Robotics and Animatronics in Disney

Lecture 6: Humanoid Modeling and Control

Katsu Yamane
kyamane@disneyresearch.com
Disney Research Pittsburgh

Goals

- Introduce a method for automatically simplifying dynamics models for humanoid robots
- Introduce model identification methods for humanoid robots
- Discuss the right level of details for robot modeling

Automatic Model Reduction for Humanoid Robots

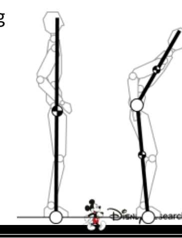
[Yamane 2012]

Disney Research Pittsburgh

Motivation

Building simplified models for humanoid control

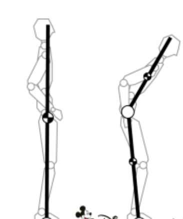
1. Choose a simple mechanical system
2. Derive and linearize the equation of motion
3. Define state and input mapping



Motivation

Issues

- Accuracy
- Model parameters
- State/input mapping
- Other models/configuration



Contributions

Automatic derivation of simplified dynamics models

- Given: nominal pose, contact constraints, reduced DOF

Unified state and input mapping

- Use the same code for any simplified model

Related Work

- Humanoid robots: inverted pendulum [Kajita 1995], cart-table [Kajita 2003], IP with reaction wheel [Lee 2007] …
  - Based on intuition
  - Comparison to the original dynamics [Goswami 2008]
- Model reduction in structural mechanics and fluid dynamics [Hyland 1983; Lall 2003], graphics [Treuille 2006; James 2003]
  - Thousands of degrees of freedom but only a few inputs
  - Often assume stable system

Simplification Process

Full equation of motion (n joints) → Full constrained dynamics model

Contact constraints (m independent constraints) → Full constrained dynamics model

Full constrained dynamics model → (linearization) → Linear full constrained dynamics model

Linear full constrained dynamics model → (dimensionality reduction) → Linear reduced dynamics model (k DOF ($k \ll n$))

Linear reduced dynamics model → State-space model ($2k$ states)

Disney Research, Pittsburgh

Simplification Process

Full equation of motion (n joints) → Full constrained dynamics model

Contact constraints (m independent constraints) → Full constrained dynamics model

Full constrained dynamics model → (linearization) → Linear full constrained dynamics model

Linear full constrained dynamics model → (dimensionality reduction) → Linear reduced dynamics model (k DOF ($k \ll n$))

Linear reduced dynamics model → State-space model ($2k$ states)

Disney Research, Pittsburgh

Full Model with Contacts

Full equation of motion

$$M(\theta)\ddot{\theta} + c(\theta,\dot{\theta}) + g(\theta) = S^T\tau + J_C^T(\theta)f_C$$

($M > 0$, symmetric) $\theta \in \mathbb{R}^n$

Contact constraints

$$J_C\ddot{\theta} + \dot{J}_C\dot{\theta} = 0$$

Full constrained dynamics model

$$\ddot{\theta} = \Phi(\theta)S^T\tau + \phi(\theta,\dot{\theta})$$

($\Phi \geq 0$, symmetric)

$$\begin{cases} \Phi = M^{-1} - M^{-1}J_C^T\left(J_C M^{-1} J_C^T\right)^{-1} J_C M^{-1} \\ \phi = -M^{-1}J_C^T\left(J_C M^{-1} J_C^T\right)^{-1} \dot{J}_C\dot{\theta} - \Phi(c+g) \end{cases}$$

Disney Research, Pittsburgh

Linearization

Full constrained dynamics model

$$\ddot{\theta} = \Phi(\theta)S^T\tau + \phi(\theta,\dot{\theta})$$

$\theta = \theta_0 + \delta\theta, \dot{\theta} = 0 + \delta\dot{\theta}, \ddot{\theta} = 0 + \delta\ddot{\theta}$

Linear full constrained dynamics model

$$\delta\ddot{\theta} = \Phi_0 S^T\delta\tau + \Gamma\delta\theta + \Lambda\delta\dot{\theta}$$

$\Phi_0 = \Phi(\theta_0)$

Disney Research, Pittsburgh

Simplification Process

Full equation of motion (n joints) → Full constrained dynamics model

Contact constraints (m independent constraints) → Full constrained dynamics model

Full constrained dynamics model → (linearization) → Linear full constrained dynamics model

Linear full constrained dynamics model → (dimensionality reduction) → Linear reduced dynamics model (k DOF ($k \ll n$))

Linear reduced dynamics model → State-space model ($2k$ states)

Disney Research, Pittsburgh

Dimensionality Reduction

Linear full constrained dynamics model

? $\delta\ddot{\theta} = \Phi_0 S^T \delta\tau + \Gamma\delta\theta + \Lambda\delta\dot{\theta}$

Linear reduced dynamics model k DOF ($k \ll n$)

$\hat{M}\ddot{q} + \hat{C}\dot{q} + \hat{G}q = u$

$q \in \mathbb{R}^k$: generalized coordinates of the simplified model

What to maintain after reduction?

- Somewhat task-dependent
- Here we choose kinetic energy

Kinetic Energy of the Constrained System

- Need inertia matrix, but Φ_0 is not invertible
- Singular value decomposition: $\Phi_0 = U\Sigma U^T$

Singular values

$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_l > \sigma_{l+1} = \sigma_{l+2} = \cdots = \sigma_n = 0$

$\bar{\Sigma}$

$\bar{\Phi}_0 = \bar{U}\bar{\Sigma}\bar{U}^T$

$\bar{U} \in \mathbb{R}^{n \times l}$

$\bar{\Phi}_0 > 0$

Inertia matrix: $\bar{\Phi}_0^{-1} = \bar{U}\bar{\Sigma}^{-1}\bar{U}^T$

Kinetic energy: $T = \frac{1}{2}\delta\dot{\theta}^T\bar{\Phi}_0^{-1}\delta\dot{\theta}$

Approximating Kinetic Energy

$T = \frac{1}{2}\delta\dot{\theta}^T\bar{U}\bar{\Sigma}^{-1}\bar{U}^T\delta\dot{\theta}$

Singular values

$\frac{1}{\sigma_l} \geq \frac{1}{\sigma_{l-1}} \geq \cdots \geq \frac{1}{\sigma_{l-k+1}} \geq \cdots \geq \frac{1}{\sigma_1}$

Top k singular values

$\hat{\Phi}_0^{-1} = \hat{U}\hat{\Sigma}^{-1}\hat{U}^T$ $\quad \hat{U} \in \mathbb{R}^{n \times k}$

$\hat{T} = \frac{1}{2}\delta\dot{\theta}^T\hat{U}\hat{\Sigma}^{-1}\hat{U}^T\delta\dot{\theta}$

State mapping $\dot{q} = \hat{U}^T\delta\dot{\theta}$ or $\delta\dot{\theta} = \hat{U}\dot{q}$

$\hat{T} = \frac{1}{2}\dot{q}^T\hat{\Sigma}^{-1}\dot{q}$ → Inertia matrix $\hat{M} = \hat{\Sigma}^{-1}$

Input Mapping

Consider the power applied by the joint torques

- Full model: $\delta\dot{\theta}^T S^T\tau = \dot{q}^T\hat{U}^T S^T\delta\tau$ (State mapping)
- Simplified model: $\dot{q}^T u$

→ $u = \hat{U}^T S^T \delta\tau$

Computing $\hat{G}$ and $\hat{C}$

$\hat{M}\ddot{q} + \hat{C}\dot{q} + \hat{G}q = u$

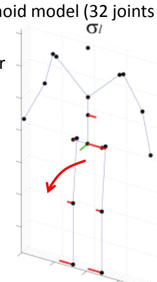
- Compute inverse dynamics at many static poses $\theta_0 + \Delta\theta_i$ and compute joint torque $\tau_i = \tau_0 + \delta\tau_i$
- $q = \hat{U}^T\Delta\theta_i, u = \hat{U}^T S^T\delta\tau_i$
  → $\hat{G}\hat{U}^T\Delta\theta_i = \hat{U}^T S^T\delta\tau_i$
- Compute the $\hat{G}$ that best fits the samples

Example

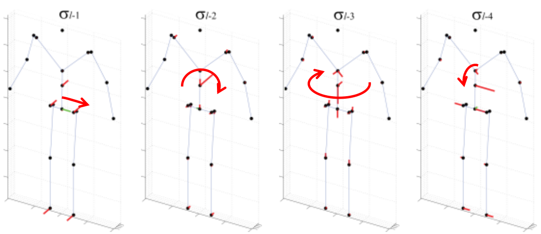
38 DOF humanoid model (32 joints + 6 DOF pelvis)

σ_l

1st singular vector



Example



2nd 3rd 4th 5th

Control

- Input (joint torque) mapping: $u = \hat{U}^T S^T \delta\tau$
  - Number of inputs: k (simplified) < $n - 6$ (original)
- Optimization
  - Cost function: input mapping error, desired joint trajectories
  - Constraints: dynamic, friction, and COP

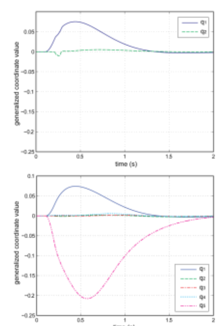
Simulation

- 2 or 5 DOF models for each task
  - 2 DOF models don't work in some tasks, possibly due to unmodeled dynamics
  - Various nominal poses
- Balance controller: linear quadratic regulator (LQR)

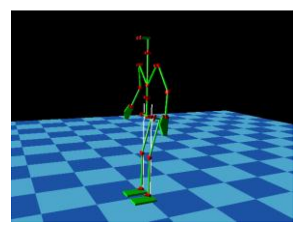
$$J = \int_0^\infty \left(\begin{pmatrix} q^T & \dot{q}^T \end{pmatrix} \begin{pmatrix} Q_1 & 0 \\ 0 & Q_2 \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} + u^T R u \right) dt$$

$$Q_1 = 100, Q_2 = 1 \times 10^{-3}, R = 1 \times 10^{-2}$$

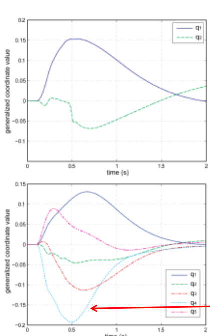
Simulation: Pushed from Back



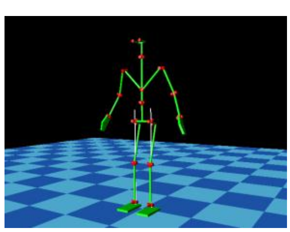
200N for 0.1s



Pushed at Right Shoulder

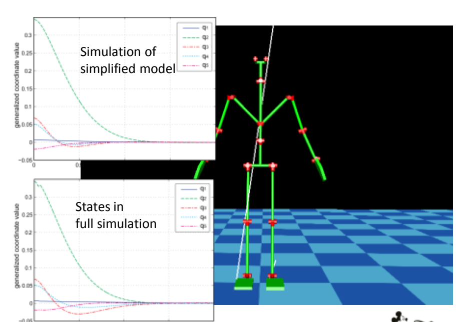


2DOF model cannot maintain balance



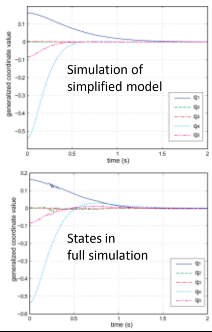
Twist not modeled in the 2DOF model

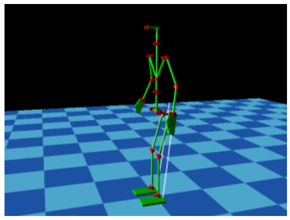
Changing to a New Pose



Changing to a New Pose

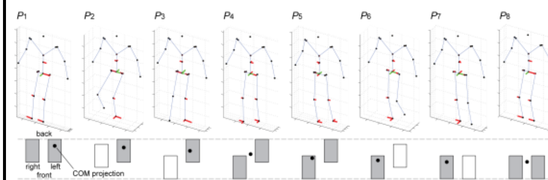
2DOF model cannot maintain balance





Disney Research, Pittsburgh

Combine Multiple Controllers



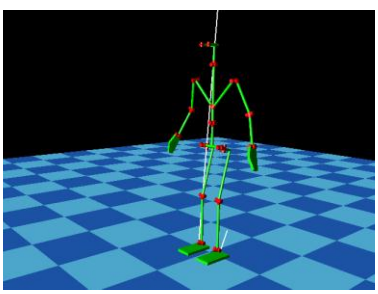
Manually designed nominal poses and contact constraints

No contact

Contact

Disney Research, Pittsburgh

Combine Multiple Controllers



Disney Research, Pittsburgh

Summary

- Automatically generate simplified models of humanoid robots with contacts given:
  - Nominal configuration
  - Contact constraints
  - Reduced DOF
- State/input mapping
  - State mapping uses the same code for any model
  - Input mapping is redundant and allows other control objectives

Disney Research, Pittsburgh

Model Identification

[Yamane 2011]

Disney Research
Pittsburgh

Target Parameters

- Kinematic model: joint angle sensor offsets
  - Potentiometer's zero angle drift
- Inertial model: link mass and center of mass
  - Total weight from CAD model ~60kg
  - Actual weight ~90kg
  - Larger discrepancy than electric robots due to hose and oil
  - Omit moments of inertia

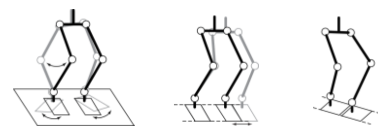
Disney Research, Pittsburgh

Kinematic Parameters

- Existing solutions
  - External measurements (e.g., motion capture): cumbersome to set up, not always available
  - Calibration jig: not enough samples for complex robots

Kinematic Parameters

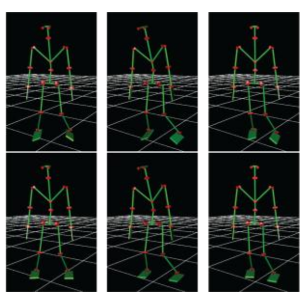
- Our solution
  - IMU for global information
  - Easily enforced kinematic constraints (e.g., flat on floor, parallel feet)



Kinematic Parameters

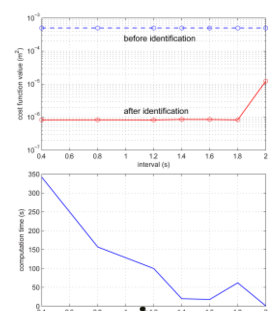
Before

After



Kinematic Parameters

- Cost function error vs. sample interval



Inertial Parameters

- Equation of motion with inertial parameters ϕ:

$$M(\theta,\phi)\ddot{\theta} + c(\theta,\dot{\theta},\phi) + g(\theta,\phi) = S^T\tau + J_c^T(\theta)f_c$$

$$F(\theta,\dot{\theta},\ddot{\theta},\phi) = S^T\tau + J_c^T(\theta)f_c$$

- F is a linear function of ϕ [Mayeda et al. 1984]

$$F(\theta,\dot{\theta},\ddot{\theta},\phi) = A(\theta,\dot{\theta},\ddot{\theta})\phi$$

Regressor

Identification

- Existing solutions
  - Collect samples of $\theta_k, \dot{\theta}_k, \ddot{\theta}_k, \tau_k, f_{ck}$
  - Concatenate all samples: $\bar{A}\phi = \bar{F} = \bar{S}^T\bar{\tau} + \bar{J}_c^T\bar{f}_c$

  $\rightarrow \phi = \bar{A}^{\#}F$

- Issues with humanoid robots
  - All τ and f_c may not be available
  - Difficult to derive a symbolic representation
  - Difficult to obtain enough excitation

Partial Force Measurement

[Mistry et al. 2009]

- Generalization of [Ayusawa et al. 2009]: identification only from contact force measurement
- Divide into available and unavailable measurements

$$\bar{F} = \bar{H}_a^T \begin{pmatrix} \bar{\tau}_a \\ \bar{f}_{ca} \end{pmatrix} + \bar{H}_u^T \begin{pmatrix} \bar{\tau}_u \\ \bar{f}_{cu} \end{pmatrix}$$

- $\bar{N}_u$: null space basis of $\bar{H}_u^T$

$$\bar{F}_N = \bar{N}_u \bar{F} = \bar{N}_u \bar{H}_a^T \begin{pmatrix} \bar{\tau}_a \\ \bar{f}_{ca} \end{pmatrix}$$

- $\bar{N}_u \bar{A} \phi = \bar{F}_N$: only includes measurable forces

Disney Research, Pittsburgh

Computing the Regressor

- Strictly identifiable ϕ is a linear combination of actual inertial parameters (mass, local center of mass)
- Difficult to obtain ϕ for humanoid robots due to complexity
- We use standard inertial parameters instead and compute the regressor by numerical inverse dynamics and finite difference of inertial parameters

Disney Research, Pittsburgh

Estimation

- Difficult to obtain enough excitation
  - Robot needs to balance
  - Results in invalid parameters (negative mass etc.)
- Ideas
  - Ignore unreliable parameter space: omit small singular values of A and use pseudo inverse ("least square")
  - Prevent inconsistent results: gradient-based optimization with lower and upper bounds ("gradient")

Disney Research, Pittsburgh

Results

- Test data: 3 trials for "teapot" tracking; 1 for identification, 2 for cross-validation
- Torque information is not available for position-controlled joints (upper body)
- Parameters: with (LS) and without (L) symmetry constraint

Disney Research, Pittsburgh

Results

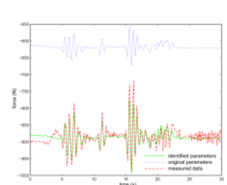
Force estimation errors

	parameter set	L			LS		
	number of parameters	92			56		
	condition number of regressor	1.38×10^5			3.81×10^4		
	maximum condition number	1×10^6	1×10^3	1×10^2	1×10^6	1×10^3	1×10^2
least square	direct validation	n/a	n/a	1.56×10^5	n/a	n/a	1.97×10^5
	cross validation (1)	n/a	n/a	7.13×10^5	n/a	n/a	5.77×10^5
	cross validation (2)	n/a	n/a	7.93×10^5	n/a	n/a	9.47×10^5
gradient	direct validation	1.52×10^5	1.52×10^5	1.52×10^5	1.99×10^5	1.99×10^5	1.99×10^5
	cross validation (1)	5.28×10^5	5.28×10^5	5.29×10^5	4.51×10^5	4.51×10^5	4.50×10^5
	cross validation (2)	7.29×10^5	7.29×10^5	7.30×10^5	5.81×10^5	5.81×10^5	5.84×10^5

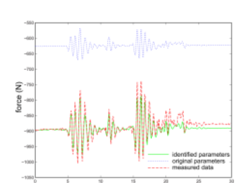
n/a: Resulted in negative mass

Disney Research, Pittsburgh

Results



Direct validation



Cross validation

Vertical force of base

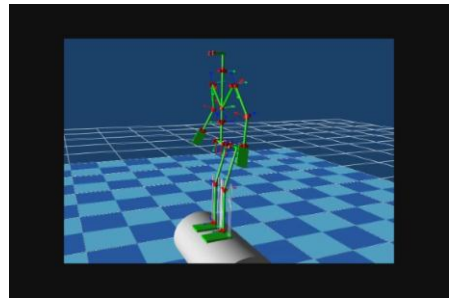
Disney Research, Pittsburgh

Humanoid Robot Control in Dynamic Environments

[Zheng and Yamane Humanoids 2011]

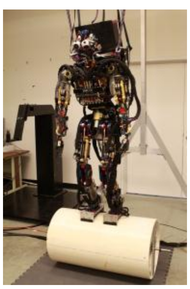
Disney Research Pittsburgh

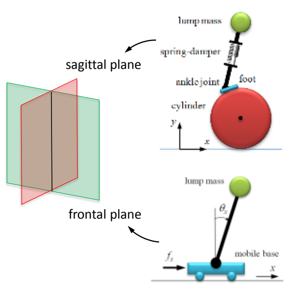
Balancing Control on a Cylinder



Disney Research, Pittsburgh

Motion Decomposition

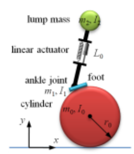




3-D motion → two 2-D motions

Disney Research, Pittsburgh

Simplified Model in Sagittal Plane

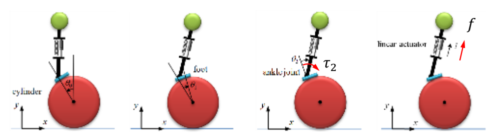


Configuration $\theta = (\theta_0 \quad \theta_1 \quad \theta_2 \quad l)^T$

Inputs $\tau = (0 \quad 0 \quad \tau_2 \quad f)^T$

ankle torque τ_2

actuator force f



Disney Research, Pittsburgh

Equation of Motion in Sagittal Plane

Linearized equation of motion around $\theta = 0$

$$M\ddot{\theta} + G\theta = \tau$$

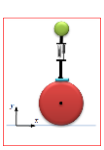
State-space equation

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

where $x = (\theta^T \quad \dot{\theta}^T)^T$

$u = (\tau_2 \quad f)^T$

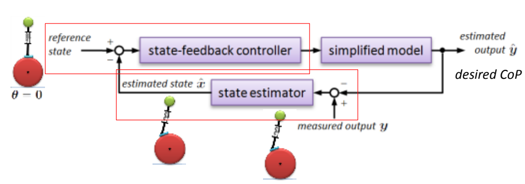
$y = \theta$



$x = 0$ is only equilibrium state since A has full rank

Disney Research, Pittsburgh

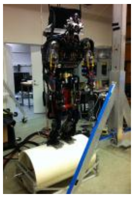
Balance Controller

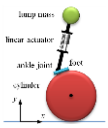


Disney Research, Pittsburgh

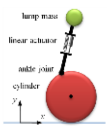
Flat Feet vs. Geta Feet

Sarcos robot with flat feet





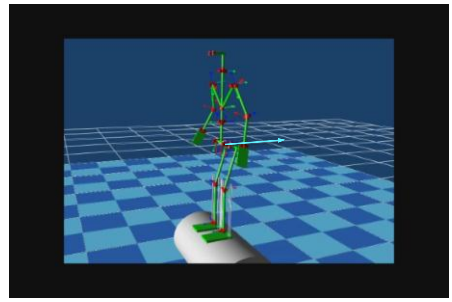
Sarcos robot with geta feet



geta shoes

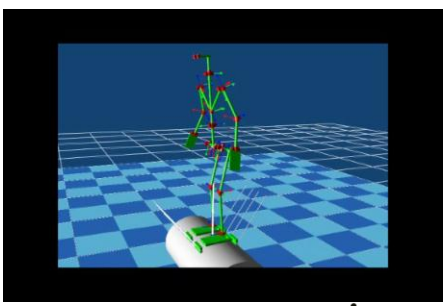
Disney Research, Pittsburgh

Simulation Result with Flat Feet



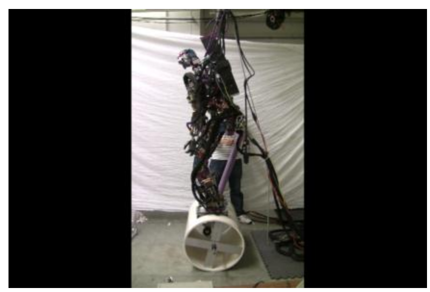
Disney Research, Pittsburgh

Simulation Result with Geta Feet



Disney Research, Pittsburgh

First Hardware Trial



Disney Research, Pittsburgh

Discussion

- Accurately estimating model parameters is difficult
- Articulated rigid body models don't capture many aspects of humanoid robot dynamics
  - Joint friction, backlash
  - Link deformation
- What is the right level of detail for
  - Control
  - Simulation

Disney Research, Pittsburgh