Planar Biped Locomotion: Background & Modeling

Jessy W. Grizzle  Eric R. Westervelt

Outline

• Why is bipedal locomotion a hard problem
• Heuristics and the ZMP principle
• What is RABBIT & why this morphology
• Systematic derivation of planar models
  – Lagrangian dynamics for SS (continuous portion)
  – Rigid impact model for DS (discrete portion)
  – System with impulse effects
• Automating model computations within MATLAB
  – Files available online at :
    www.eecs.umich.edu/~grizzle/CDC2003Workshop/
Why biped locomotion is hard

Inherent difficulties:

1. High DOF system with low DOF task

2. Effectively underactuated [Kajita ’96]
Why biped locomotion is hard

Inherent difficulties:

1. High DOF system with low DOF task
2. Effectively underactuated \[\text{[Kajita '96]}\]
3. Static instability during swing phase
Why biped locomotion is hard

Inherent difficulties:

1. High DOF system with low DOF task
2. Effectively underactuated [Kajita '96]
3. Static instability during swing phase
4. Desire periodic motions that are stable
5. Impacts and/or nontrivial double support phase
Consequence: Impressive Advances in Mechanisms, but status quo in Control Notions

Jogging Johnnie

Control Architecture

Control Block Diagram

ZMP: "Stability" is determined by the trajectory generator and NOT the within-stride feedback loop.

ZMP “Principle”
ZMP Principle (Heuristic)

Minority View: Inherent Complexity does NOT Render Analysis Impossible!

- Carefully develop low level behaviors
- Obtain complex behaviors via composition of simpler behaviors
  - Koditschek (Michigan)
  - Krishnaprasad (Maryland)
  - The present workshop
- See also Workshop M-4: “Motion Description Languages for Multi-Modal Control”, CDC 2003
RABBIT: Simplest Mechanism Capable of Quasi-Anthropomorphic Gait

- Sagittal plane dynamics
- Two legs, knees, torso
- No feet = No ZMP = Need control theory!

ROBEA: Robotique et Entités Artificielles (1997)

- Supported by CNRS and links seven French laboratories
  - Gabriel Abba, Carlos Canudas-de-Wit, C. Chevallereau, ....
- Michigan joined in late 1998 through a sabbatical in Strasbourg; NSF support came in September 2000
- Themes:
  - modeling of underactuated systems, systems with impacts
  - dynamic gaits (fast walking and running)
  - optimal motions
  - control of underactuated & hybrid systems
  - experimentation
- Robot named RABBIT:
  - LAG: Laboratoire Automatique de Grenoble
  - See Carlos Canduas-de-Wit for details!
RABBIT Prototype

4 actuators and 5 links
32 kg mass and 1.425 m tall

Robot Model:  SS + DS = Hybrid

Normal walking:

... SS, DS, SS, DS, ...

SS — Single Support  DS — Double Support
Robot Model: \( SS + DS = \text{Hybrid} \)

Normal walking:

\[ \ldots SS, DS, SS, DS, \ldots \]

- **SS** — Single Support
- **DS** — Double Support

Lagrangian Dynamics

Impact Dynamics

\[ \frac{\partial^2 L}{\partial \dot{q} \partial q} - \frac{\partial L}{\partial q} = \Gamma \]

\[ L = K - V \]

- **\( K \)** = Kinetic Energy
- **\( V \)** = Potential Energy
- **\( \Gamma \)** = Forces & Torques

Typical Form in Robotics

\[ \ddot{D}(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = Bu \]
Computing the Lagrangian

• Basic Assumptions in SS
  – Stance leg acts as a pivot
  – Swing leg is not interacting with the ground
  – Rigid links, rigid joints, no joint friction,…

• Five Degrees of Freedom
  – need 5 generalized coordinates

Define a reference (world) frame and a set of five generalized coordinates.

\[ q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} \]
Computing the Lagrangian

• For each link, assign coordinates to
  – the center of mass
    \[
    \begin{bmatrix}
    x_i^c \\
y_i^c
    \end{bmatrix}
    \]
  – and the angle of the link with respect to the world frame
    \( q_i^g \)

Computing the Lagrangian

• And express these as a function of the generalized coordinates, \( q \)
  – the center of mass
    \[
    \begin{bmatrix}
    x_i^c \\
y_i^c
    \end{bmatrix} = \begin{bmatrix} x_i^c(q) \\ y_i^c(q) \end{bmatrix}
    \]
  – angle of the link with respect to the world frame
    \( q_i^g = q_i^g(q) \)
Computing the Lagrangian

- By the chain rule
  - velocity of center of mass
    \[
    \begin{bmatrix}
    \dot{x}_i^c \\
    \dot{y}_i^c
    \end{bmatrix} = \begin{bmatrix}
    \frac{\partial x_i^c(q)}{\partial q} \\
    \frac{\partial y_i^c(q)}{\partial q}
    \end{bmatrix} \dot{q}
    \]
  - angular velocity of link
    \[
    \dot{\dot{q}}_i^a = \frac{\partial q_i^a(q)}{\partial q} \dot{q}
    \]

Computing the Lagrangian

Kinetic Energy of i-th Link

\[
K_i = \frac{1}{2} m_i \left( (\dot{x}_i^c)^2 + (\dot{y}_i^c)^2 \right) + \frac{1}{2} I_i (\dot{q}_i^a)^2
\]

\[
K_i = \frac{1}{2} q_i^a T D_i(q) \dot{q}
\]

Potential Energy of i-th Link

\[
V_i = m_i g y_i^c
\]

\[
V_i = m_i g y_i^c(q)
\]
Computing the Lagrangian

Total Kinetic Energy

\[ K = \sum_{i=1}^{5} K_i = \frac{1}{2} \dot{q}^T D(q) \dot{q} \]

Total Potential Energy

\[ V = \sum_{i=1}^{5} V_i \]

Lagrangian

\[ L = K - V \]

Dynamic Model in SS

Lagrange’s Equation

\[ \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \Gamma \]

Form of the Equations

\[ D(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = Bu \]

Underactuated: 5 DOF & 4 Controls
A Property of the SS Model

Coordinates: 4 relative and 1 absolute

\[ \frac{d}{dt} \frac{\partial L}{\partial q_k} - \frac{\partial L}{\partial \dot{q}_k} = \begin{cases} u_k & k = 1, \ldots, 4 \\ 0 & k = 5 \end{cases} \]

\[ \frac{\partial K}{\partial \dot{q}_5} = 0 \quad \text{cyclic coordinate} \]

Angular momentum about stance leg

\[ \sigma = \frac{\partial L}{\partial q_5} \]

\[ \sigma = \sum_{k=1}^5 d_{5,k}(q_1, \ldots, q_4) \dot{q}_k \]

\[ \sigma = -\frac{\partial V}{\partial q_5}(q) \]

Alternate choice of the absolute coordinate
Robot Model: \( SS + DS = \text{Hybrid} \)

Normal walking:

\[ SS_{i-1}, DS_{i}, SS_{i}, DS_{i}, \ldots \]

**SS** — Single Support  \hspace{2cm} **DS** — Double Support

\[
D(q) \ddot{q} + C(q, \dot{q}, \dot{q}) + G(q) = F(u)
\]

- joint angles & velocities
- motor torques

Terms in the Model... Oh my!

<table>
<thead>
<tr>
<th>Term</th>
<th>Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>(D_{x,1}(q))</td>
<td>(-2q_1M_1L_1 \cos(q_1 - q_2 - q_3 - q_4))</td>
</tr>
<tr>
<td>(D_{x,2}(q))</td>
<td>(I_1 I_f \cos(q_1 - q_2 - q_3 - q_4) + 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,3}(q))</td>
<td>(I_1 + I_f + I_2 + 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,4}(q))</td>
<td>(-2q_1 M_1 L_1 \cos(q_1) - 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,5}(q))</td>
<td>(2q_1 M_1 L_1 \cos(q_1) - 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,6}(q))</td>
<td>(2q_1 M_1 L_1 \cos(q_1) - 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,7}(q))</td>
<td>(2q_1 M_1 L_1 \cos(q_1) - 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
<tr>
<td>(D_{x,8}(q))</td>
<td>(2q_1 M_1 L_1 \cos(q_1) - 2q_1 M_1 L_1 \cos(q_1))</td>
</tr>
</tbody>
</table>

---

15
Using MATLAB to Obtain the Model

- Very convenient to derive model, compute control laws, and perform simulations in a common environment:
  - we have been using MATLAB
  - model derived using SYMBOLIC TOOLBOX
  - m-files for ODE45 or Simulink are automatically generated from the symbolic computations
  - relevant files can be downloaded at

  www.eecs.umich.edu/~grizzle/CDC2003Workshop/

Robot Model: \( SS + DS = \text{Hybrid} \)

Normal walking:

\[ \ldots \text{SS, DS, SS, DS, } \ldots \]

SS — Single Support  

\[ D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = B\ddot{q} \]

DS — Double Support

Impact Dynamics

Impact Dynamics
**DS: Rigid Impact Model**

- Basic Assumptions in DS
  - Impact instantaneous
    - \( \Rightarrow \) impulsive contact forces
  - No rebound, no slip at impact
  - Former stance leg releases freely and does not interact with the ground
    - Yields conservation of angular momentum about impact point
    - Positions are continuous but instantaneous jump in the velocities
  - Relabel coordinates so that previous SS model can be reused

\[ \mathcal{S} = \{ q \mid y_2(q) = 0, x_2(q) > 0 \} \quad \text{Impact Surface} \]

\[ q^+ = Rq^- \quad \text{(relabel states)} \]
[- is just before impact and + is just after]

\[ \dot{q}^+ = \Delta q^\dot{}^- \quad \text{(jump in velocities + relabeling)} \]

\[ x^+ = \begin{bmatrix} q^+ \end{bmatrix} = \begin{bmatrix} Rq^- \end{bmatrix} = \Delta(x^-) \quad \text{(overall effect of impact)} \]
DS: Rigid Impact Model

- Derivation is provided in handout, but is NOT covered in oral presentation
- See web site for MATLAB symbolic code

www.eecs.umich.edu/~grizzle/CDC2003Workshop/

DS: Rigid Impact Model


**DS: Rigid Impact Model**

Need to use 7 DOF model. Introduce Cartesian coordinates of stance leg end
\[
\begin{bmatrix}
  x_1 \\
  y_1 
\end{bmatrix}
\]
and define
\[
q_e = \begin{bmatrix}
  q \\
  x_1 \\
  y_1 
\end{bmatrix}
\]

Use Lagrange to Compute 7DOF Model
\[
D_e(q_e)\ddot{q}_e + C_e(q_e, \dot{q}_e)\dot{q}_e + G_e(q_e) = \Gamma_e
\]
**DS: Rigid Impact Model**

\[ \dot{q}_e^- = \text{velocity just before impact} \]

\[ \dot{q}_e^+ = \text{velocity just after impact} \]

\[ F_e = \begin{bmatrix} F_T \\ F_N \end{bmatrix} = \text{impact force (intensities)} \]

\[ E = \begin{bmatrix} \frac{\partial x_2(q_e)}{\partial q_e} \\ \frac{\partial g_2(q_e)}{\partial q_e} \end{bmatrix}^T \]

\[ \Gamma_e = B_e u + EF_e \]

**Assumption:** Impact forces are impulses while motor torques are not.

---

**DS: Rigid Impact Model**

Conservation of momentum about impact point

\[ D_e (\dot{q}_e^+ - \dot{q}_e^-) = EF_e \]

No rebound nor slip at impact of swing leg end

\[ E^T \dot{q}_e^+ = 0 \]

Prior stance leg acted as pivot

\[ \dot{q}_e^- = \begin{bmatrix} \dot{q}^- \\ 0 \\ 0 \end{bmatrix} \]
**DS: Rigid Impact Model**

Solve equations on previous page and then remove Cartesian components to obtain

\[
\dot{q}^+ = \Delta \dot{q}^-
\]

\[
F_c = A(q)\dot{q}^-
\]

**DS: Rigid Impact Model**

In order to re-use the previous SS model at the next step, must re-label the coordinates

\[
q^+ = Rq^-
\]

\[
q^+ = R\Delta \dot{q}^- = \Delta \dot{q}^-
\]

Final result is:

\[
x^+ = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}^+ = \begin{bmatrix} Rq \\ \Delta \dot{q}^- \end{bmatrix} = \Delta (x^-)
\]
Robot Model: $SS + DS = \text{Hybrid}$

Normal walking:

\[ u \rightarrow \begin{array}{c} \vdots \\ \text{SS, DS, SS, DS, \ldots} \end{array} \rightarrow x \]

$SS$ — Single Support $DS$ — Double Support

\[
\dot{x} = f(x) + g(x)u \\
x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}
\]

\[ x^+ = \Delta(x^-) \]

10 differential equations & impact map

(Hybrid Model)
Robot Model: $SS + DS = Hybrid$

Normal walking:

$\ldots SS, DS, SS, DS, \ldots$

$SS$ — Single Support  $DS$ — Double Support

10 differential equations & impact map

$\phi(x) = 0$

$\dot{x} = f(x) + g(x)u$

$x^+ = \Delta(x^-)$

Model as System with Impulse Effects

$\sum : \begin{cases} 
\dot{x}(t) = f(x(t)) & x^-(t) \not\in S \\
x^+(t) = \Delta(x^-(t)) & x^-(t) \in S 
\end{cases}$

$x^+(t) := \lim_{\tau \downarrow t} x(\tau) \quad x^-(t) := \lim_{\tau \uparrow t} x(\tau)$


$S = \{x \in \mathcal{X} \mid \text{swing leg contacts the ground}\}$

Stable walking ⇔ Asymptotically stable limit cycle (transversal to $S$)
Is the Model Complete?

- Standard friction compensation is used in the final stage of controller implementation.
- Lagrangian model is crucial for inverse pendulum dynamics w.r.t. the support leg.

Conclusions: Background & Modeling

- Bipedal locomotion is a hard problem due to high DOF, impacts, periodic orbits, and underactuation.
- Models are necessarily hybrid in nature:
  - Lagrangian dynamics for SS (continuous portion)
  - Rigid impact model for DS (discrete portion)
  - System with impulse effects
- RABBIT conceived to enhance understanding.
- Systematic derivation of planar model presented.
- MATLAB tools shared with participants.