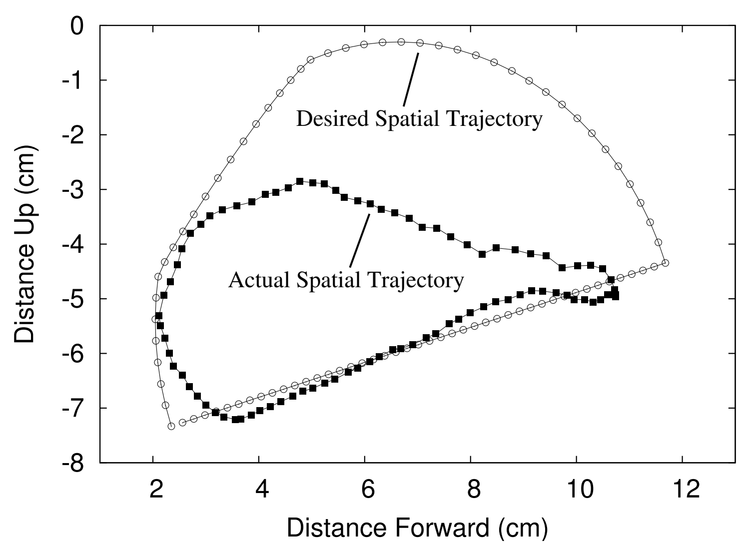
A Model-Based Approach to Robot Joint Control

Daniel Stronger and Peter Stone

University of Texas at Austin

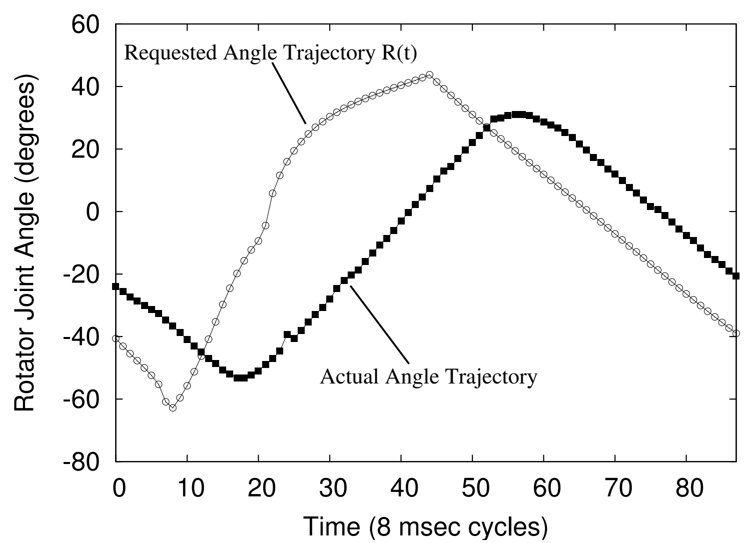
Introduction

- Robotic joints do not always behave as desired.
- We create a model of the joint's behavior.
- We use the model to make requests that yield the desired behavior.
- This approach is implemented and validated on a Sony Aibo ERS-210A.



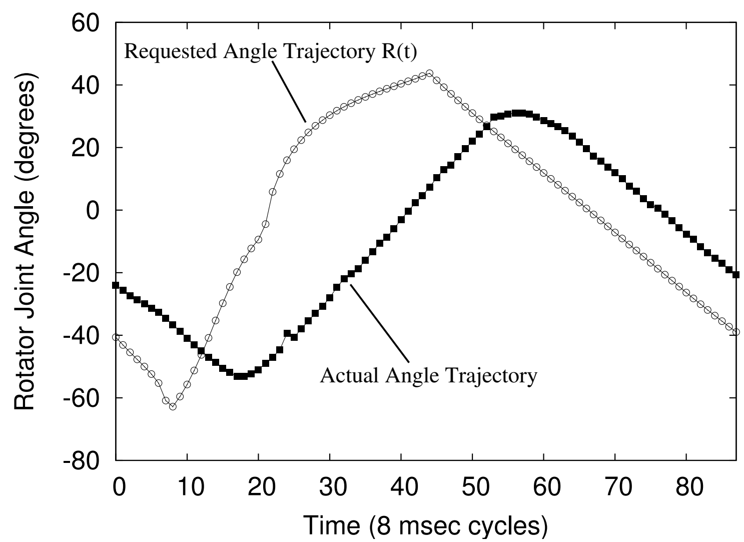
Constructing a Model

- The Aibo's four legs each have three joints.
- We use inverse kinematics to convert desired foot locations into desired joint angles.
- We can understand the inaccuracies in the foot location by analyzing inaccuracies in the joint angles.
  - Compare *requested angles* and *actual angles*



Constructing a Model

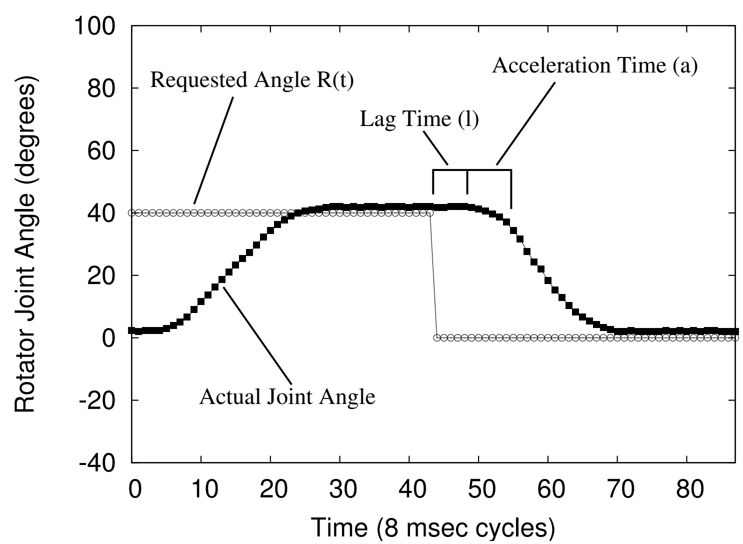
- Identify the features of the model.
  - Time lag
  - Angular velocity cap?
  - Angular acceleration cap?



Performing Experiments

- Request experimental trajectories.
- Observe resultant actual angles.
- With θ_{test} of 40 degrees:

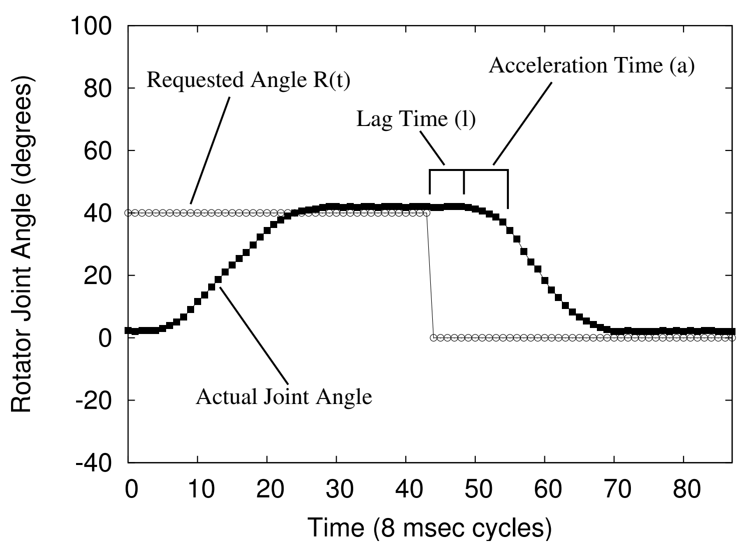
$$R(t) = \begin{cases} \theta_{test} & \text{if } t < \frac{t_{step}}{2} \\ 0 & \text{if } t \geq \frac{t_{step}}{2} \end{cases}$$

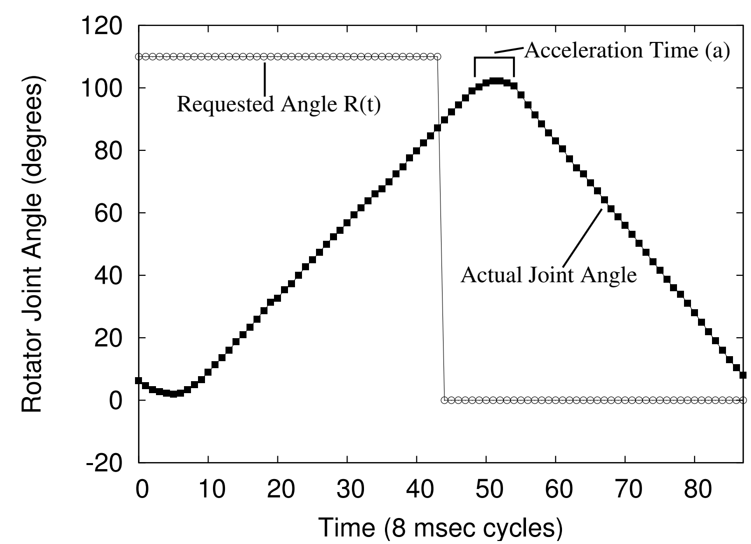


Performing Experiments

- Request experimental trajectories.
- Observe resultant actual angles.
- With θ_{test} of 40 and 110 degrees:

$$R(t) = \begin{cases} \theta_{test} & \text{if } t < \frac{t_{step}}{2} \\ 0 & \text{if } t \geq \frac{t_{step}}{2} \end{cases}$$

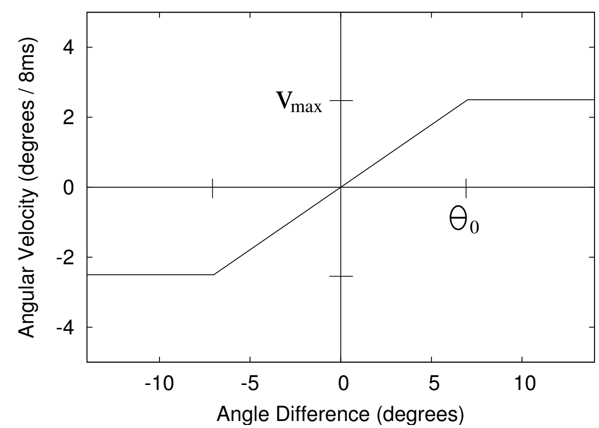




Experimental Findings

- Lag time $l = 4t_u$ ($t_u = 8$ms).
- Maximum velocity $v_{\max} = 2.5$ degrees/t_u.
- Acceleration **time** $a = 6t_u$.
- Despite maximum velocity, within a threshold higher angular differences mean higher velocities.

$$f(x) = \begin{cases} v_max & \text{if } x \geq \theta_0 \\ x \cdot \frac{v_{\max}}{\theta_0} & \text{if } -\theta_0 < x < \theta_0 \\ -v_{max} & \text{if } x \leq -\theta_0 \end{cases}$$



  - Angle distance threshold $\theta_0 = 7$ degrees.

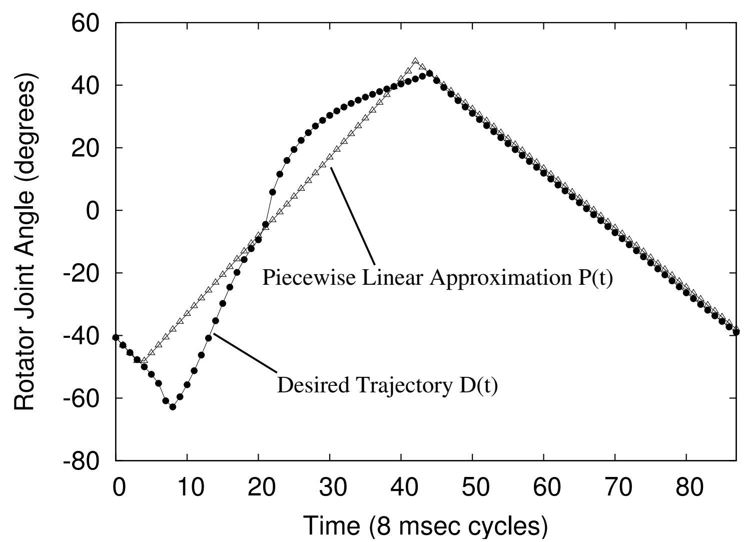
The Joint Model

- Need model to satisfy observed behavior
- Use averaging to achieve desired effect:

$$M_R(t) = M_R(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(R(t-i) - M_R(t-1))$$

Inverting the Model

- Invert the model to find requests that yield the desired behavior according to the model.
- Difficult to invert model mathematically
  - Desired trajectories exceed the velocity restriction.
  - Difficult to find trajectories in range of model.
- Solution: use piecewise linear approximation.



Inverting Lines

- What requests cause the joint to move at a constant velocity, m?

Inverting Lines

- What requests cause the joint to move at a constant velocity, m?
- Requests that move at the same velocity, offset by a constant, C_m.

Inverting Lines

- What requests cause the joint to move at a constant velocity, m?
- Requests that move at the same velocity, offset by a constant, C_m.
- We define a linear series of requests, $L(t)$, that moves at velocity m, i.e. $L(t) = L(t-1) + m$.
- By applying the model to L, we can find the offset C_m that applies at this slope.

$$M_L(t) = L(t) - C_m$$

- This is what we need to find the inverse of lines.

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(L(t-i) - M_L(t-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(L(t-1) - m(i-1) - M_L(t-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

$$S(x) = \frac{1}{a} \sum_{i=l+1}^{l+a} f(x - m(i-1))$$

$$M_L(t) = M_L(t-1) + S(\delta(t-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

$$S(x) = \frac{1}{a} \sum_{i=l+1}^{l+a} f(x - m(i-1))$$

$$L(t) - \delta(t) = L(t-1) - \delta(t-1) + S(\delta(t-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a} \sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

$$S(x) = \frac{1}{a} \sum_{i=l+1}^{l+a} f(x - m(i-1))$$

$$\delta(t) = \delta(t-1) + m - S(\delta(t-1))$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a}\sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

$$S(x) = \frac{1}{a}\sum_{i=l+1}^{l+a} f(x - m(i-1))$$

$$\delta(t) = \delta(t-1) + m - S(\delta(t-1))$$

$$S(\delta(t)) = m$$

Inverting Lines

- First, define $\delta(t) = L(t) - M_L(t)$.
- Plug in L for the requests:

$$M_L(t) = M_L(t-1) + \frac{1}{a}\sum_{i=l+1}^{l+a} f(\delta(t-1) - m(i-1))$$

$$S(x) = \frac{1}{a}\sum_{i=l+1}^{l+a} f(x - m(i-1))$$

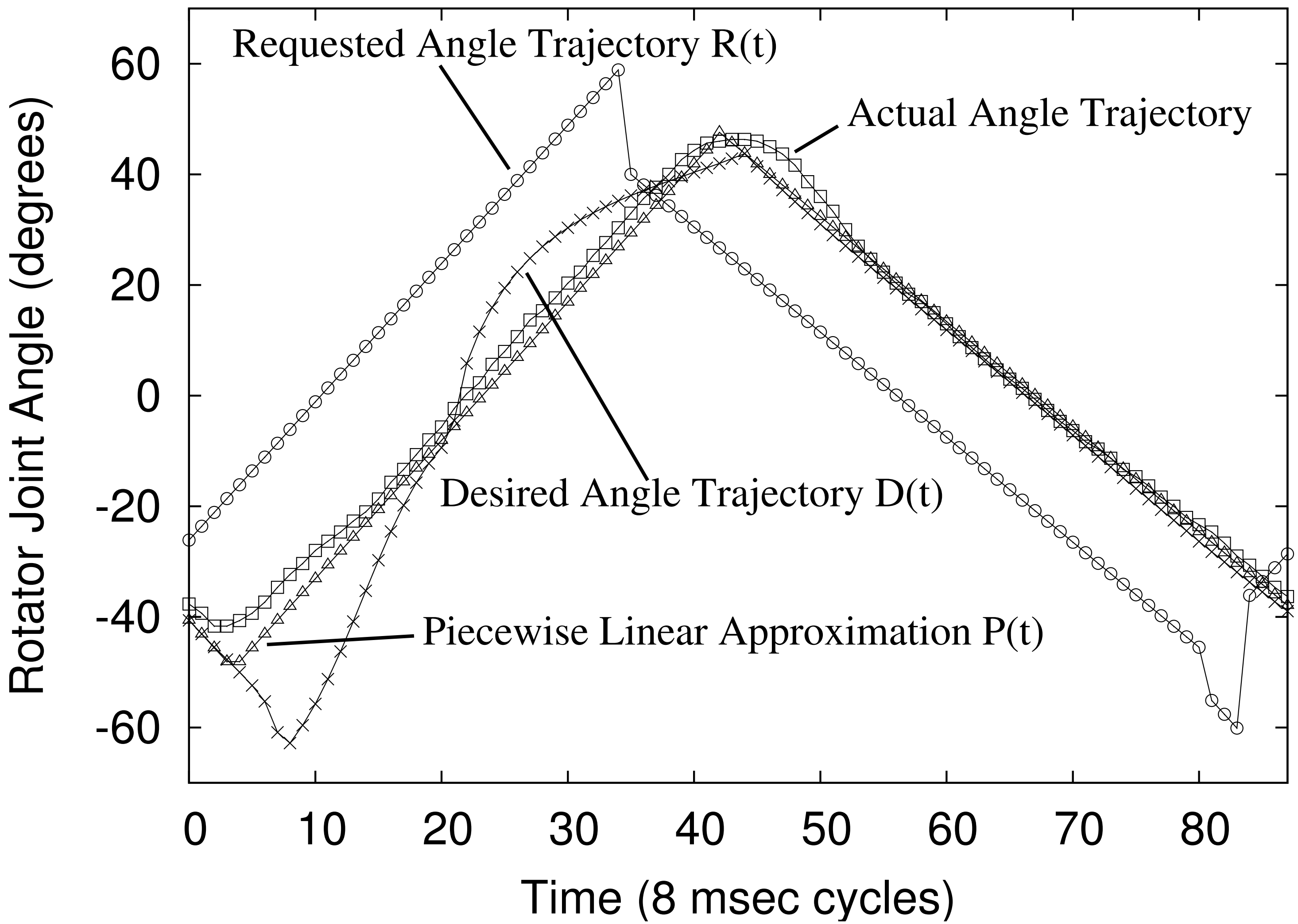
$$\delta(t) = \delta(t-1) + m - S(\delta(t-1))$$

$$S(C_m) = m$$

Combining Inverted Line Segments

- Need appropriate transitions between lines.
- Switch between inverted lines before desired lines.
  - For the lag: l
  - For half the acceleration time: $\frac{a}{2}$
- Transition between inverted lines $l + \frac{a}{2}$ before corresponding transition between desired lines.

Experimental results



Experimental Results

- Compute distances between angular trajectories
  - Des: Desired angles
  - Dir: Angles attained by requesting the desired angles directly
  - Pwl: Piecewise linear approximation
  - MB: Angles attained using model-based method
- Treat trajectories as vectors.
  - Use L_2 norm.
  - Use L_∞ norm.

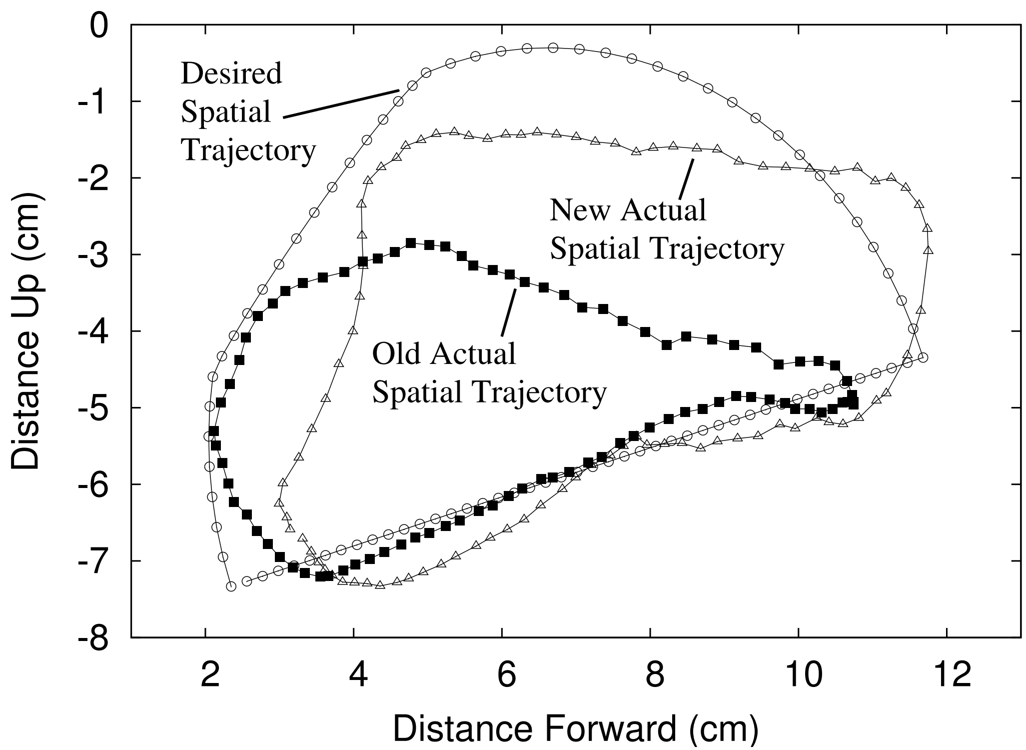
Experimental Results

- Compute distances between angular trajectories

Comparison	Rotator	Abductor	Knee
$\boldsymbol{L_2(Des, Dir)}$	**31.0(±0.2)**	**29.0(±0.2)**	**20.1(±0.1)**
$L_\infty(Des, Dir)$	57.2(±0.3)	59.5(±0.5)	42.6(±0.3)
$\boldsymbol{L_2(Des, MB)}$	**9.1(±0.2)**	**10.4(±0.1)**	**5.6(±0.2)**
$L_\infty(Des, MB)$	29.4(±0.8)	24.5(±0.7)	11.1(±0.5)
$L_2(Pwl, MB)$	2.7(±0.4)	2.7(±0.3)	2.6(±0.2)
$L_\infty(Pwl, MB)$	6.4(±0.6)	6.0(±0.4)	6.2(±0.7)

Experimental Results

- Compare foot location in physical space.
- Direct method: L_2: 3.23 ± 0.01 cm; L_∞: 4.61 ± 0.05 cm
- Model-based method: L_2: 1.21 ± 0.01 cm; L_∞: 2.34 ± 0.01 cm



Conclusion and Future Work

- By modeling the inaccuracies in robotic joints, we can compute joint requests that more closely yield the desired effects.
- Possibilities for future work:
  - Implement this approach on other platforms.
  - Model the effects of external forces.
  - Have the robot learn its own joint models.