

Control of Rotational Dynamics for Ground Behaviors

David F. Brown*
University of British Columbia

Adriano Macchietto

KangKang Yin†
National University of Singapore

Victor Zordan‡
University of California, Riverside



Figure 1: A parkour style forward roll.

Abstract

This paper proposes a physics-based framework to generate rolling behaviors with significant rotational components. The proposed technique is a general approach for guiding coordinated action that can be layered over existing control architectures through the purposeful regulation of specific whole-body features. Namely, we apply control for rotation through the specification and execution of specific desired ‘rotation indices’ for whole-body orientation, angular velocity and angular momentum control. We account for the stylistic components of behaviors through reference posture control. The novelty of the described work includes control over behaviors with considerable rotational components as well as a number of characteristics useful for general control, such as flexible posture tracking and contact control planning.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: physics-based animation, character control, human motion

1 Introduction

Rotational motion contributes to a wide class of interesting behaviors, including tumbling on the ground, gymnastic actions such as flips and back handsprings, and many martial arts and dance behaviors, for example spinning kicks and break dancing. With the exception of a few cases, such as the work of Wooten and Hodgins [2000], most physics-based control papers for character animation have overlooked these types of motions in lieu of other actions, largely locomotion. While we limit our scope in this paper to rotational ground behaviors such as rolling, our interest is set on control of rotational dynamics in general.

What makes rotation-rich behaviors distinct in terms of control is

*e-mail:dfbrown@cs.ubc.ca

†e-mail:kkyin@comp.nus.edu.sg

‡e-mail:vbz@cs.ucr.edu

the need to manage whole-body spin in conjunction with the other aspects of control. That is — in addition to balance, body displacement, and support placement — facing orientation and angular velocity must be guided in meaningful ways to accomplish the proper spin of an action. While this concept can be stated simply, it adds to the complexity in control through the coupling of rotational and translational components both in terms of timing and spatial alignment. Thus while simple feedback may be used, for example, to adjust foot placement in taking a step for locomotion, there is an inherent assumption that the facing orientation is known and/or close to a known or fixed orientation. However, this assumption becomes invalid when a motion includes a significant rotational component.

Previous techniques sidestepped this issue by discovering specific workable solutions that overcome the issues of this coupling, either through manual tuning [Wooten and Hodgins 2000] or sampling [Liu et al. 2010]. However, our goal is to address the issue of coupled rotation and translation effects in a direct manner, by explicitly controlling both simultaneously in a general manner. To achieve this goal requires better control of rotational dynamics than has been seen to date in computer animation. Further, it raises a number of challenges unique to rotational action. Diverging from non-rotational tasks such as walking, running, and balance, we have identified that rotational ground behaviors have two aspects that make their control challenging currently:

- Various control *indices* have been proposed for balance and locomotion tasks, e.g., zero moment point (ZMP), but which quantity or quantities to monitor and control during rotational behaviors is not well understood or documented. For example, how can we control smooth progression in a continuous rolling action? General rotational indices have not been proposed, and are not commonly known or discussed. Our investigations in this paper suggest that a collection of rotation indices can be controlled, even synergistically, paralleling observations that no single balance index has been shown to solve all locomotion tasks.
- The contact state and its progression in most locomotion tasks is reasonably clean and phase dependent. Left foot follows right, contact goes from heel to toe, and so on. However, contact states in rotational behaviors span a large spectrum. For example in a sideways (log) roll several body parts can contact and push off the ground at various times in the cycle. What contacts should be used to propel a given roll at a given time is unclear and changes based on the situation and style of the behavior. To regulate contact-rich rotational motion successfully, a controller must have access to means for managing control of the character in the presence of any number of po-

tentially unknown contact configurations.

We propose a hierarchical control technique that handles both of these factors by combining a low-level local controller which can follow any of a variety of rotational inputs along with a supervisory control module employed to manage contact-dependent control issues.

Our contributions include:

- The introduction of a systematic study of control of rotational dynamics and rotation indices for simulation of humanlike motion;
- The realization of control over abstract rotation indices through torque regulation using multiobjective optimization;
- The introduction of a practical contact-planning mechanism to aid in contact-rich control for rolling and similar behaviors;

2 Related Work

Recent publication trends reveal an increased interest in physically based characters. In particular, a host of control techniques have been proposed for behavior activities including running [Wampler and Popović 2009; Coros et al. 2010; de Lasa et al. 2010; Kwon and Hodgins 2010; Mordatch et al. 2010], walking [Sok et al. 2007; Yin et al. 2007; da Silva et al. 2008; Wang et al. 2009; Lee et al. 2010; Wu and Popović 2010], leaping [de Lasa et al. 2010], and standing (balance) [Abe et al. 2007; Macchietto et al. 2009]. To date, most full-body control systems have employed some variant of center of mass (CM) control, often coupled with controls or constraints that manage ground contact forces. Several control methods date back to early legged motion controllers, specifically Raibert’s control approach for legged robot locomotion [Raibert and Hodgins 1991] which was adapted for general character locomotion in Simbicon-type controllers [Yin et al. 2007; Wang et al. 2009; Lee et al. 2010; Coros et al. 2011]. These controllers adjust foot placement in order to control CM position and velocity. Other approaches use an abstract CM objective which is realized through optimization to maintain full-body control [Abe et al. 2007; da Silva et al. 2008; Macchietto et al. 2009; Muico et al. 2009; de Lasa et al. 2010; Kwon and Hodgins 2010; Mordatch et al. 2010; Wu and Popović 2010; Wu and Zordan 2010; Ye and Liu 2010]. However, CM-based control alone cannot account for whole-body rotations in an explicit manner. General methods for controlling large purposeful rotation are lacking.

A select number of examples appear in the motion editing literature that consider such physical aspects of rotating (twisting) hops and flipping [Abe et al. 2004; Majkowska and Faloutsos 2007]. However, these methods do not involve forward dynamics simulation and control. For human characters, a limited number of control techniques appear for flight-based rotational behaviors [Hodgins et al. 1995; Wooten and Hodgins 2000; da Silva et al. 2009; de Lasa et al. 2010], and they offer little in terms of general methods for rotation control. Rolling [Kry et al. 2009] and flipping [Kim and Pollard 2011] controllers have been demonstrated for non-human characters, using modal analysis and user interaction to generate the control, respectively. However it is not clear how to generalize these techniques for rotation control of more complex, humanoid characters. For humanlike rolling control specifically, little previous work appears. The sampling approach proposed by Liu et al. [2010] demonstrates that open-loop control can be generated for humanoid rolling, but such *open-loop* control cannot act responsively to changes in the environment. Liu et al. [2012] and Ha et al. [2012] show closed-loop control for isolated rolling motion. However, such techniques do not account for perturbations

that can accumulate over time (after multiple rolls), indicating a need for more careful control over rotational dynamics. In contrast, our framework systematically controls rotation through identified rotation indices and can generate continuous rolling as well as correction in the presence of disturbances.

3 Overview

Our system is derived from a layered hierarchical architecture (See Figure 2). At its lowest level, we employ a semi-implicit multibody dynamics module and force-based constraint solver (Appendix A). We compute activation with a torque controller (Section 4) which has the form of previous feature-based frameworks [de Lasa et al. 2010]. Two abstract features are employed, one to control whole-body linear and angular momenta changes (4.1) and the second to track poses (4.2). Using a blend-tracking objective, the system is able to select control from a range, designated by multiple input poses (4.3). Also, the control optimization is guided by a supervisory level contact controller which decides, of the body parts in contact with the ground, which to employ for advancing control (Section 5). Taken as a whole, the system provides means for controlling rotation in general through the specification of rotation indices described next.

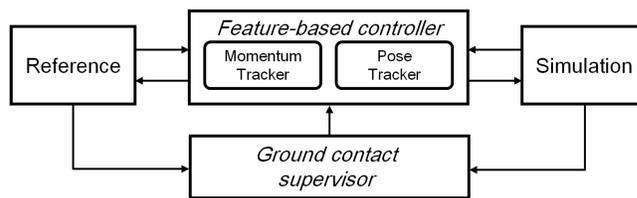


Figure 2: System Layout

Many successful locomotion, stepping, and standing control schemes have been proposed. Most such research efforts rely on a *balance index* for monitoring and controlling the balance and stability of the character or robots. These indices include center of mass (CM), center of pressure (CP), zero moment point (ZMP), foot rotation index (FRI), centroidal moment pivot (CMP), and several others [Popovic et al. 2005]. In the course of our investigations in rotation, we found it important to identify of analogous *rotation indices* for controlling and maintaining balance in rotational behaviors. We define a set of such below and bind these to control through our abstract feature objectives.

Angular Momentum, AM: Whole-body angular momentum, H , about the CM is an obvious choice for rotation control. It is low dimensional and has been shown recently to be helpful in inducing full-body rotational effects for various activities [Macchietto et al. 2009; de Lasa et al. 2010; Ye and Liu 2010]. However, as a quantity, it is not directly intuitive because it is an aggregate of two factors, whole-body inertia and angular velocity. Therefore, while we do exploit angular momentum in our control, we use it as a computed quantity that is driven by other rotation indices.

Angular Velocity, AV: The whole-body angular velocity about the center of mass, ω , can be computed as $I_c^{-1}H$ where I_c is the whole body composite inertia matrix about the CM. This value has two features that make it attractive as a control index. First, it is fairly intuitive and, second, it can be manipulated through purposeful shaping of body which leads to change in inertia. Thus, through guiding the body pose as one of our abstract features, we retain control over the angular velocity during flight. Under these considerations, whole-body angular velocity becomes a powerful choice for general control of rotation.

Angular Excursion, AE: Angular velocity itself cannot provide precise control over the orientation of the body. However, to carefully control full-body orientation requires a robust definition of body rotation. While various surrogates have been used to indicate body rotation in locomotion, such as the root or trunk orientation (designated **ROOT** control in later discussion), large full-body rotations are not well-described by such substitutions. We propose the use of the whole-body *angular excursion*, ϕ , for quantifying *total postural orientation* of the character. This value is defined as

$$\phi(t) = \int_0^t \omega dt + \phi_0 \quad (1)$$

where ϕ_0 is a fixed known reference, i.e. the zero or starting orientation. Employed by [Popovic et al. 2004a] for motion analysis (not control), such whole-body angular excursion may be seen as the angular analog of the CM. We found this value provides a useful handle to regulate full-body orientation of the character.

4 Feature-based control

Our controller is formulated as a convex quadratic optimization problem with linear constraints. We solve this problem using quadratic programming (QP) with a feature-control strategy [de Lasa et al. 2010]. In this strategy, each abstract objective has the form

$$E(x) = \frac{1}{2} \|Ax - b\|_W^2$$

where x is the optimization state and A and b are a matrix and vector describing linear and constant terms of each objective task function, and W is a weighting matrix that scales the objective error, allowing the user to control the relative importance of each objective within the minimization. The optimization state is the concatenation of the generalized accelerations $\ddot{\theta} \in R^n$, the generalized actuator forces $\tau \in R^{n-\delta}$, the contact constraint forces $\lambda \in R^{3m}$, and a scalar pose-blending selection parameter, γ :

$$x = \begin{bmatrix} \ddot{\theta} \\ \tau \\ \lambda \\ \gamma \end{bmatrix}$$

where n and m are the degrees of freedom (DOFs) of the character and the number of contact points, respectively. γ is the blend value used as an interpolant weight for blend tracking, as described in Section 4.3. Note, only the optimized generalized actuator forces are directly applied to the simulator, the rest of the state is ignored in the simulation update.

In contact, we uphold the zero-work complementarity condition as a heavily weighted objective by enforcing $v_{t+h} = 0$, where v_{t+h} is the post-halfstep velocity (i.e. after velocity integration, but before position integration). We use the post-halfstep velocity, rather than acceleration, to estimate impacts within the control. v_{t+h} is related to the generalized accelerations by the formula:

$$v_{t+h} = J(\theta_t) \dot{\theta}_{t+h} = J(\theta_t) (\dot{\theta}_t + h \ddot{\theta}_t)$$

where J is the constraint Jacobian, and $\dot{\theta}_{t+h} = \dot{\theta}_t + h \ddot{\theta}_t$ by Euler integration rules. In previous work [de Lasa et al. 2010], this complementarity condition is not upheld in the control phase, and the effect of the control is compromised by forcing the zero-work condition to be introduced in the simulation. In contrast, we found

our approach reduces deviation between the desired outcome of the control and its resulting effect in the simulation. As in similar quadratic control formulations [Abe et al. 2007; da Silva et al. 2008; de Lasa et al. 2010], we use a linearization of Coulomb friction, $\|\lambda_T^{(k)}\|_\infty \leq \mu |\lambda_N^{(k)}|$ instead of $\|\lambda_T^{(k)}\|_2 \leq \mu |\lambda_N^{(k)}|$, to keep friction constraints linear. $\lambda_N^{(k)}$ and $\lambda_T^{(k)}$ are the normal and tangent force magnitudes of the k 'th contact point.

The optimization problem is summarized as follows:

$$\begin{aligned} \min_x \quad & \frac{1}{2} \|Ax - b\|_W^2 + \frac{1}{2} \|v_{t+h}\|_{W_v}^2 \\ \text{subject to} \quad & \lambda_N^{(i)} \geq 0, \\ & \|\lambda_T^{(i)}\|_\infty \leq \mu \lambda_N^{(i)}, \quad i = 1, \dots, m \\ & S\tau = M\ddot{\theta} + C. \end{aligned} \quad (2)$$

where the last constraint ensures that the generalized forces and accelerations are consistent with the dynamics equations. M is the generalized mass matrix, C are the generalized Coriolis, centrifugal and external forces, and S is a selection matrix that zeros the 6 DOF of the unactuated root. W_v is set to 1000 contrasting unit values in the other objective weights. Both the simulator and control optimization run in lockstep at 60 *hz*.

4.1 Momentum tracking

To control linear and angular momenta, L and H respectively, we employ two straightforward objectives that measure the deviation of their derivative values from set target values \dot{L}_d and \dot{H}_d ,

$$E_L(x) = \frac{1}{2} \|\dot{L}_d - \dot{L}\|^2$$

$$E_H(x) = \frac{1}{2} \|\dot{H}_d - \dot{H}\|^2$$

To compute the momentum objectives, we must specify the respective target values. We determine desired linear momentum change based on CM, c , and its derivative. Letting quantities with $\hat{\cdot}$ represent the reference/target values, and K_s , K_d , be the spring and damping gain matrices,

$$\dot{L}_d = m \left(K_s^L (\hat{c} - c) + K_d^L (\hat{\dot{c}} - \dot{c}) \right). \quad (3)$$

For rotation control, we introduce an analogous function for angular momentum which controls rotation via two rotation indices, **AE** and **AV**, denoted ϕ and ω respectively.

$$\dot{H}_d = I \left(K_s^H D(\hat{\phi}, \phi) + K_d^H (\hat{\omega} - \omega) \right) \quad (4)$$

where we define function D to be the arithmetic distance operation for the excursion error values following [da Silva et al. 2008]. We refer interested readers to the related previous work [de Lasa et al. 2010; Macchietto et al. 2009] regarding details omitted here for brevity.

While such **AM** control has seen some use in recent physics-based animation publications, its control has been used primarily to prevent rotation (e.g. tipping) rather than to induce it. Macchietto et al. [2009] employ **AM** regulation to gain control over the center of pressure in balanced standing. Several other researchers [de Lasa

et al. 2010; Ye and Liu 2010; Wu and Zordan 2010] follow a zero-spin strategy to damp **AM**, as described in biomechanics for locomotion [Popovic et al. 2004a; Popovic et al. 2004b]. One exception is de Lasa et al. [2010] where they induce **AM** about the vertical axis to produce a turning jump. In contrast, we employ an abstract angular momentum target value to control rotation through the proposed **AE** and **AV** rotation indices.

4.2 Pose tracking

Unlike prior optimization-based controllers [Abe et al. 2007; da Silva et al. 2008; Macchietto et al. 2009] that use acceleration-level pose trackers, our *force-based* pose tracker objective takes the form

$$E_P(x) = \frac{1}{2} \|\tau_d - \tau\|^2 \quad (5)$$

where

$$\tau_d = I \left(K_s D(\hat{\theta}, \theta) + K_d (\dot{\hat{\theta}} - \dot{\theta}) \right). \quad (6)$$

We found this tracker lead to more stable, compliant pose tracking over one driven by accelerations. Wu and Popović [2010] demonstrate similar compliance using “reaction frame” pairs which transmit forces through the character, our controller is similar but acts at the joint level to produces inter-body response to impacts and unmodeled contacts. Our approach shares similarities with Tan et al. [2011] although the control and testbeds are quite different.

4.3 Automatic blend tracking

To make the pose tracking more flexible, we add automatic blend tracking. That is, we provide the system with two reference poses r_a, r_b and allow the control optimization to select γ , a blend value to track between these poses. The blend tracker objective function has the form

$$E_B(x) = \frac{1}{2} \|\gamma \tau_a + (1 - \gamma) \tau_b - \tau\|^2 \quad (7)$$

where τ_a and τ_b are computed using Equ. 6 with $\hat{\theta}$ set to r_a and r_b respectively. ($\dot{\hat{\theta}}$ s are set to 0.) Through additional constraints added to Equ. 2 to ensure $0 \leq \gamma \leq 1$, this blend tracker allows the optimization to track a linear blend of the desired torques associated with each input pose, without incurring additional cost. Intuitively, this relaxes the demand to track a specific pose and provides the system some freedom over pose selection while maintaining naturalness by staying within the range of the input poses.

For control, we employ this *pose-blend* tracking in two distinct ways. First, we can use extreme poses to sweep out a desired torque space for the given task. For example in rolling, we identify two extreme poses of the desired roll as a tight tuck and fully extended. A noted benefit is that the controller can then automatically select from a meaningful activation range which is defined easily by an animator. Second, we can also use the pose-blend tracker to flexibly follow a reference animation from either motion capture or hand animation. Instead of directly tracking a single pose based on time, we track a blend surrounding a current pose. To do this for rolling, we find the frame of the reference motion that corresponds to the phase of the roll based on the character’s angular excursion about the rolling axis, ϕ_r . We then select the blend poses from the reference motion using a window centered at the found frame’s time,

t_r . Thus, $r_a = r(t_r - \Delta t)$ and $r_b = r(t_r + \Delta t)$ where $r(t)$ is the reference animation and Δt is a fixed look-ahead/behind duration. The blend poses are updated with each optimization pass based on the simulation state. By decoupling the pose selection from time, the controller has an opportunity to advance or stall the progression of the roll cycle based on the simulation’s progress. We found that this approach leads to a more robust roll without deviating greatly from the style embedded in the reference motion.

5 Ground contact supervisor

Contact selection control is an issue when various contacts are in flux, as is the case with rolling, because often certain contacts should be employed for control, while others should be ignored. The choice of what contacts to employ for control depends both on the dynamics as well as the strategy. Clearly, humans use specific strategies to induce rolling, for example, it is often desirable *not* to use the head for administering rolling control forces even if the head incidentally comes in contact with the floor. In addition, there are contacts that should not be used for control as they may impede desired strategies intended for the given body part or limb - for example swinging an arm or leg in place for future support. Along with issues, related to strategy, we found a basic need for directing the described controller to avoid enforcing the zero work complementarity condition on contact points unnecessarily. Since our approximation of the zero work condition minimizes contact velocity, the controller will attempt to maintain (stall) the position of all points of contact without discretion. The effect of this is that contact points can become “sticky”, forces to remain on the ground by the controller when they would otherwise be lifted off. This problem is exacerbated in rolling where contact is highly irregular and many incidental collisions take place that should be ignored entirely by the control routine.

To address this problem, a supervisory routine performs an *inclusion test* to discern which of the contacts the control is to use at a given time step. While search could be employed to explore the optimal contact (e.g. to determine the set of contacts that leads to the best performance), testing every possible contact configuration would be too time consuming. Instead we experimented with several inclusion tests and found one empirically that was suitable for our purposes. Specifically, the contact supervisor performs an initial pass of the control optimization using the full set of contact points, minus contacts on the head held out for stylistic purposes by default. From this operation, it determines what forces the controller would use naively based on the current conditions of the character. (Note, while we call this naive, it does also embed strategies that might be derived from the reference motion, such as lifting the hands to prepare for a future contact.) It then selects only contacts for which force or torque about the center of mass is above a certain threshold. These contacts are then given to the control optimizer to perform the *actual* optimization. The results of this optimization are used to drive the simulated character. Note, the subset of the contacts is only used for determining the joint torques for control, while the entire set of contacts is always used for ground contact calculations in the forward simulation step. Thus, the supervisor is not producing a change in the physical correctness, only the control strategy surrounding contact. This process is repeated for each time step.

We experimented with several other *inclusion tests* for contact selection before settling upon the one described. Others we tested included a minimal set that gave the controller the ability to generate the necessary force/torque without employing more than was necessary. As well, we tested sets which added biases to contacts lying on the convex hull of all of the contact points. Further, we explored performing inclusion testing following the complete simula-

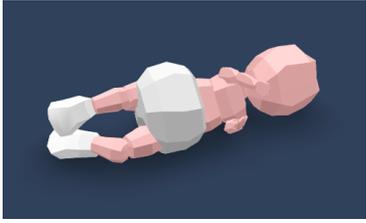


Figure 3: *Baby roll.*

tion step (and rewinding), with the momenta objectives both turned on and held out. We opted to select our inclusion test because others revealed various visual artifacts and undesirable features. Although we found our solution was suitable for the requirements of ground rotation control and rolling behaviors, we believe more optimal contact selection strategies are likely.

6 Experiments

We experimented with continuous rolling in various styles and under various conditions. We also use the example of a baby rolling over (Figure 3) to explore control over target orientation. Specific behaviors are derived from simple strategies for specifying the desired task. All experiments use a fixed 60hz timestep for the simulation and optimization. The system is single threaded and runs at 20% - 40% of real time on a 2.4GHz quad core CPU.

Continuous rolling control. For continuous forward, backward, and side (log) rolling control, we consider control in the direction of the roll (“in-plane”) and the rest (“out-of-plane”). The video corresponding to the paper shows a number of cases for each and we highlight those here. For example, we experimented with controlling the orientation and angular velocity of the **ROOT** body and one can clearly see from the video that this approach leads to highly unnatural, unsatisfactory motion. Ultimately we found a combination of control over both **AV** and **AE** together lead to success for continuous rolling.

Our resulting rolling control technique manages in-plane and out-of-plane motion separately. Specifically, in-plane, we maintain **AV**, $\hat{\omega}_y$, as a constant, rolling in the x -direction with z up. $\hat{\omega}_y$ is an easily hand-tuned rolling velocity, see Table 1. We allow **AE** to be *uncontrolled*, by setting $K_{yy}^H = 0$ in Equ. 4. If instead we chose a non-zero value and attempt to control **AE** (e.g. to follow a prescribed reference trajectory), errors would propagate and lead to poor quality rolling as the controller attempted to keep up with the desired excursion value(s). The video shows an example. Out of the plane, both target **AV** and **AE** are set to zero, thereby resisting rotation. If out-of-plane **AE** was not controlled, small errors would quickly lead to rolling off-axis (see video.) In a similar fashion, linear velocity in-plane, \hat{c}_x , is held constant (Table 1). And $K_{xx}^L = 0$ in Equ. 3 to allow center of mass position errors to be overlooked in the direction of rolling. $\dot{c}_{(x|y)}$ is simply damped by setting its target value to zero.

Orientation control. In a more whimsical experiment, we conduct an orienting task of rolling over for a baby character, as shown in Figure 3. We specify this control by setting **AE** to the equivalent of a 180-degree roll about the primary axis. **AV** is set to zero, as are all values for center of mass. With the same working values demonstrated in the rolling behaviors, the baby easily, effortlessly rolls over. However, by lowering K_x^H , this character can be made too weak to accomplish the task, and yet the resulting animation reveals a level of determination that communicates the child’s intention to roll over, even in the “failed” case. The video showcases

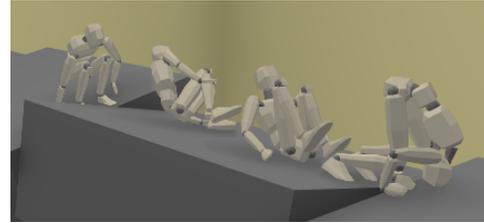


Figure 4: *Forward roll on uneven terrain.*

Example	Input type	$\hat{\omega}_y$ rev./s	\hat{c}_x m/s
Forward Roll 1	Keyframe	1.3	1.2
Forward Roll 2	Reference	0.9	1.2
Forward Roll 3 (parkour)	Reference	1.5	1.8
Backward Roll 1	Keyframe	1.3	1.0
Backward Roll 2	Reference	1.0	1.2
Sideways Roll	Reference	1.0	1.5
Uneven terrain	Reference	0.9	1.1
Incline	Reference	1.8	1.5
Decline	Reference	0.4	1.0
Stairs	Reference	0.6	0.3

Table 1: *Linear and angular velocities in-plane (shown) varied between examples but all other control and gain values, including pose tracking gains, were fixed for all but extreme examples.*

both of these experiments for orientation control.

Style Control. At the minimum the system only needs two poses to accomplish either the continuous roll or the roll over task. Here, the power of the blend controller from Section 4.3 is truly demonstrated. For continuous rolling, the two poses are hand-designed to include one which is tucked and one which has the body extended as r_a and r_b respectively. The value of γ leads to all pose variation observed in the video for the keyframe examples. We show results of this approach for forward and backward rolls in the video (called Forward Roll 1 and Backward Roll 1). Note the shape of the two crafted poses dictate (to some extent) the style of the final motion.

To control style further, we employ human reference motions of a single cycle each for four behaviors, including two different types of forward rolls, a backward roll, and a sideways “log” roll. As we see in the video, we chose to modify our basic roll cycle (called Forward Roll 2) to ensure it was symmetric and cycled well, however, we found it was unnecessary to do this in general and none of the other examples were modified after simply segmenting to make each a single complete cycle. The filmstrip in Figure 1 showcases an asymmetric “parkour”-style forward roll (designated as Forward Roll 3 in the video). To employ these references, we “phase-lock” the reference cycle to the current state of the animation via its in-plane excursion value, ϕ_y , as described in Section 4.3.

Robustness testing. To test the robustness of our control method, we expose the character to various conditions including uneven terrain and environmental “hazards”. An example of such appears in Figure 4. For rolling along incline and decline slopes, we adjust the linear and angular velocities in expected ways (Table 1). Note, we also found it necessary to align the linear velocity with the direction of the slope, that is, parallel to the slope. Without the adjustment, the motion reveals undesirable characteristics, such as the character leaping off the ground in a downhill roll. We also show results demonstrating correction following a response due to an unpredicted (out-of-plane) impulse. The rolling character reacts in a physically consistent manner and then returns to the rolling task within a short time frame.

7 Discussion and conclusion

Ultimately, functional control for rotation in rolling is derived simply in our system from the selection of desired angular velocity (**AV**) and corresponding linear velocity terms which are selected once (easily) and held fixed for the duration of the behavior. **AE** (held to zero) is also necessary to gain precise control, avoiding small orientation errors from propagating. We also experimented with **AM** and, at least for rolling, we found it similar to **AV** (likely due to the fact that whole-body inertia is largely unchanging in rolling). However, like **AV**, **AM** showed a need for feedback on orientation via **AE**. We found the momenta values to be less intuitive than angular velocity and opted to use **AV** in our control.

Without setting the described angular (and linear) values properly, reference motion alone did not lead to successful rolling, as defined by the ability to roll continuously and to correct from disturbances. However, we did observe some benefit from the various changes in the body derived from the reference motion. While we can assume such changes aid the behaviors, as they affect the contact and inertia, it is difficult to tease out the specific influences of each. Qualitatively, we did observe richer, more natural-appearing motion with properly aligned reference data. Fundamentally, rotation in the real world is derived from the combination of the control over ground forces and control over the shape of the pose, leading to change in inertia. In our controller, we make a conscious choice to manage the **AV** which allows the pose control to ultimately control the inertia. While this seemed to work well for our experiments, it is unclear that there won't be better solutions ahead that integrate body shape more explicitly. We plan to explore this topic in future work.

While it is outside the scope of this paper, control for aerial behaviors with rotational components are of great interest to us. However, they require an additional component to the control described here, in order to plan for inflight phases of the motions. We have experimented with a handful of such behaviors thus far and we observe many of the same traits with respect to control for rotation with the caveat that once the character has left the ground, the game changes and either careful planning, or inertia-driven pose control is required. In addition, balance is more important, for landing. But control over rotation is derived from the same fundamentals we explore in this paper, including the importance of each of the rotation indices identified and investigated here.

Given the lack of examples coupled with a number of reported issues on motions with large rotations in many of the control results, the exploration of the topic of control of rotational dynamics seems timely and needed. We see this paper not as a conclusory one, but as broaching the subject - which is largely why we highlight the introduction and exploration of possible rotation indices. Our findings indicate that multiple indices are useful depending on circumstance and our observations reveal the potential for more such exploration. We also note that we introduce **AE** as a useful index for managing orientation which has not been applied in control before, to our knowledge. Further, we hypothesize that body-orientation control is a missing component of existing approaches and with even simple control (as we describe for our rolling examples without an explicit balance strategy in place), many frameworks would realize improvement in robustness and visual quality.

In conclusion, as the first effort that systematically studies and synthesizes rotation behaviors, our methodologies provide a solid point of departure for future research effort. While we have not tested our control on flight phase rotation, such actions need planning mechanisms that are orthogonal to the focus of this paper. Similarly, we have not tested rotation behaviors around non-principal axes of inertia. Such rotations, however, are not stable and normally do not appear in voluntary (controlled) rotations. We have presented

a comprehensive system for the simulation and control of rotationally rich rolling behaviors. Rotation behaviors with different style and contact characteristics can be controlled and simulated within the same framework. A collection of rotation indices is proposed and synergistically controlled by our control system. When put together, the rotation indices and control components enable a powerful and complete system for synthesizing motion skills that have purposeful full-body rotations with built-in physical realism.

Acknowledgement

The authors thank Stevie Giovanni and Thai-Duong Hoang for their help in modeling and rendering as well as our reviewers for their helpful and constructive feedback. This work was partially supported by Singapore Ministry of Education Academic Research Fund Tier 2 (MOE2011-T2-2-152).

Appendix A Constrained multibody dynamics

We employ a semi-implicit simulation which can act stably at large timesteps (60hz). This is ideal for character animation, especially for games and interactive settings. Also, we can exploit this formulation for control by informing the controller about current impacts. In contrast, previous methods either ignore such forces or suffer from stability issues that require smaller timesteps. The technique we use sidesteps both of these limitations. Specifically, the simulation extends the constraint resolution method [Erleben 2007] to work with reduced-coordinate constrained multi-body dynamics system [Featherstone 1987] so it may be applied to character animation. We use Featherstone to handle the body constraints within the character and incorporate Lagrange multipliers to deal with external constraints and kinematic loops between the character and the environment. While Erleben's approach works well for unconstrained rigid bodies, for character motion articulated motion is required. The reduced-coordinate approach has the benefit of not suffering from constraint-drift between bodies and is therefore well-suited for character systems. The resulting hybrid technique adds clean constraint resolution at large simulation rates without sacrificing quality of the final character motion.

The linear complementarity problem (LCP) problem of Erleben [Erleben 2007] is given by:

$$\begin{aligned}
 u_{t+h} &= u_t + hJM^{-1}J^T\lambda - hJM^{-1}f_{ext} \\
 u_{t+h}^{(i)}\lambda^{(i)} &= 0 \quad \text{iff } \lambda_{l_o} < \lambda^{(i)} < \lambda_{h_i} \\
 u_{t+h}^{(i)}\lambda^{(i)} &< 0 \quad \text{iff } \lambda^{(i)} = \lambda_{h_i}^{(i)} \\
 u_{t+h}^{(i)}\lambda^{(i)} &> 0 \quad \text{iff } \lambda^{(i)} = \lambda_{l_o}^{(i)}
 \end{aligned} \tag{8}$$

where M is the $6k \times 6k$ symmetric, positive-definite, block diagonal matrix composed of k 6×6 rigid-body mass matrix elements; J is the constraint Jacobian which relates change of body coordinates to change in constraint error; λ , λ_{l_o} , λ_{h_i} are the constraint force and its bounds; u_t and u_{t+h} are the pre-step and post-step constraint error velocities; f_{ext} are coriolis and external body forces. To add constrained multibody support, we add a generalized $n \times n$ mass matrix for the kinematic chain to M as a block diagonal element and modify the constraint Jacobians to account for the generalized coordinates of the bodies. The LCP is then solved using the iterative projected Gauss-Seidel solver of Erleben which outputs constraint forces. These constraint forces are fed into the Featherstone forward dynamics algorithm to produce generalized accelerations, and semi-implicit Euler is used to integrate the resulting accelerations to update generalized velocities and positions.

References

- ABE, Y., LIU, C. K., AND POPOVIĆ, Z. 2004. Momentum-based parameterization of dynamic character motion. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 173–182.
- ABE, Y., DASILVA, M., AND POPOVIĆ, J. 2007. Multiobjective control with frictional contacts. *ACM SIGGRAPH/Eurographics symposium on Computer animation*, 249–258.
- COROS, S., BEAUDOIN, P., AND VAN DE PANNE, M. 2010. Generalized biped walking control. *ACM Trans. Graph.* 29, 4, Article 130.
- COROS, S., KARPATY, A., JONES, B., REVERET, L., AND VAN DE PANNE, M. 2011. Locomotion skills for simulated quadrupeds. *ACM Trans. Graph.* 30, 4, Article 59.
- DA SILVA, M., ABE, Y., AND POPOVIĆ, J. 2008. Interactive simulation of stylized human locomotion. *ACM Trans. Graph.* 27, 3, Article 82.
- DA SILVA, M., DURAND, F., AND POPOVIĆ, J. 2009. Linear bellman combination for control of character animation. *ACM Trans. Graph.* 28, 3, Article 82.
- DE LASA, M., MORDATCH, I., AND HERTZMANN, A. 2010. Feature-based locomotion controllers. *ACM Trans. Graph.* 29, 4, Article 131.
- ERLEBEN, K. 2007. Velocity-based shock propagation for multi-body dynamics animation. *ACM Trans. Graph.* 26, Article 12.
- FEATHERSTONE, R. 1987. *Robot Dynamics Algorithm*. Kluwer Academic Publishers.
- HA, S., YE, Y., AND LIU, C. K. 2012. Falling and landing motion control for character animation. *ACM Transactions on Graphics* 31, 6, Article 155.
- HODGINS, J. K., WOOTEN, W. L., BROGAN, D. C., AND O'BRIEN, J. F. 1995. Animating human athletics. In *Proceedings of SIGGRAPH 1995*, 71–78.
- KIM, J., AND POLLARD, N. S. 2011. Direct control of simulated nonhuman characters. *IEEE Computer Graphics and Applications* 31, 4, 56–65.
- KRY, P. G., REVERET, L., FAURE, F., AND CANI, M.-P. 2009. Modal locomotion: Animating virtual characters with natural vibrations. *Comput. Graph. Forum* 28, 2, 289–298.
- KWON, T., AND HODGINS, J. 2010. Control systems for human running using an inverted pendulum model and a reference motion capture sequence. In *ACM SIGGRAPH/Eurographics symposium on Computer animation*, 129–138.
- LEE, Y., KIM, S., AND LEE, J. 2010. Data-driven biped control. *ACM Trans. Graph.* 29, 4, Article 129.
- LIU, L., YIN, K., VAN DE PANNE, M., SHAO, T., AND XU, W. 2010. Sampling-based contact-rich motion control. *ACM Trans. Graph.* 29, 4, Article 128.
- LIU, L., YIN, K., VAN DE PANNE, M., AND GUO, B. 2012. Terrain runner: Control, parameterization, composition, and planning for highly dynamic motions. *ACM Trans. Graph.* 31, 6, Article 154.
- MACCHIETTO, A., ZORDAN, V., AND SHELTON, C. R. 2009. Momentum control for balance. *ACM Trans. Graph.* 28, 3, Article 80.
- MAJKOWSKA, A., AND FALOUTSOS, P. 2007. Flipping with physics: motion editing for acrobatics. In *ACM SIGGRAPH/Eurographics symposium on Computer animation*, 35–44.
- MORDATCH, I., DE LASA, M., AND HERTZMANN, A. 2010. Robust physics-based locomotion using low-dimensional planning. *ACM Trans. Graph.* 29, 4, Article 71.
- MUICO, U., LEE, Y., POPOVIĆ, J., AND POPOVIĆ, Z. 2009. Contact-aware nonlinear control of dynamic characters. *ACM Trans. Graph.* 28, 3.
- POPOVIC, M., HOFMANN, A., AND HERR, H. 2004. Angular momentum regulation during human walking: biomechanics and control. In *IEEE Int. Conf. Robotics and Automation*, vol. 3, 2405–2411.
- POPOVIC, M., HOFMANN, A., AND HERR, H. 2004. Zero spin angular momentum control: definition and applicability. *IEEE/RAS International Conference on Humanoid Robots*, 478–493.
- POPOVIC, M., GOSWAMI, A., AND HERR, H. 2005. Ground Reference Points in Legged Locomotion: Definitions, Biological Trajectories and Control Implications. *The International Journal of Robotics Research* 24, 12, 1013.
- RAIBERT, M. H., AND HODGINS, J. K. 1991. Animation of dynamic legged locomotion. In *Proceedings of SIGGRAPH '91*, 349–358.
- SOK, K. W., KIM, M., AND LEE, J. 2007. Simulating biped behaviors from human motion data. *ACM Trans. Graph.* 26, 3, Article 107.
- TAN, J., LIU, C. K., AND TURK, G. 2011. Stable proportional-derivative controllers. *IEEE Computer Graphics and Applications* 31, 4, 34–44.
- WAMPLER, K., AND POPOVIĆ, Z. 2009. Optimal gait and form for animal locomotion. *ACM Transactions on Graphics* 28, 3, Article 60.
- WANG, J. M., FLEET, D. J., AND HERTZMANN, A. 2009. Optimizing walking controllers. *ACM Trans. Graph.* 28, 5, Article 168.
- WOOTEN, W., AND HODGINS, J. 2000. Simulating leaping, tumbling, landing and balancing humans. In *IEEE Int. Conf. Robotics and Automation*, vol. 1, 656–662.
- WU, J.-C., AND POPOVIĆ, Z. 2010. Terrain-adaptive bipedal locomotion control. *ACM Trans. Graph.* 29, 4, Article 72.
- WU, C.-C., AND ZORDAN, V. 2010. Goal-directed stepping with momentum control. In *ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, 113–118.
- YE, Y., AND LIU, C. K. 2010. Optimal feedback control for character animation using an abstract model. *ACM Trans. Graph.* 29, 4, Article 74.
- YIN, K., LOKEN, K., AND VAN DE PANNE, M. 2007. Simbicon: Simple biped locomotion control. *ACM Trans. Graph.* 26, 3, Article 105.