\dot{\theta}$$

$$M(\theta) = \sum^{n} J_i^T(\theta)\mathcal{M}_i J_i(\theta)$$

$$T_i(\theta, \dot{\theta}) = \frac{1}{2}(V^b_{sl_i})^T \mathcal{M}_i V^b_{sl_i} = \frac{1}{2}\dot{\theta}^T J_i^T(\theta)\mathcal{M}_i J_i(\theta)\dot{\theta},$$

$$\boxed{\begin{aligned} T(\theta, \dot{\theta}) &= \sum_{i=1}^{n} T_i(\theta, \dot{\theta}) =: \frac{1}{2}\dot{\theta}^T M(\theta)\dot{\theta}. \\ V(\theta) &= \sum_{i=1} V_i(\theta) = \sum_{i=1} m_i g h_i(\theta). \end{aligned}}$$



Equations of Motion

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + G(\theta) = \Upsilon$$

$$\frac{d}{dt}\frac{\partial}{\partial\dot{\theta}}\left(\dot{\theta}^T M(\theta)\dot{\theta}\right)$$

$$\frac{d}{dt}\frac{\partial V(\theta,\dot{\theta})}{\partial\dot{\theta}} + \frac{\partial}{\partial\theta}\left(\dot{\theta}^T M(\theta)\dot{\theta}\right)$$

$$\frac{\partial V(\theta)}{\partial\theta}$$

Limits of Lagrangian Dynamics

- Lagrangian dynamics require a *vector* of parameters
- Dynamics only make sense with good parameters
- Lagrangian dynamics can only deal with *conservative forces*

Sources

MLS Ch 2, Ch 3, Ch 4, Appendix A