Limit Cycle Analysis of a Passive Dynamic Walker

Final Project Presentation

by

Thomas Johnston

UC DAVIS
UNIVERSITY OF CALIFORNIA

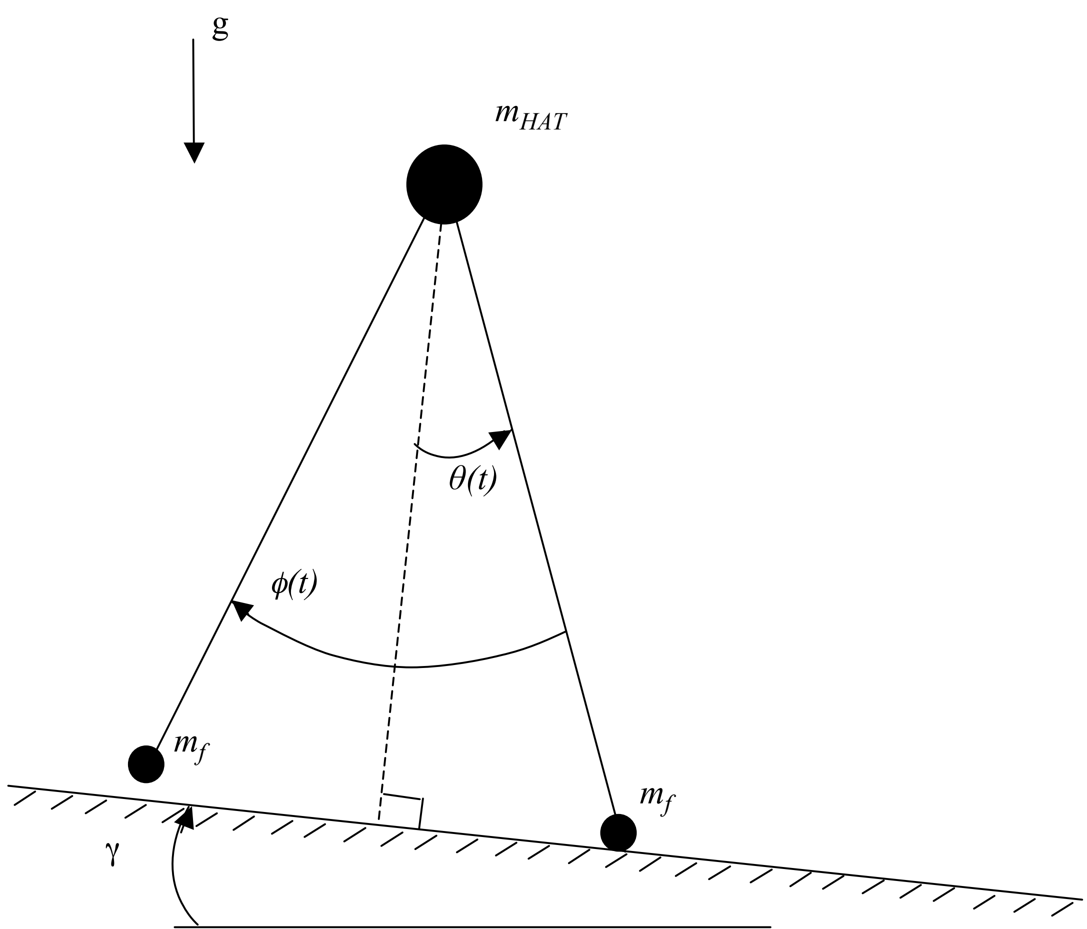
“2-D” Passive Dynamic Walker

http://www.3me.tudelft.nl/live/pagina.jsp?id=c4fa06f1-b767-4a67-a19e-ea3356400f06&lang=en&binary=/img/picsimple_no_springs.gif

3-D Robot with Knees & Arms

http://ruina.tam.cornell.edu/research/topics/locomotion_and_robotics/papers/3d_passive_dynamic/from_angle.mpg

Methods (1) – Define Model



Garcia M, Chatterjee A, Ruina A, Coleman M. The simplest walking model: Stability, complexity, and scaling. ASME Journal of Biomechanical Engineering 1998;120(2):281-288.

Methods (2) – Create StrideMap

StrideMap: $\vec{\mathbf{x}}_{n+1} = f(\vec{\mathbf{x}}_n)$

- EOM (swing phase)
- Footstrike
- Post-impact state (= transition equations)

StrideMap Error: $g(\vec{\mathbf{x}}_n) = f(\vec{\mathbf{x}}_n) - \vec{\mathbf{x}}_n = \vec{\mathbf{x}}_{n+1} - \vec{\mathbf{x}}_n$

Methods (3) - Multidimensional NRA part 1

$$F_i(x_1, x_2, \ldots, x_N) = 0 \qquad i = 1, 2, \ldots, N. \tag{1}$$

***x* = (*x_1* , *x_2* , … , *x_n*):** **vector of variables,**

F **: vector of function.**

In the neighborhood of *x*,

F **can be expanded in a Taylor series:**

$$F_i(\mathbf{x} + \delta\mathbf{x}) = F_i(\mathbf{x}) + \sum_{j=1}^{N} \frac{\partial F_i}{\partial x_j} \delta x_j + O(\delta\mathbf{x}^2).$$

$$J_{ij} \equiv \frac{\partial F_i}{\partial x_j}.$$

$$F_i(\mathbf{x} + \delta\mathbf{x}) = F_i(\mathbf{x}) + \sum_{j=1}^{N} J_{ij} \delta x_j + O(\delta\mathbf{x}^2). \tag{2}$$

Methods (3) - Multidimensional NRA part 2

neglecting : $O(\delta \mathbf{x}^2)$ $F_i(\mathbf{x}+\delta\mathbf{x}) = F_i(\mathbf{x}) + \sum_{j=1}^{N} J_{ij}\,\delta x_j$

setting: $F_i(\mathbf{x}+\delta\mathbf{x}) = 0$, $F_i(\mathbf{x}) + \sum_{j=1}^{N} J_{ij}\,\delta x_j = 0$

Unknowns

Coefficients

Right-hand side vector

$$\sum_{j=1}^{N} J_{ij}\,\delta x_j = -F_i(\mathbf{x}) \qquad (3)$$

Eq.(3) is a set of linear equations!

The coefficient matrix { J_{ij} } can be solved by Gaussian Elimination, LU, … methods.

Eq. (3) $\delta\mathbf{x}$ → $\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \delta\mathbf{x}$

"Newton step"

Methods (4) - Stability Analysis

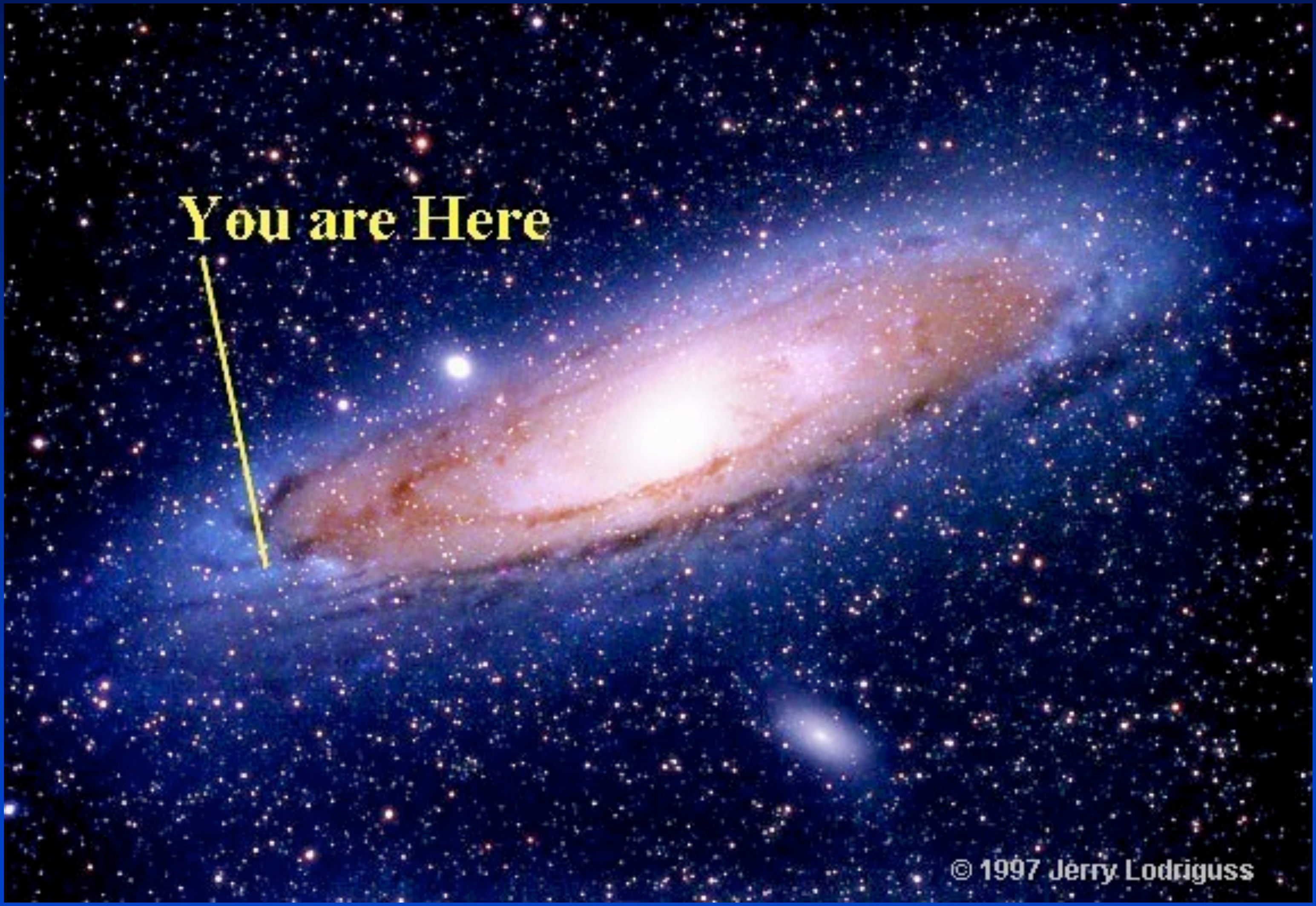
- Numerically approximate the Jacobian and check eigenvalues
  - requires n+1 evaluations of StrideMap, $f(\vec{\mathbf{x}}_n)$

- Adjust model parameters?

Method Summary

- Define model
- Evaluate StrideMap Error: $g(\vec{\mathbf{x}}_n) = f(\vec{\mathbf{x}}_n) - \vec{\mathbf{x}}_n$
  - Integrate ODEs
  - Detect footstrike
  - Post-impact state
- Find fixed point(s) & check stability

Questions?



Reserve Slides

Equations of Motion

$$+ \begin{bmatrix} (\beta \cdot g/L) \cdot [\sin(\theta - \phi - \gamma) - \sin(\theta - \gamma)] - g/L \cdot \sin(\theta - \gamma) \\ (\beta \cdot g/L) \cdot \sin(\theta - \phi - \gamma) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

where, $mf \approx 0 \quad \Rightarrow$

$$\ddot{\theta}(t) - \sin(\theta - \gamma) = 0$$

$$\ddot{\theta}(t) - \ddot{\phi}(t) + \dot{\theta}(t)^2 \sin(\phi) - \cos(\theta - \gamma)\sin(\phi) = 0$$

Transition Equations

- Not full rank!
- Contact condition:
- Swing foot makes no contribution to $\vec{H}_{sys}$