Smooth discrete feedback control of walking robots

An intermediate between fully passive and high bandwidth feedback control

Pranav A. Bhounsule
(work with Andy Ruina)

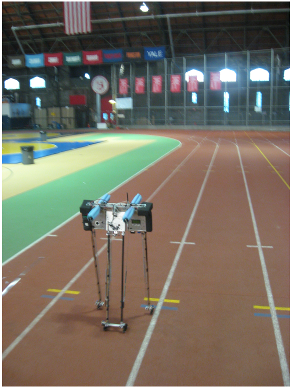
Biorobotics and Locomotion Laboratory
Cornell University, Ithaca, NY 14853, USA

World Congress on Biomechanics, 2010

Funding: NSF

What is interesting about this control architecture?

What is interesting about this control architecture?



Cornell Ranger walked 14.3 miles (=23 kms.) using only 3 cents worth of electricity!

State of the art in walking robots

High Bandwidth Feedback Control

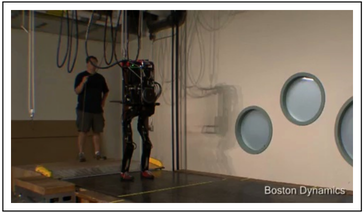


Figure: PETMAN from Boston Dynamics

- Robust, Versatile.
- Energy Inefficient.

Passive Dynamic Walkers

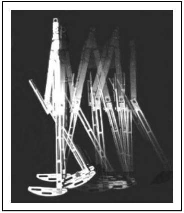


Figure: Passive Kneed Walker

- Energy efficient
- Not Robust.

Is there an intermediate approach?

- can be made close to optimal feedback control.
- can be made robust.
- can be made dumb.

The controller architecture

1. Model the system.
    - equations of motion
    - system identification

The controller architecture

1. Model the system.
   - equations of motion
   - system identification
2. Find open loop optimal control policy
   - Torques as a function of time.
   - Few hundred parameters.

The controller architecture

1. Model the system.
   - equations of motion
   - system identification
2. Find open loop optimal control policy
   - Torques as a function of time.
   - Few hundred parameters.
3. Find open loop approximate optimal control policy (design choice)
   - Torques as a function of state.
   - Maximize simplicity of representation.
   - Minimize number of parameters.

The controller architecture

1. Model the system.
   - equations of motion
   - system identification
2. Find open loop optimal control policy
   - Torques as a function of time.
   - Few hundred parameters.
3. Find open loop approximate optimal control policy (design choice)
   - Torques as a function of state.
   - Maximize simplicity of representation.
   - Minimize number of parameters.
4. Stabilize the approximate optimal control policy (details ahead)

Stability is easy

Equation of motion

$$\delta x_{n+1} = A\delta x_n + B\delta u_n$$

Output equation

$$\delta z_{n+1} = C\delta x_n + D\delta u_n$$

- n = step number*
- δx_n = differential about nominal value for state vector.
- δu_n = differential about nominal value for control vector.
- δz_n = differential about nominal value for output vector.
- A = Jacobian = $\partial(\delta x_{n+1})/\partial(\delta x_n)$
- B = Sensitivity = $\partial(\delta x_{n+1})/\partial(\delta u_n)$
- C = Jacobian = $\partial(\delta z_{n+1})/\partial(\delta x_n)$
- D = Sensitivity = $\partial(\delta z_{n+1})/\partial(\delta u_n)$

* could be time or state based

Stability is easy

How to get the output equations?

$$\delta z_{n+1} = C\delta x_n + D\delta u_n$$

Stability is easy

How to get the output equations?

$$\delta z_{n+1} = C\delta x_n + D\delta u_n$$

Hold u_n constant, perturb x_n

$$\delta z_{n+1} = C\delta x_n$$

Stability is easy

How to get the output equations?

$$\delta z_{n+1} = C\delta x_n + D\delta u_n$$

Hold x_n constant, perturb u_n

$$\delta z_{n+1} = D\delta u_n$$

Stability is easy

Output Equation

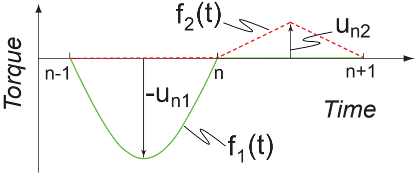
$$\delta z_{n+1} = C\delta x_n + D\delta u_n$$

NOTE:

- δu_n is intermittent.
- δu_n can be made smooth.

e.g.

$$\delta u_n = \begin{bmatrix} u_{n1} f_1(t) \\ u_{n2} f_2(t) \end{bmatrix}$$



Stability is easy

How to use the output equation for control?

For linear control,

$$\delta u_n = K \delta x_n$$

Thus,

$$\delta z_{n+1} = C \delta x_n + D \underbrace{K \delta x_n}_{\delta u_n}$$

Stability is easy

How to use the output equation for control?

For linear control,

$$\delta u_n = K\delta x_n$$

Thus,

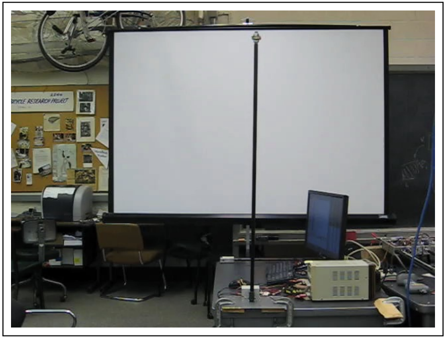
$$\delta z_{n+1} = C\delta x_n + D\underbrace{K\delta x_n}_{\delta u_n}$$

e.g. dead beat control
$(\delta z_{n+1} = 0)$

$$\delta u_n = K\delta x_n = -D^{-1}C\delta x_n$$

Example 1: Inverted Pendulum

Linearization over 1 sec.
i.e. n (seconds) = 1,2,3 ...



Output Equation

$$\overbrace{\delta x_{n+1}}^{state} = C\delta x_n + D\,\underbrace{K\delta x_n}_{\delta u_n}$$

For dead beat control
$\delta x_{n+1} = 0$

$$\delta u_n = K\delta x_n = -D^{-1}C\delta x_n$$

Example 2: Walking Robot

Linearization about upright position.
and n (step number) = 1,2,3 ...

Output Equation

$$\overbrace{\delta E_{n+1}}^{\text{Mech. Energy}} = F\delta x_n + G \underbrace{\overbrace{K\delta x_n}^{\text{Hip Torque}}}_{\delta u_n}$$

For dead beat control $\delta E_{n+1} = 0$

$$\delta u_n = K\delta x_n = -D^{-1}C\delta x_n$$

Walk Statistics

Distance:

14.3 mi (=23 km).

Time:

10 hrs, 40 min, 48 sec

Speed:

1.34 mi/hr (=2.15 km/hr)

No. of Steps:

65,185.

Power:

24.5 watt

Energy:

262 watt hours

COT = Energy/(Distance × Weight):

0.49

Thank You

Walk Statistics

Distance:
: 14.3 mi (=23 km).

Time:
: 10 hrs, 40 min, 48 sec

Speed:
: 1.34 mi/hr (=2.15 km/hr)

No. of Steps:
: 65,185.

Power:
: 24.5 watt

Energy:
: 262 watt hours

COT = Energy/(Distance × Weight):
: 0.49