CSE 888.14 Advanced Computer Animation
Final Presentation

Topic: Locomotion

Kang-che Lee

2009 Fall
Locomotion

• How a character moves from place to place.

• Optimizing walking controllers.

Optimizing walking controllers (Siggraph Asia 2009)

• Describes a method for optimizing the parameters of a physics-based controller for full-body, 3D walking.

• Observed how to choose critical parameters for tuning to achieve better walking control and reasonable walking style.

• Resulting gaits exhibit key properties of natural walking, for example, energy efficiency.
  – toe-off, heel-strike

• The system does not require any motion capture data.
Optimizing walking controllers
(Siggraph Asia 2009)

• Optimizing a controller involves
  – searching for control parameters
  – a start state that together produce good character simulations.

• Objective Function
  – weighted sum of several terms
  – features of human walking
    – constraints: User gait, Required gait, Head and body,
    – Efficiency and power terms

• A modified version of the SIMBICON controller is optimized.
Character Model and Controllers

• Character Model:
  • 30 internal degrees-of-freedom (DOFs)
  • toe blocks, which are connected to the feet by hinge joints
    – provide more flexibility during landing and ankle toe-off
• Single-state controller:
• Torque for each joint DOF (drive each joint to its desired local angle)

$$\tau = k_p (\theta_d - \theta) + k_d \dot{\theta},$$

gain and damping coefficients ($k_p$, $k_d$)

- target angle
- angular velocity
- current joint angle

Department of Computer Science and Engineering
Character Model and Controllers

• Single-state controller:
• target angle of the arm DOF (in the sagittal plane), allows the model to synchronize arm swing with the legs

\[ \theta_{larm} = \alpha_{arm}(\theta_{rhip} - \theta_{lhip}), \quad \theta_{rarm} = -\theta_{larm} \]

scale factor.
Character Model and Controllers

• State machine and transitions

<table>
<thead>
<tr>
<th>State #</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left Foot</td>
<td>Swing</td>
<td>Swing</td>
<td>Contact</td>
<td>Stance</td>
</tr>
<tr>
<td>Right Foot</td>
<td>Stance</td>
<td>Heel-off</td>
<td>Swing</td>
<td>Swing</td>
</tr>
</tbody>
</table>

• The transition to state 1: when the signed horizontal distance (in the sagittal plane) between the center-of-mass (COM) of the body and the ankle of the stance foot exceeds a threshold
  – motivated by our observations of when stance ankle push-off appears to occur
  – differs from SIMBICON which uses a time-based transition.
Character Model and Controllers

• State machine and transitions

<table>
<thead>
<tr>
<th>State #</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left Foot</td>
<td>Swing</td>
<td>Swing</td>
<td>Contact</td>
<td>Stance</td>
</tr>
<tr>
<td>Right Foot</td>
<td>Stance</td>
<td>Heel-off</td>
<td>Swing</td>
<td>Swing</td>
</tr>
</tbody>
</table>

• During state 1, the swing leg prepares for landing, and the stance ankle push-off begins.

• Transition to state 2 occurs when the swing foot makes contact with the ground.

• States 2 and 3 are left/right reflected versions of states 0 and 1, with mirrored parameters.
Character Model and Controllers

• Start state:
  – manually initialized to when the left leg is in the middle of its swing phase, prior to the transition from state 0 to state 1.

<table>
<thead>
<tr>
<th>Start State</th>
<th>q_0</th>
<th>\dot{q}_0</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{glo}bpos_x</td>
<td>free</td>
<td>1.3</td>
</tr>
<tr>
<td>lhip_x</td>
<td>-0.4</td>
<td>0.3</td>
</tr>
<tr>
<td>lknee_y</td>
<td>1.35</td>
<td>0.1</td>
</tr>
<tr>
<td>lankle_y</td>
<td>0.35</td>
<td>-15</td>
</tr>
<tr>
<td>rhip_y</td>
<td>-0.4</td>
<td>1.0</td>
</tr>
<tr>
<td>rknee_y</td>
<td>0.6</td>
<td>2.0</td>
</tr>
<tr>
<td>rankle_y</td>
<td>-0.2</td>
<td>-9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
</tr>
<tr>
<td>k_p</td>
</tr>
<tr>
<td>back_x</td>
</tr>
<tr>
<td>rhip_y</td>
</tr>
<tr>
<td>lknee_y</td>
</tr>
<tr>
<td>lankle_x</td>
</tr>
<tr>
<td>lankle_y</td>
</tr>
<tr>
<td>rhip_y</td>
</tr>
<tr>
<td>rknee_y</td>
</tr>
<tr>
<td>rtoe_y</td>
</tr>
<tr>
<td>ltoe_y</td>
</tr>
</tbody>
</table>

• Fix DOFs and other parameters that are unlikely to contribute to the task:
  – joint DOFs that rotate with respect to the local x and z axis are fixed to zero, back joint is an exception (for trunk rotation in the gait).
Controller Optimization

• Objective function:
  – evaluates simulations of duration 10 seconds (T simulation time-steps).
  – a weighted combination of terms motivated by task constraints and biomechanical features of human walking.

• thresholded quadratic

\[
Q(d; \epsilon) = \begin{cases} 
  d^2, & \text{if } |d| > \epsilon \\
  0, & \text{otherwise}
\end{cases}
\]
Controller Optimization

• User gait constraints
  \[ E_{user} = Q(v_x - \hat{v}_x; \epsilon_{vel}) + Q(s - \hat{s}; \epsilon_{step}) \]
  - Encourages \( v_y \) and \( v_z \) to be small, and the start velocity \( V_0x \) to be similar to the average x velocity of the simulated motion. (optimize for walking in the positive x).
  - \( E_{vel} = Q(v_y; \epsilon_{vel}) + \lambda_{vel} [Q(v_{0x} - v_x; \epsilon_{vel}) + Q(v_z; 2\epsilon_{vel})] \)

• Required gait constraints
  - Encourages left-right symmetric timing of the controller. (requires left and right strides to have approximately the same duration.)
  \[ E_{sym} = Q(\Delta t_0 - \Delta t_2; \epsilon_t) + Q(\Delta t_1 - \Delta t_3; \epsilon_t) \]
Controller Optimization

• Head and body constraints
  – prevent unnatural arm swing, where the arms and legs are badly out of phase.

  \[
  \text{avg}L = \frac{1}{T} \sum_{t=1}^{T} \hat{L}_t^2,
  \]

  \[
  E_{\text{ang}} = Q(\sqrt{\text{avg}L}; \sqrt{\epsilon_{\text{ang}}}),
  \]

  \[\hat{L}_t\] is the derivative of the normalized angular momentum about the COM in the vertical direction

  – stabilize the head motion: the lateral and vertical motions of the head are typically smooth with small amplitudes ( helps to stabilize the visual and vestibular systems )

  \[
  E_{\text{head}} = Q(\sqrt{\sigma_{\text{head}}}; \sqrt{\epsilon_{\text{head}}}) + \frac{\lambda_{\text{head}}}{T} \sum_{t=1}^{T} \text{orient}_t,
  \]
Controller Optimization

• Efficiency and power terms
  – human walking is powered more by the ankle than the hip

• Complete objective

**Complete objective.** The complete objective function for walking is given by

\[ E = \sum_{s} w_s E_s, \]  

where \( s \in \{ \text{user}, \text{vel}, \text{land}, \text{fail}, \text{sym}, \text{ang}, \text{head}, \text{power}, \text{ratio} \} \).

We use the following parameters for all experiments: \( w_{\text{user}} = 100, w_{\text{vel}} = 100, w_{\text{land}} = 1.2, w_{\text{fail}} = 120000, w_{\text{sym}} = 100, w_{\text{ang}} = 10, w_{\text{head}} = 100, w_{\text{power}} = 10^{-5} (70/\text{mass}), w_{\text{ratio}} = 1, \lambda_{\text{vel}} = 0.01, \lambda_{\text{head}} = 0.012, \epsilon_{\text{vel}} = 0.025 \text{ m/s}, \epsilon_{\text{step}} = 0.025 \text{ m}, \epsilon_{t} = 0.025 \text{ s}, \epsilon_{\text{ang}} = 0.05/\text{s}^2, \epsilon_{\text{head}} = 0.1 \text{ m/s}. \)
Figure 3: Top: Optimization of “short” (Figure 8(bottom)) walking in 1.0 m/s. Bottom: Optimization without $E_{ratio}$. The lack of the power ratio term leads to a semi-crouching style.
Optimizing walking controllers
(Siggraph Asia 2009)

• Number of limitations
  – requires an expensive optimization procedure, and depends on a reasonable initialization
  – anticipate that it may be possible to learn the parameters from mocap data.
Thank you!