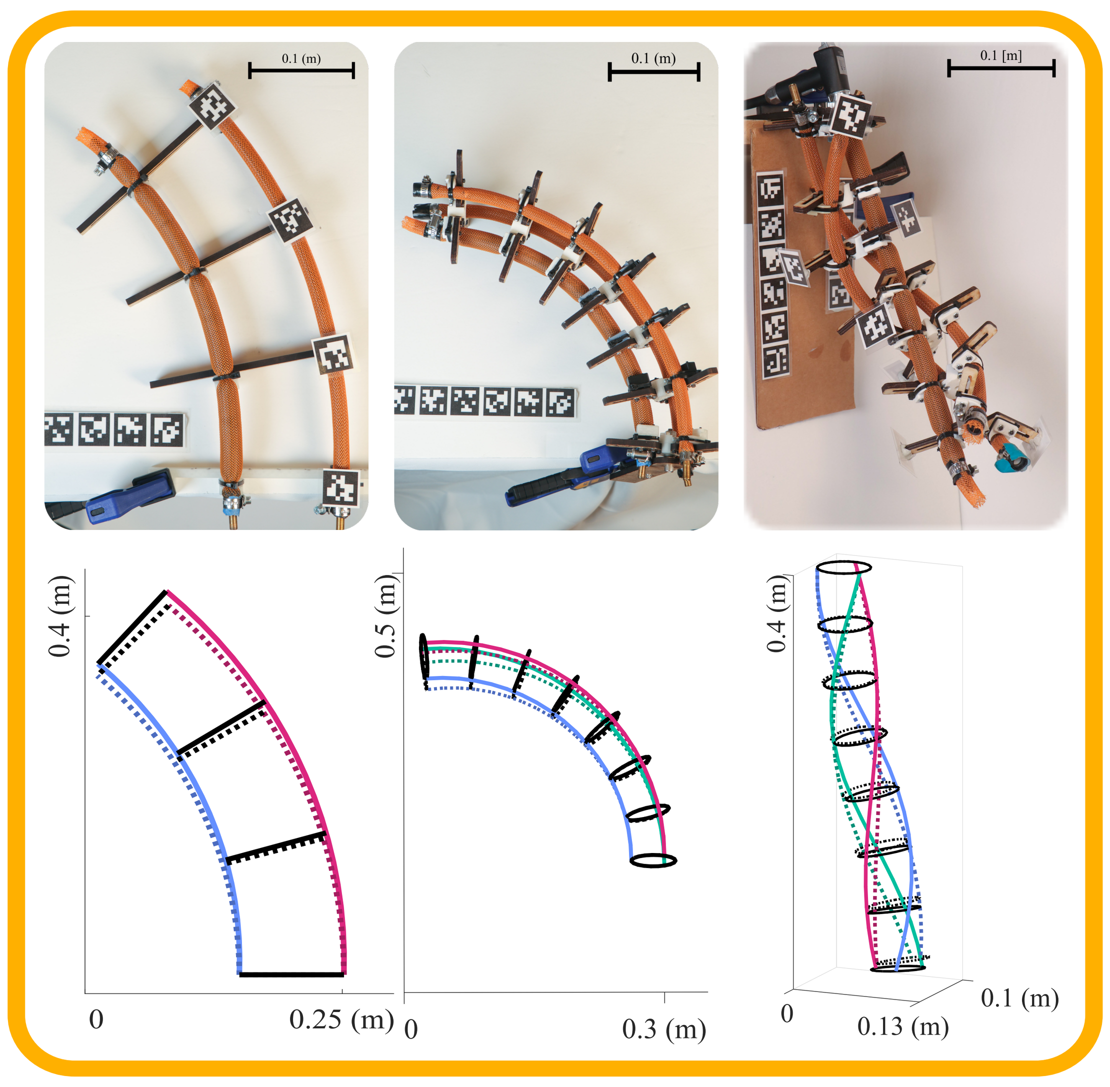
Linear Kinematics for General Constant Twist Manipulators

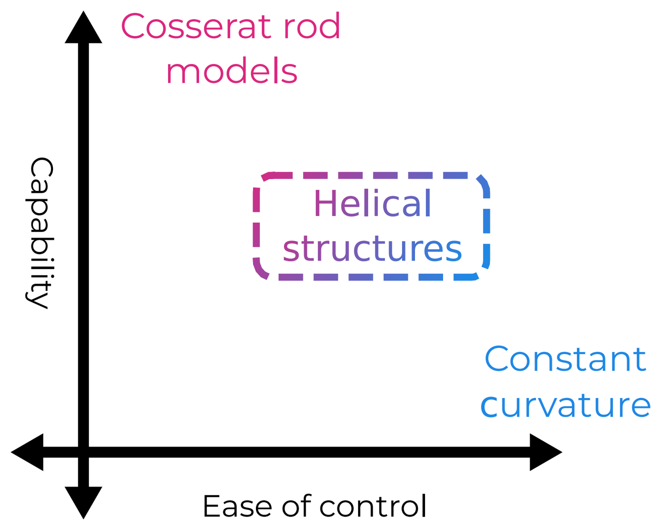
Olin College of Engineering
Oregon State University

Bill Fan[1], Farhan Rozaidi[2], Capprin Bass[2], Gina Olson[3], Melinda Malley[1], Ross L Hatton[2]
[1]Olin College of Engineering; [2]Oregon State University; [3]University of Massachusetts, Amherst.



Motivation

- HISSbot (design and construction in Rozaidi et al., inspection applications in Wilson et al., both RoboSoft 2023) uses helical actuators to create sidewinding snake motion.
- Other works have taken inspiration from octopus tentacles and elephant trunks to show that helical structures can improve manipulator dexterity.
- To control these emergent helical robots, we need a model that combines the simplicity of constant curvature models with the generality of Cosserat rod models.



Contribution

We present a novel model for constant twist manipulator mechanics that:

1. Linearly maps actuator configuration to manipulator geometry
2. Generalizes between different manipulators using shared parameters.
3. Generalizes across different types of actuators: muscles v.s. tendons

We validate our model on physical manipulators with McKibben muscles.

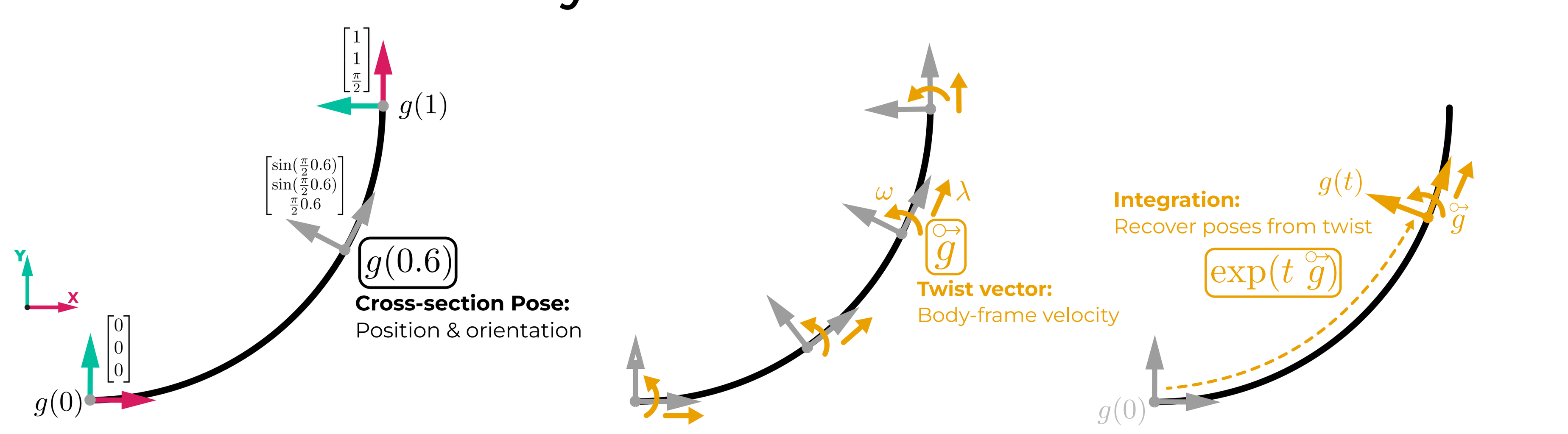
Manipulator Geometry — Vector of actuator neutral lengths given inputs $\mathbf{q}$

$$\overset{\circ\!\to}{g} = \underbrace{\mathbf{A}}_{\text{Manipulator cross-section geometry}}\mathbf{l}(\mathbf{q})$$

References:
- F. Rozaidi, E. Waters, O. Dawes, J. Yang, J. Davidson, R.L. Hatton, "HISSBot: Sidewinding with a Soft Snake Robot". Robosoft 2023
- C. Wilson, J. Karam, C. Votzke, F. Rozaidi, C.J. Palmer, R.L. Hatton, M.L. Johnston, "Modular Sensor Integration into Soft Robots Using Stretchable Wires for Nuclear Infrastructure Inspection and Radiation Spectroscopy". Robosoft 2023

Kinematics

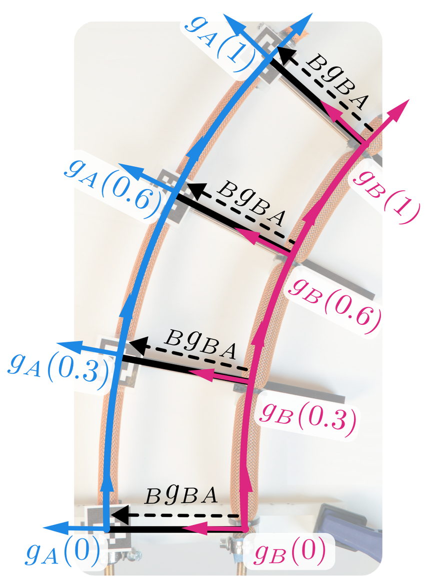
1. Actuator Geometry



$$g(t)=\begin{bmatrix}x\\y\\\theta\end{bmatrix}=\begin{bmatrix}\text{x posn.}\\\text{y posn.}\\\text{heading}\end{bmatrix}=\begin{bmatrix}\sin(\frac{\pi}{2}t)\\\sin(\frac{\pi}{2}t)\\\frac{\pi}{2}t\end{bmatrix}\qquad \vec{g}=\begin{bmatrix}\lambda\\\gamma\\\omega\end{bmatrix}=\begin{bmatrix}\text{forward vel.}\\\text{lateral vel.}\\\text{angular vel.}\end{bmatrix}=\begin{bmatrix}1\\0\\\frac{\pi}{2}\end{bmatrix}\qquad g(t)=g(0)\circ\exp(t\,\vec{g})$$

- The cross-section poses of an actuator describe its geometry along its length.
- Actuators with constant deformation have cross-sections that change with constant velocity in the body frame.
- Integrating this body-frame velocity via the exp. map recovers the geometry along the rod.

2. Manipulator Geometry

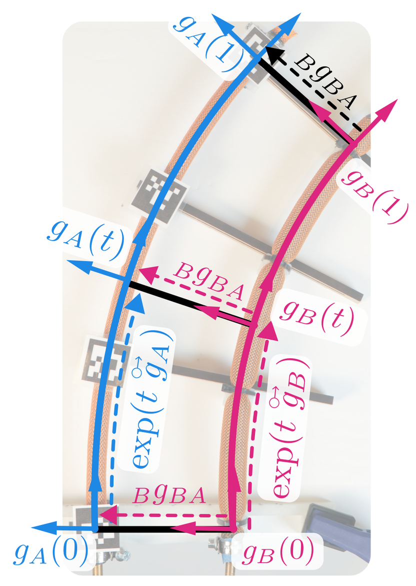


- Cross-section seperators require transformations between poses along actuators to be identical at points where the seperators are present.
- This creates a set of constraints between discrete poses:

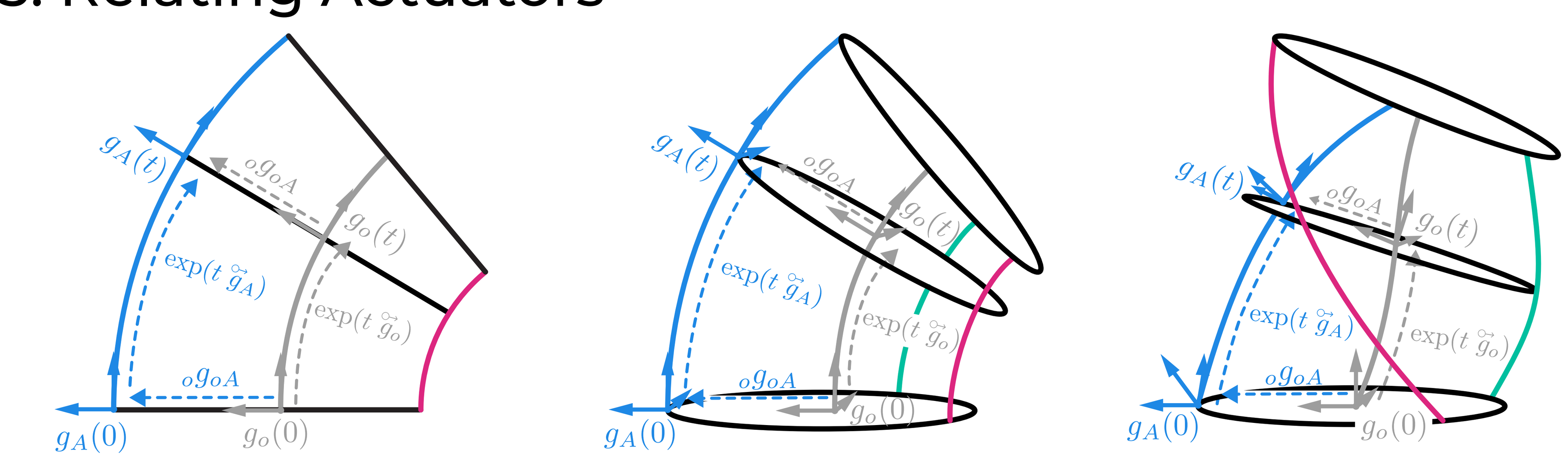
$$g_A(t)=g_B(t)\circ{}_Bg_{BA}$$
for all t in $[0, 0.3, 0.6, 1]$

- We extend this constraint uniformly across the whole manipulator, to points without seperators.
- We now formulate the constraints in terms of twist vectors:

$${}_Bg_{BA}\circ\exp(t\,\vec{g}_A)=\exp(t\,\vec{g}_A)\circ{}_Bg_{BA}$$
for all $0\le t\le 1$



3. Relating Actuators



- Each actuator's twist vector is related to the centerline twist vector as two paths that arrive at the same end.

$${}_og_{oA}\circ\exp(t\vec{g}_A)=\exp(t\vec{g}_o)\circ{}_og_{oA}$$

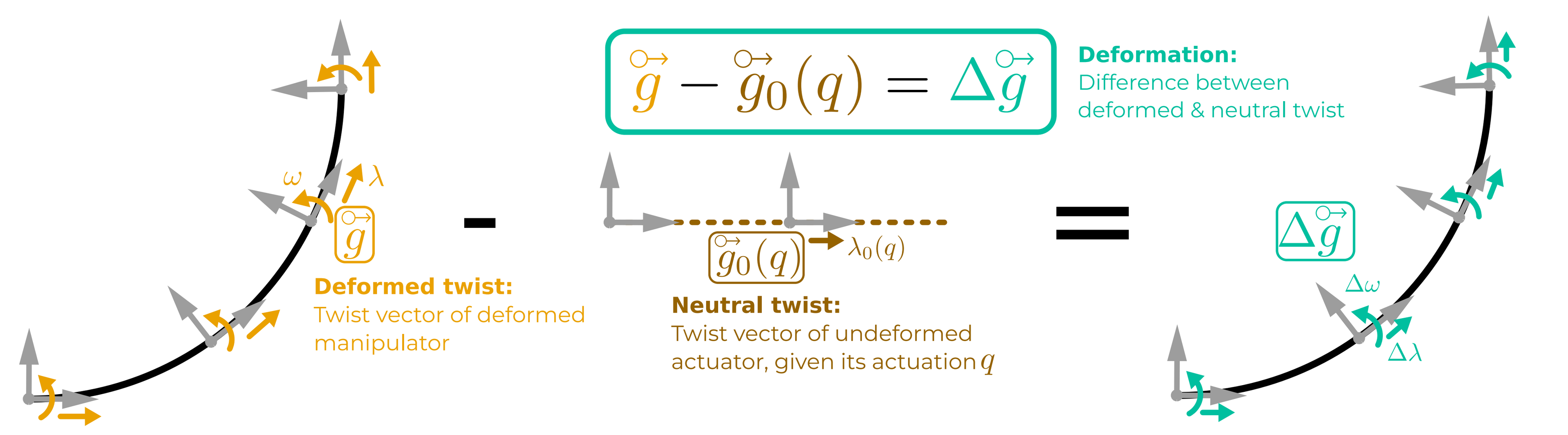
- Given the cross-section geometry, the Adjoint map expresses this relationship as a linear mapping between twist vectors.

$$\boxed{\vec{g}_A=\mathrm{Ad}^{-1}_{{}_og_{oA}}\vec{g}_o}$$
Adjoint Map: Maps twist vector of centerline to actuators given cross-section geometry

Mechanics

1. Actuator Deformation

$$\boxed{\vec{g}-\vec{g}_0(q)=\Delta\vec{g}}$$
Deformation: Difference between deformed & neutral twist



$$\vec{g}=\begin{bmatrix}\lambda\\\gamma\\\omega\end{bmatrix}=\begin{bmatrix}1\\0\\\frac{\pi}{2}\end{bmatrix}\qquad \vec{g}_0(q)=\begin{bmatrix}\lambda_0(q)\\0\\0\end{bmatrix}=\begin{bmatrix}0.7\\0\\0\end{bmatrix}\qquad \Delta\vec{g}=\begin{bmatrix}\Delta\lambda\\\Delta\gamma\\\Delta\omega\end{bmatrix}=\begin{bmatrix}\text{stretch}\\\text{shear}\\\text{bend}\end{bmatrix}=\begin{bmatrix}0.3\\0\\\frac{\pi}{2}\end{bmatrix}$$

- Consider a rod deformed from its neutral state via bending and stretching.
- The neutral state of an actuator contains its actuated length $\lambda_0(q)$, but no additional deformation.
- The difference between an actuator's actual and neutral twist vectors is its deformation.

Constituitive Laws:
- Linear elasticity grants simple relations for the forces and elastic energy of a single actuator.

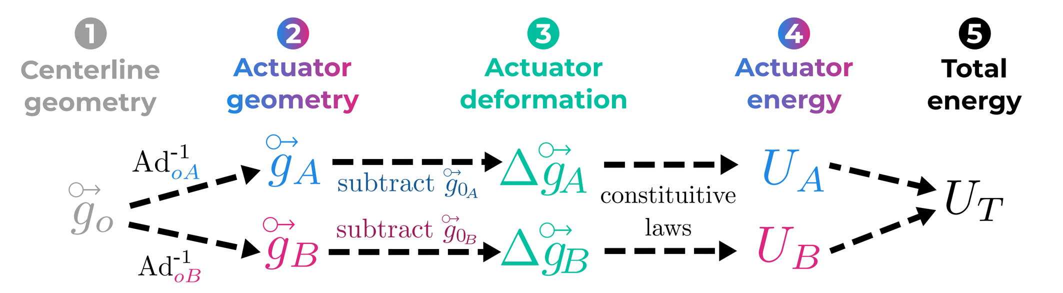
Force & Moments: $f=K\Delta\vec{g}$

Elastic energy: $U=\frac{1}{2}\Delta\vec{g}^TK\Delta\vec{g}$

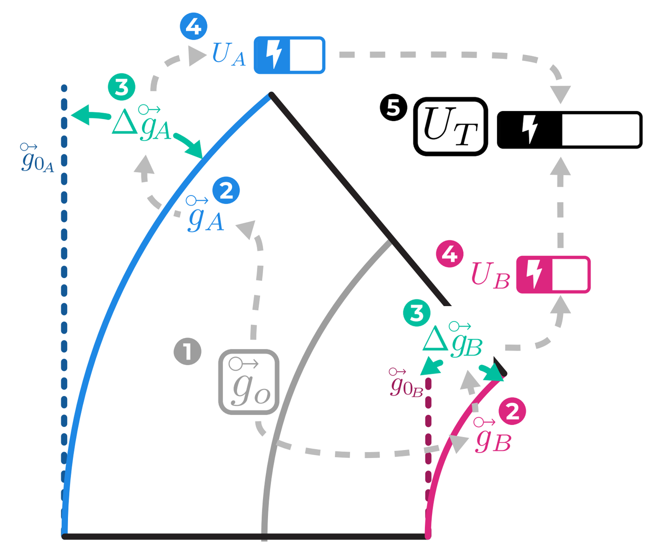
Linear stiffness: $K=\begin{bmatrix}K_\lambda&0&0\\0&K_\gamma&0\\0&0&K_\omega\end{bmatrix}$

2. Manipulator Deformation

We combine all prior tools to map any manipulator centerline twist $\vec{g}_o$ to the total elastic energy stored in the manipulator:

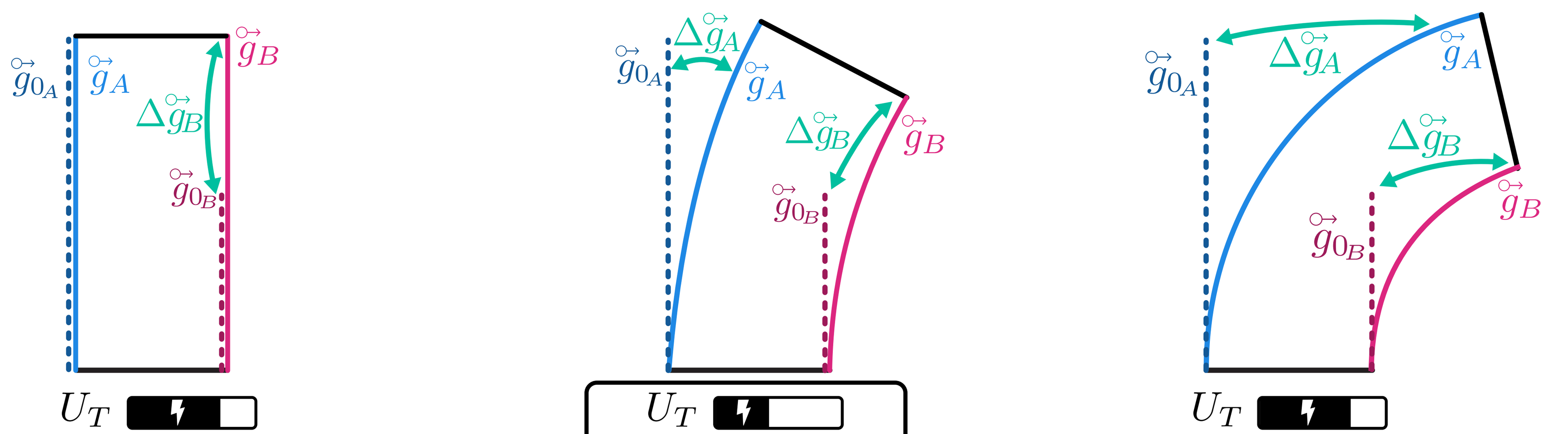


$$\vec{g}_o\xrightarrow{\mathrm{Ad}^{-1}_{oA}}\vec{g}_A\xrightarrow{\text{subtract }\vec{g}_{0_A}}\Delta\vec{g}_A\xrightarrow{\text{constituitive laws}}U_A\to U_T$$
$$\vec{g}_o\xrightarrow{\mathrm{Ad}^{-1}_{oB}}\vec{g}_B\xrightarrow{\text{subtract }\vec{g}_{0_B}}\Delta\vec{g}_B\to U_B\to U_T$$



3. Manipulator Equilibrium

- Find the manipulator centerline twist vector $\vec{g}_o$ that minimizes the total elastic energy U_T.
- Because energy is a sum-of-squares in terms of $\vec{g}_o$, this minimization is a linealeast-squares problem.



Model Validation