Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2010)
M. Otaduy and Z. Popovic (Editors)

Goal-Directed Stepping with Momentum Control

Chun-Chih Wu† and Victor Zordan

University of California, Riverside

Abstract

This paper proposes a technique for animating simulated characters to perform controlled steps. The desired step is controlled by high-level goals, namely step position and step duration. These stepping goals guide the desired time-varying values for the center of mass and the stepping foot which in turn lead to objectives dictating the desired changes in momentum and joint angles over the duration of the step. Our approach employs a multiobjective optimization to solve for joint accelerations from the objectives and uses inverse dynamics to compute joint torques. Our approach can guide a character with purposeful, directable steps for controlling careful navigation of the character's position and orientation. In addition, the same system can be used to create protective steps to prevent falling as a reaction to a disturbance. A novel supervisory routine automatically chooses when and where to step based on an analysis of the momentum conditions for the character. We contrast this approach to previous methods for step recovery using the inverted pendulum.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation; I.6.8 [Simulation and Modeling]: Types of Simulation—Reactive responses

1. Introduction

Creating controllable responsive characters is a challenging open problem in computer animation and is essential for real-time applications such as electronic games. Physically based simulation holds promise for animating realistic characters in interactive environments. Through simulation, the interplay between characters, objects, and their surroundings can be generated automatically by constraining the motion to follow the dynamic equations of motion. However, developing robust and flexible controllers for simulated characters remains a difficult problem. In this paper, we present a controller which allows characters to step in arbitrary directions both voluntarily and as a result of disturbances.

Stepping is a fundamental skill involved in common activities such as walking and full-body maneuvering from foot repositioning. As such, stepping is a critical behavior for applications involving virtual human avatars. For example, characters in electronic games must be able to change their stance and facing direction in response to a player's inputs. We propose a controller to conveniently synthesize a wide range of stepping behaviors. The inputs to our controller are generic task goals, namely step position and duration, which allow us to apply our technique to various situations and different character morphology.

Further, we introduce a hierarchical control approach to direct stepping that employs a novel momentum-based analysis in a supervisory stage to determine both when and where to step. Given the supervisor's selection of stepping

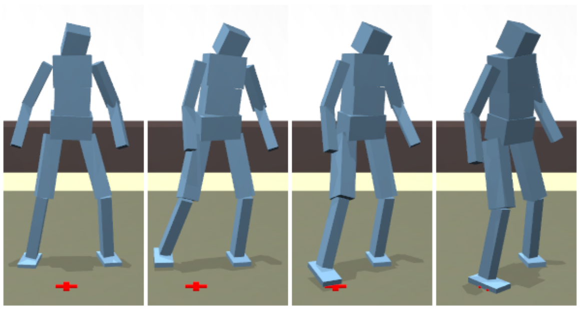

Figure 1: *A sample output for a directed step*

† e-mail: ccwu@cs.ucr.edu

goals, a parameterized curve generator computes desired trajectories for the center of mass (CM) and the stepping foot. These values lead to behavior-specific objectives which guide changes in character's linear momentum and joint angles over the duration of the step. The automatic conversion from high-level goals to low-level control signals has been applied to generate procedural gaits and steps using kinematic models [SM01,WMZ08]. Here we apply the technique to generate controllable steps using a physical model.

Contrasting our approach to inverted pendulum (IP) techniques [PT05,PCDG06], we find that considering both linear and angular momenta is important for step response to disturbance. This finding is supported by biomechanists whom have shown that humans carefully regulate angular momentum in activities such as walking [HP08]. Although the IP plus flywheel was also introduced in [PCDG06] to incorporate angular momentum, our formulation is unique and straightforward as we explain in Section 4.

The main contribution of this paper is our formulation for choosing when and where to step in response to disturbances. Including angular momentum in our formulation improves character's robustness to pushes compared to IP models which only consider linear momentum. A second contribution is our high-level control for the flexible synthesis of goal-directed stepping. We demonstrate the generality of our technique through exploration of two classes of stepping behaviors: directed stepping for navigation and reactive stepping. We also show that our stepping controller can be seamlessly applied to different character morphology, such as a character in handstand.

2. Related Work

Recently, several new motion control approaches have been proposed that take advantage of the realism of data examples while employing simulation to create characters with controllable movements that are both high quality and can interact in a physically responsive manner [ADP07, SKL07, YLvdP07, DAP08, MLPP09, MZS09]. Ours uses a similar framework, most specifically [MZS09]. However, of these efforts the proposed methods have focused on locomotion and standing, but none have focused on the problem of control for stepping. A distinction between these previous efforts and our own work is that a single fixed reference motion is not acceptable for stepping (potentially in any direction at any time).

Other physics-based techniques have been proposed to generate protective steps [FvdPT01, KKI06, JYL09]. Other researchers have focused on closely related biomechanical principles of trip recovery during walking [SCCH09]. Closest to our own [KKI06] chose the desired foot placement using an IP model with its parameters extracted from motion capture data. Also [JYL09] picked the desired step position such that the CM will lie in the center of the support polygon after stepping. The main distinction of our paper is that the choice of when and where to step is automatically computed by our supervisor based on assessment of momenta.

Several roboticists have extended the IP model to account for the change in angular momentum due to disturbances. In particular, the angular momentum pendulum model [KLK04] and the IP plus flywheel [PCDG06] are close models to our own momentum-based supervisor. One difference is that the IP focuses on single stance while our supervisor considers double stance for step recovery. Also, our goal of the step recovery is to place the foot at the proper position to remove all linear and angular momenta induced by a push. This strategy is supported by the study of fall recovery in the biomechanics literature [MMG08].

Data-driven techniques for generating steps in response to unpredicted disturbances without physical simulation require the collection of a large database [AFO05, YPvdP05]. In our approach, we synthesize motion without a reference trajectory through our goal-directed stepping model. Our stepping model is designed based on knowledge extracted about the principle goals of the behavior. We find that controlling CM and swing foot encapsulates many physical attributes of a single step. Previous work [SC92b, SC92a] has the same spirit of CM and end-effector planning.

3. Control Structure

A tiered architecture controls the character to step. At the lowest level we employ the control technique described by Macchietto et al. [MZS09]. That is, a multiobjective optimization solver determines desired joint accelerations while inverse dynamics computes joint torques from the output accelerations to drive a character simulation. In our case, the solver objectives are informed by input signals which are computed once per footstep, automatically, based on the conditions and goals of the specific behavior.

At the core of our controller, the system directs the step through the automatic specification of two straightforward "goal" input signals, one for the CM and one for the swing foot. Their desired trajectories are modeled by two parametric curves based on empirical models built to mimic similar paths extracted from motion capture data [WMZ08]. To convert the input signals to the objectives, we interpret the CM acceleration as linear momentum change and use inverse kinematics (IK) [TGB00] to compute a pose that will achieve the desired foot trajectory.

At the highest level, a user or a supervisory routine directs the high-level characteristics of the behavior, namely the position of the swing foot and the duration of the step. For a reactive step, the supervisor guides the choice of when and where to step based on an analysis of the momentum conditions for the character. We highlight details with respect to the supervisor next.

4. Goal-Directed Stepping

Starting from double stance, the character can take intentional steps by employing the step controller. In the simplest manner, the user can direct the system by specifying a new location for one of the feet. A reasonably large range of foot positions can be controlled. Default timing and stepping height are employed, although these values can also be controlled to change the style of the step. Generating motion in this manner is similar to driving animation with footprints as in [vdP97].

4.1. Reactive Stepping

For reactive stepping, the supervisor automatically determines when to step by assessing character's current momenta. Unlike the linear IP with flywheel [PCDG06], our formulation does not have the constraint of a constant height CM and avoids the simplification of the flywheel. Instead, we use the relationship between full-body momenta changes and control over the CM and center of pressure (CP) [MZS09].

We know that CP can be expressed as a function of the linear momentum change, $\dot{L}$, angular momentum change, $\dot{H}$, and the CM, c,

$$p_x = c_x - \frac{\dot{L}_x}{f_z} c_z - \frac{\dot{H}_y}{f_z} \tag{1}$$

$$p_y = c_y - \frac{\dot{L}_y}{f_z} c_z + \frac{\dot{H}_x}{f_z} \tag{2}$$

where $f_z = \dot{L}_z + mg$ is the vertical ground reaction force, m is the total mass of the character and g is the positive gravitational acceleration. The above equations are the expansion of $\dot{H} = (p - c) \times (\dot{L} + mg)$ from a static analysis of momenta.The same equations are referred to as the predicted zero moment point (ZMP) in robotics [PGH05]. CP and ZMP are equivalent [Gos99] if the predicted ZMP is within the support and the character is only in contact with flat ground. The value at the edge of the support means the support is or will rotate and is a powerful indicator that the character should take a step.

According to the fall recovery mechanisms reviewed in [MMG08], linear and angular momenta induced by a push are neutralized during the impact phase of swing foot contact. We infer that a reactive step arrests momenta through proper foot placement. Assuming current linear and angular momenta of the character are L and H, simple but effective desired momenta changes can be specified as

$$\dot{L}_{des} = -d_l \cdot L \tag{3}$$

$$\dot{H}_{des} = -d_h \cdot H \tag{4}$$

where d_l and d_h are damping variables. Substituting these desired momenta changes into Equations 1 and 2 gives us a new *desired* CP which accounts for the desired momentum changes:

$$p_{x_{des}} = c_x + \frac{d_l \cdot L_x}{f_z} c_z + \frac{d_h \cdot H_y}{f_z} \tag{5}$$

$$p_{y_{des}} = c_y + \frac{d_l \cdot L_y}{f_z} c_z - \frac{d_h \cdot H_x}{f_z} \,. \tag{6}$$

We use Equations 5 and 6 to determine whether the character should step or not. We use the condition that the desired CP is outside of the current support as an indicator for when the character needs to step. When this occurrence is indicated, we anticipate that the support foot will soon rotate and therefore the character should take a step to prevent it. In contrast, Equations 1 and 2 do not indicate a step until the CP (ZMP) reaches the actual edge.

CM position and velocity have been used for the prediction of step initiation based on the IP model [PP97, PT05]. Ours is different in that we also consider the angular momentum around the CM (Equations 5 and 6). The combination of linear and angular momenta provides better prediction of character's stability than the IP. We also note that the damping values d_l and d_h affect the character's tendency to step. Higher damping values imply the character is more conservative and is more apt to take protective steps. Equations 1 and 2 give us no equivalent control over this tendency. In all of our results, we set $d_l = 4$ and $d_h = 6$.

4.2. Where to Step

To step, we select the foot which is closest to the new desired CP. Next, we employ Equations 5 and 6 with increased gain values ($d_l = 9$ and $d_h = 18$) to compute a conservative position for where to step. Anecdotally, we opt to place the character's foot at a (conservative) estimate for the desired CP value in order to enable the ability for the character to push from that point on the ground plane. In practice, this simplification works well, perhaps because the stepping foot location provides the most promising vantage point from which to push through the desired CP. This is especially clear in situations when multiple steps are required because the old support is lifted quickly following the stepping foot's touchdown. In this case, the CP must be within the new support footprint or additional angular momentum will be induced.

4.3. Comparison to Capture Point

Before going on to step synthesis, we perform a brief analysis of our method in comparison to capture point control, which is an alternative technique used to choose where to step in robotics. Capture point [PT05, PCDG06] is based on an IP model with a constant height and can be shown to be the same as the position of CM plus a velocity-scaled term or

$$x_{foot} = c_x + \sqrt{\frac{c_z}{g}}\dot{c}_x \, . \qquad (7)$$

(We focus on the x axis for brevity). While this result is derived from an energy analysis, upon observation we see that our approach adds an extension to the capture point with a term that accounts for the change in full-body angular momentum induced by external disturbances.

With careful inspection, we can reduce the differences between the capture point and our method. First, since capture point does not consider angular momentum, we could ignore angular momentum by zeroing d_h in Equation 5. Second, capture point's constant height assumption implies $f_z = mg$ and by definition $L_x = m\dot{c}_x$. Further, capture point is a model of single support; therefore the desired CP coincides with the foot. Applying these differences to Equation 5 gives us the following expression:

$$x_{foot} = c_x + \frac{d_l \cdot c_z}{g}\dot{c}_x \, . \qquad (8)$$

Comparing this simplified version of our system to capture point, we see that if we choose $d_l = \sqrt{\frac{g}{c_z}}$, Equation 8 is exactly the same as capture point. Assuming an average human height of 1 meter, $\sqrt{\frac{g}{c_z}} \approx 3.1$ is not far from our choice of $d_l = 4$. Further comparison appears in our animation results.

5. Parameterized Stepping Model

Based on the stepping goals specified by the supervisor, our system automatically plans the desired positions of the CM and the swing foot over the duration of the step. The desired trajectories are idealized by two parametric curves based on empirical evidence extracted from motion capture data.

To fit within the multiobjective controller described in [MZS09], our parameterized step model should provide both tracking and momenta objective values. The tracking objective requires joint accelerations which follow a given reference trajectory. In our case, we use a default pose and modify it using IK to follow a synthesized foot path. For momentum, we control the CM trajectory and convert this trivially to desired changes in linear momentum. We also control angular momentum change, but only about the vertical axis since angular momentum about the horizontal axes is controlled by the step position.

5.1. Swing Foot Control

We found the appearance of the overall behavior particularly sensitive to the chosen stepping path. After some experimentation, we model the desired motion of the swing foot as if it is performing a point-to-point reach - that is, considering the foot as if it is the end effector and treating the step as if it is a reaching task [WMZ08]. There has been in-depth investigation performed on hand point-to-point movement [ABM82, FH85] and in this body of work the hand traverses an approximately straight line path with a idealized bell-shaped speed profile. We adopt a similar estimate for the foot trajectory (See Figure 2). We use a synthetic Gaussian function to serve as our speed profile with a width set to 0.08, as in [WMZ08]. We automatically tune the Gaussian by scaling its amplitude such that the traversing distance matches the desired step displacement.

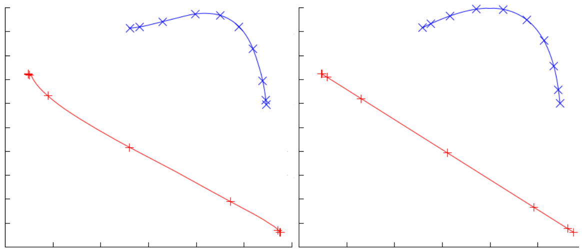

Figure 2: *Cartesian plots for the paths of swing foot (line) and CM (curve) for a motion capture example (left) and our model (right).*

For tracking, we compute the desired acceleration for the joint angles:

$$\ddot{\theta}_{des} = k_T(\theta_r - \theta) + d_T(\dot{\theta}_r - \dot{\theta}) + \ddot{\theta}_r \qquad (9)$$

where θ_r, $\dot{\theta}_r$, $\ddot{\theta}_r$ are the reference joint angle, velocity and acceleration computed at runtime based on the swing foot trajectory. The reference joint angle is resolved at runtime using IK from the simulated CM. Ideal positions of the CM and the swing foot are treated as the desired root and end-effector for solving ideal motion and reference joint velocity and acceleration are approximated using finite differencing from this ideal motion. In practice, we also found it necessary to lift the foot slightly to avoid unwanted contact between the foot and the floor - a second 0.08-width Gaussian function with controllable maximum height served for this requirement.

5.2. Center of Mass Control

Empirically, we have found that the path of CM observed in the motion capture data could be reasonably mapped using a quadratic curve (See Figure 2). Specifically, we employ a quadratic Bézier with the position of the current CM, the support foot (pivot), and the midpoint between the pivot and the desired step position as the successive control

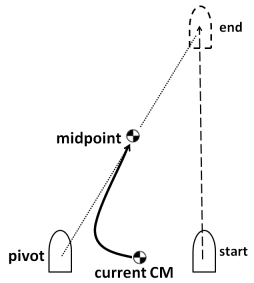

Figure 3: *The desired CM trajectory can be automatically computed by using current CM, the pivot and the midpoint as control points.*

points [WMZ08]. Although this CM trajectory seems overly simplified, while in single support the CM does follow closely to a quadratic curve, since the body is in a controlled fall. This simplification is consistent with the IP model commonly used for prediction in robotics [KKK*01, PCDG06].

We compute the desired CM acceleration using the following equation:

$$\ddot{c}_{des} = k_L(c_r - c) + d_L(\dot{c}_r - \dot{c}) + \ddot{c}_r \tag{10}$$

where c_r, $\dot{c}_r$, $\ddot{c}_r$ are the reference CM position, velocity and acceleration respectively. The values k_L and d_L are manually selected and kept as constants in all our results. We sample the entire curve using an ease-in/ease-out function to determine the reference CM positions. Reference CM velocity and acceleration are approximated numerically from the sampled CM positions over time. Equation 10 is then transformed to linear momentum change by multiplying the character's mass.

Finally, we found it necessary to control the angular momentum in the vertical axis. More specifically, damping the angular momentum around the vertical axis creates the swing of the arms. This result is supported by work in both biomechanics [HP08] and robotics [KKK*03] fields. We accomplished this by adding a simple damper in the angular momentum objective:

$$\dot{H}_z = -d_z \cdot H_z \,. \tag{11}$$

6. Implementation and Results

To demonstrate the power of our approach we present a series of animation results that highlight unique aspects of our system. All simulations were performed in real-time on a 2.4 GHz processor. The multiobjective optimization was solved at a frequency of 60 Hz and the inverse dynamics computed joint torques at the simulation rate 2000 Hz.

Directed stepping. To show the basic operation of the tool, we input a series of footsteps for the character to follow. Each footstep is shown as a red indicator in the animation (as in Figure 1). We demonstrate that we can reorient and position the character by taking a small number of directed steps. Further, because only high-level goals are controlled and no character specific parameters are set, we can change the character's configuration. We showcase the value of this aspect of our system by making the character take steps in a handstand.

Reactive stepping. Responsivity is an important feature of the controller. In the related animations, we show that the character can sustain multiple impulses by taking steps in various directions. Each impact is 170 N applied for 0.1 sec. The resulting action is both complex and believable, especially considering no motion capture data was used (Figure 4). In addition, we show that the supervisor can opt not to take a step and instead use standing balance control [MZS09] to respond to the impact. As mentioned in Section 4.1, d_l and d_h can be used to control the character's tendency to step.

Our stepping mode is considered between statically stable steps, i.e. zero initial velocity and zero target velocity. However, the impact force created by the swing foot contact might cause non-zero target velocity after each step. This effect is not a problem as our system can recursively apply the supervisor after each step to automatically determine if another step is needed.

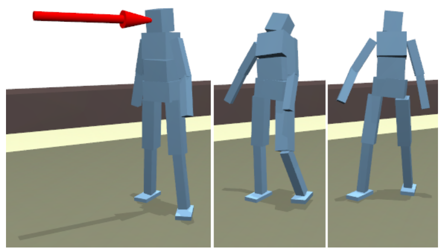

Figure 4: *A reactive step generated automatically in response to a disturbance.*

Comparison to IP. We contrast our momentum supervisor to IP by following the descriptions in Section 4.3. In this comparison, the IP does not include the angular momentum terms in Equations 5 and 6. The result shows the momentum supervisor initiates the recovery earlier than the IP under the same impact. Under small disturbance, both IP and momentum supervisor are able to maintain character's balance by stepping. However, being able to quickly initiate a step is especially important under large disturbance. We show that our supervisor can still keep the character in balance while the IP fails when the force is increased to 250 N for 0.1 sec. For clarity we note that this is not exactly the capture point since our domain is double support while capture point is single support.

7. Conclusions

In this paper, we present a goal-directed controller for simulated characters to perform directed and reactive steps by guiding the CM, and the swing foot. The character is able to follow the desired step positions (footprints) specified by the user. The same controller works for different character morphology. To react to a disturbance, the character can take protective steps computed automatically by our momentum-based supervisor. Considering both linear and angular momenta in the supervisor improves character's robustness to disturbances. Lastly, the required number of steps is automatically determined by applying the supervisor after each step.

Acknowledgments

The first author is supported by the USDA Grant CALE-2008-02162 and UCR Dissertation Year Fellowship. We acknowledge Vicon Motion Systems for their equipment donation and Adriano Macchietto for his input and use of his software.

References

[ABM82] ABEND W., BIZZI E., MORASSO P.: Human arm trajectory formation. *Brain*, 105 (June 1982), 331–348.

[ADP07] ABE Y., DASILVA M., POPOVIĆ J.: Multiobjective control with frictional contacts. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2007).

[AFO05] ARIKAN O., FORSYTH D. A., O'BRIEN J. F.: Pushing people around. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2005).

[DAP08] DASILVA M., ABE Y., POPOVIĆ J.: Interactive simulation of stylized human locomotion. *ACM Transactions on Graphics 27*, 3 (2008).

[FH85] FLASH T., HOGAN N.: The coordination of arm movements: an experimentally confirmed mathematical model. *Journal of Neurosciences 5*, 7 (July 1985), 1688–1703.

[FvdPT01] FALOUTSOS P., VAN DE PANNE M., TERZOPOULOS D.: Composable controllers for physics-based character animation. In *SIGGRAPH* (Aug. 2001), pp. 251–260.

[Gos99] GOSWAMI A.: Postural Stability of Biped Robots and the Foot-Rotation Indicator (FRI) Point. *The International Journal of Robotics Research 18*, 6 (1999).

[HP08] HERR H., POPOVIĆ M.: Angular momentum in human walking. *Journal of Experimental Biology 211*, 4 (Feb. 2008), 467–481.

[JYL09] JAIN S., YE Y., LIU C. K.: Optimization-based interactive motion synthesis. *ACM Transaction on Graphics 28*, 1 (2009), 1–10.

[KKI06] KUDOH S., KOMURA T., IKEUCHI K.: Stepping motion for a humanlike character to maintain balance against large perturbations. In *IEEE Int. Conf. Robotics and Automation* (2006).

[KKK*01] KAJITA S., KANEHIRO F., KANEKO K., YOKOI K., HIRUKAWA H.: The 3D Linear Inverted Pendulum Mode: A simple modeling for a biped walking pattern generation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2001), pp. 239–246.

[KKK*03] KAJITA S., KANEHIRO F., KANEKO K., FUJIWARA K., HARADA K., YOKOI K., HIRUKAWA H.: Resolved momentum control: humanoid motion planning based on the linear and angular momentum. *Intelligent Robots and Systems* (2003).

[KLK04] KOMURA T., LEUNG H., KUFFNER J.: Animating reactive motions for biped locomotion. In *Proc. ACM Symp. on Virtual Reality Software and Technology (VRST)* (2004).

[MLPP09] MUICO U., LEE Y., POPOVIĆ J., POPOVIĆ Z.: Contact-aware nonlinear control of dynamic characters. *ACM Transactions on Graphics 28*, 3 (July 2009), 81:1–81:9.

[MMG08] MATHIYAKOM W., MCNITT-GRAY J.: Regulation of angular impulse during fall recovery. *Journal of Rehabilitation Research & Development 45*, 8 (2008), 1237–1248.

[MZS09] MACCHIETTO A., ZORDAN V., SHELTON C.: Momentum control for balance. *ACM Transactions on Graphics 28*, 3 (2009).

[PCDG06] PRATT J., CARFF J., DRAKUNOV S., GOSWAMI A.: Capture Point: A Step toward humanoid push recovery. *Proceedings of the IEEE-RAS/RSJ International Conference on Humanoid Robots* (2006).

[PGH05] POPOVIĆ M., GOSWAMI A., HERR H.: Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. *International Journal of Robotics Research 24*, 12 (2005).

[PP97] PAI Y.-C., PATTON J.: Center of mass velocity-position predictions for balance control. *Journal of Biomechanics 30*, 4 (1997).

[PT05] PRATT J., TEDRAKE R.: Velocity-based stability margins for fast bipedal walking. *Fast Motions in Robotics and Biomechanics Optimization and Feedback Control* (2005).

[SC92a] STEWART A. J., CREMER J. F.: Animation of 3d human locomotion: Climbing stairs and descending stairs. In *In Eurographics Workshop on Animation and Simulation* (1992).

[SC92b] STEWART A. J., CREMER J. F.: Beyond keyframing: An algorithmic approach to animation. In *Proceedings of Graphics Interface '92* (1992), pp. 273–281.

[SCCH09] SHIRATORI T., COLEY B., CHAM R., HODGINS J. K.: Simulating balance recovery responses to trips based on biomechanical principles. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2009).

[SKL07] SOK K. W., KIM M., LEE J.: Simulating biped behaviors from human motion data. *ACM Transactions on Graphics 26*, 3 (2007).

[SM01] SUN H. C., METAXAS D. N.: Automating gait generation. In *Proceedings of ACM SIGGRAPH 2001* (Aug. 2001), Computer Graphics Proceedings, Annual Conference Series, pp. 261–270.

[TGB00] TOLANI D., GOSWAMI A., BADLER N. I.: Real-time inverse kinematics techniques for anthropomorphic limbs. *Graph. Models Image Process. 62*, 5 (2000), 353–388.

[vdP97] VAN DE PANNE M.: From footprints to animation. In *Computer Graphics Forum* (1997), vol. 16, pp. 211–223.

[WMZ08] WU C.-C., MEDINA J., ZORDAN V.: Simple steps for simply stepping. *International Symposium on Visual Computing* (2008).

[YLvdP07] YIN K., LOKEN K., VAN DE PANNE M.: Simbicon: Simple biped locomotion control. *ACM Transactions on Graphics 26*, 3 (2007).

[YPvdP05] YIN K., PAI D. K., VAN DE PANNE M.: Data-driven interactive balancing behaviors. *Pacific Graphics* (2005), 118–121.