Adaptive Dynamics With Hybrid Response

Nam H. Nguyen*
UC Riverside

Raul Arista†
UC Riverside

C. Karen Liu‡
Georgia Tech

Victor Zordan§
UC Riverside

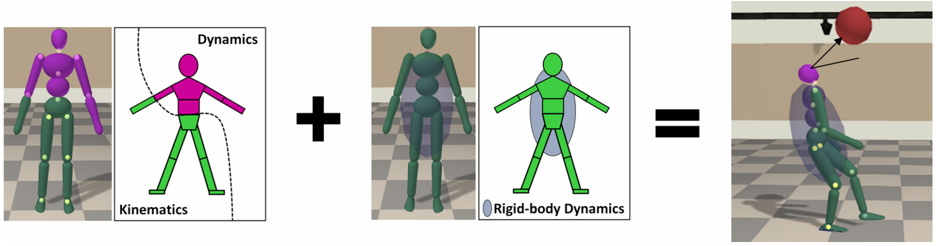

Figure 1: *Our technique layers adaptive kinematic/dynamics (left) over rigid-body dynamics (center) to create full-body interaction (right).*

Abstract

This paper describes a hybrid technique to produce physically plausible response for animated character. The approach includes the automatic, adaptive selection of the dynamic articulation necessary for local-body responses based on the changing conditions of the character. This method is combined with a whole-body reaction to disturbances which is computed and controlled separately. Combined, the resulting response effects the entire body as well as the local area of a disturbance without the need for an explicit full-body controller and the response computed is both deliberate (controlled) and passive (uncontrolled). The results showcase a variety of scenarios including fast and sustained contact.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation;

Keywords: character animation, physics based, real-time

1 Introduction

A plethora of control approaches have created a new wave of physics-based characters for a variety of behaviors [Yin et al. 2007; Abe et al. 2007; Macchietto et al. 2009; de Lasa et al. 2010; Lee et al. 2010; Coros et al. 2010; Wu and Popović 2010] among others. However, we have not yet seen adoption of these techniques in commercial media. While there are considerable benefits to be gained from the use of character physics, game developers have not yet found that the published approaches are suitable for commercial production. Two likely reasons are that the proposed character animation techniques are not robust enough for their needs and, also, such techniques do not fit well within existing pipelines.

In practice, game character animators often create the perception of physical interactivity through a combination of inverse kinematics (IK), simplified passive dynamics, and carefully crafted data examples that are played in place of physically computed interaction. In this paper, we draw from these and other techniques, but assemble a unique system in order to create a character that responds to interactions in a physically plausible manner. To this end, we introduce an *adaptive dynamics* approach that selects and simulates only the dynamics that are expressly necessary for the character at any given time. We couple this with a simplified representation of the character that controls whole-body position and orientation.

To adaptively control the dynamics of the articulation of the character, we employ a simplified inverse dynamics (ID) routine to analyze the character's state. The ID compares computed estimates of internal activation against the external influences felt by the character. When the disturbance forces lead to torques which are comparable to the internal torque estimates, the system "activates" the dynamics for a given joint, Figure 1 (Left). The benefit of this approach is twofold. It maintains high-level control over the character by treating the motion generation step as largely kinematic. In addition, the technique minimizes the need for a full-body controller and avoids complicated, possibly expensive calculations associated with control. The result is a physically based local response to a disturbance generated by changes in (only) the joints that are determined to be affected prominently by the disturbance.

In addition, we propose a whole-body response generator that computes a reaction through a rigid body approximation of the character, Figure 1 (Middle). This model influences the character's center of mass and global orientation to produce a physically based whole-body response to an interaction. The model includes a simple set of controls to maintain balance and keep the character upright. Further, a stepping controller leads the character to take reactive steps in response to interactions. The result (Figure 1, right) is a visually rich response that includes local and global effects but maintains guarantees consistent with current game pipelines, for example simple character control and preventing undesired falls.

1.1 Related Work

The approach of utilizing physical attributes to modify motion capture has been investigated in a variety of forms [Arikan et al. 2005; Yin et al. 2005; Sok et al. 2010], among others. The work presented here belongs to a specific growing sub-area within this topic — so-called *hybrid* techniques that directly combine kinematic and dynamic models for character animation. Of these, several approaches [Shapiro et al. 2003; Mandel 2004; Zordan et al. 2005] modulate between the use of kinematics and dynamics along the

*e-mail:nnguyen@cs.ucr.edu
†e-mail:raris002@ucr.edu
‡e-mail:karenliu@cc.gatech.edu
§e-mail:vbz@cs.ucr.edu

time axis, employing each as necessary based on the conditions of the character. Others [Ye and Liu 2008; Nguyen et al. 2010] splice the kinematics and dynamics across the body at a given instant in time. Our work includes features of both but in addition we treat the character as a layering of two models, the whole body and its articulations. As such, we have overlap with Ishigaki et al [2009] in that we both layer a whole-body model and an articulated model, although the specifics of each layer are unique in the respective efforts.

1.2 Overview

The layout of our system is drawn schematically in Figure 2. The physics of the character is modeled as the dual of two systems both switching automatically between kinematics and dynamics. One representation is the articulated body (AB) dynamics. The AB model (Section 2) creates a physical reaction driven through its dynamics and a joint-based tracking controller. It yields a local disturbance near the point of contact or impulse based on the collision handler. The second representation is the rigid-body (RB) model (Section 3) which has a persistent state and state derivative consisting of its own position and orientation. The RB model "feels" the influences of the full-body character and responds by determining a new whole-body orientation and location for the center of mass (CM) of the character. A specialized inverse kinematic (IK) module (Section 4) modifies the given reference pose to reflect the state of the RB model in the final (drawn) character state. In addition, a stepping supervisor assesses the state of the RB model and, if a step is necessary, it provides the RB with a target to follow as it also creates a trajectory for the stepping foot that the IK will then follow. Reference motion, $\bar{q}$, acts as the basic input to the system and drives the character (kinematically) when no external influences are present.

2 Adaptive Articulation

Our approach adaptively adjusts articulation much like the work of [Redon et al. 2005]. The animation of the AB model is produced with joint articulations that are driven by a mixture of kinematics and dynamics. Set $\mathbf{j^k}$ contains the set of joints that are kinematic, and these play motion directly either from a reference clip or a computed motion, modified by IK or stepping. Since the joints in the set $\mathbf{j^k}$ are controlled kinematically, these joints are never outside of the direct control of the animation engine making the system akin to the requirements of a standard game pipeline. The remaining joints, set $\mathbf{j^d}$, are influenced by the dynamics, including external disturbance forces. Of course, in a physical system with multiple joints, even a small disturbance will influence all the joints but limiting the articulation for $\mathbf{j^d}$ has several benefits, including creating the simplest system to control based on the conditions.

The adaptive dynamics module determines the list of joints in $\mathbf{j^d}$ based on the ones likely to yield substantive reactions to a given disturbance. Using inverse dynamics (ID), we approximate the internal activation torques for a reference motion offline and store the estimate for each joint. When a collision force or impulse is applied to the character, the torque influence of the external force at the specific joint is computed and its magnitude is compared to the internal torque estimate for the current frame of the reference. Specifically, the ID module compares three terms to determine the articulation: A which is the magnitude of the torque vector from the a priori analysis of the reference animation; B which contains the torques needed to compensate for gravity; and C, the torque vector due to the disturbance force. If $C \ll (A + B)$ we can ignore the influence of C for that joint because the torque will have a negligible influence over the joint's movement. If $C \sim (A + B)$ we add the articulation of that specific joint to the dynamics model since it will be influenced in a non-trivial manner by the disturbance. In our implementation, we test the condition $\epsilon < \frac{C}{(A+B)}$ and set ϵ to be 0.25 in the animation results in the accompanying video. If this test returns *true*, the joint is added to $\mathbf{j^d}$.

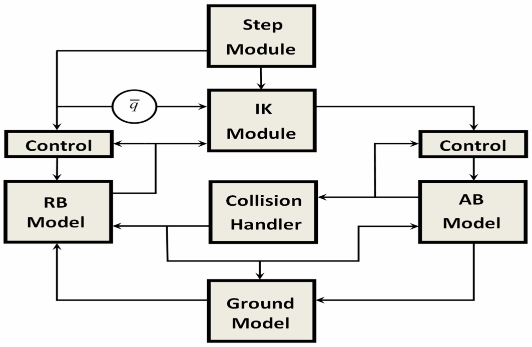

Figure 2: *Our system combines global and local effects into a single reaction through a rigid-body (RB) model and articulated body (AB) model respectively. The RB model influences the AB model through a modification of the CM and whole-body rotation in the reference motion, $\bar{q}$. The AB model influences the RB model through an approximation of the effective ground reaction force. Both the AB and RB models are influenced by external disturbance forces and gravity.*

While general purpose ID can be difficult to determine and ambiguous, we found a simple jointwise approximation of ID to suffice. Namely, given the reference motion, compute each joint's torque as $\tau_j = I_j \alpha_j$ based on the current frame's angular acceleration determined from the reference motion through finite difference. $I_j = \sum m_k {r_k}^2$ for all k outboard bodies of joint j rooted at the ground. To this estimate we add an additional torque for gravity compensation for each joint's outboard bodies based on their mass and the current torque arm. The transition from $\mathbf{j^k}$ to $\mathbf{j^d}$ happens in a single timestep.

While a joint is dynamically simulated, torque activation is computed through a joint tracker following the reference motion. Each joint in $\mathbf{j^d}$ employs a controller which follows the reference motion by tracking the joint angles of the motion capture data using internal joint torques, $\tau_\mathbf{i}$. Specifically, the joint tracker computes joint torques based on the tracking error $(\bar{q} - q)$ and joint velocities $(\dot{q})$ as

$$\tau_\mathbf{i} = k_q(\bar{q} - q) - d_q(\dot{q}) \tag{1}$$

which follows the motion $\bar{q}$ based on the values k_q and d_q. A single, inertia-scaled gain value is chosen for all gains which are set once and held constant for the duration of the animations throughout the system.

To close the loop on the adaptive dynamics switching, we employ a simple method to transition joints from $\mathbf{j^d}$ to $\mathbf{j^k}$. The tracking control force in Equation 1 drives joints in $\mathbf{j^d}$ to match the reference motion after the disturbance force is removed and the inertial effects dissipate. Once the tracker brings the joint close to the reference motion, within a given threshold, the system interpolates back to following the kinematics smoothly over a short fixed interval.

3 Whole-body Response

Whole-body response is computed through the RB model. Without it, local response creates a fairly unconvincing reaction, as seen in the video. But to create a whole-body reaction, we need to account for the gross physical effects as they apply to the body. Our RB model is influenced by a small set of factors, namely the external disturbance force, $\mathbf{F_e}$, the force due to gravity, $\mathbf{F_g} = mg$, where m is the total mass of the character and the ground reaction force, $\mathbf{F_r}$. The position, $\check{c}$, and orientation, $\check{\theta}$, of the RB model are updated through a forward Euler integration step of the equations of motion for a simple, unconstrained rigid body. (The notation ˇ is used to differentiate the RB model state from that of the AB model.)

With the aggregate influence of the forces described, we create a physically plausible passive whole-body response for the character. To further direct the character's whole-body response, we add two controls, $\mathbf{F_c}$ and τ_θ, to balance and right the RB model, respectively, (or more properly, to create the illusion of such control.) For balance, we use a Cartesian-based servo as

$$\mathbf{F_c} = k_c(\check{c} - \bar{c}) - d_c(\dot{\check{c}} - \dot{\bar{c}}) \tag{2}$$

where $\bar{c}$ and $\dot{\bar{c}}$ are computed center of mass and velocities derived from the reference motion. Gains k_c and d_c are manually tuned constants. The effect of this control is a virtual actuator [Pratt et al. 1997; Zordan and Hodgins 2002; Coros et al. 2010] for the CM, "pulling" it toward the reference motion. We compute τ_θ as

$$\tau_\theta = k_\theta(\check{\theta} - \bar{\theta}) - d_\theta(\dot{\check{\theta}} - \dot{\bar{\theta}}) \tag{3}$$

where $\bar{\theta}$ and $\dot{\bar{\theta}}$ are computed from the motion capture, and k_θ and d_θ are gain constants. This control creates a rotational response to the disturbance which stalls rotation and returns the character to upright after a short duration. In tuning, the gain values of these controllers are chosen such that they are large enough to be noticeable while not large enough to be overly dominant influences.

We note that the $\mathbf{F_e}$ term can be taken directly from the collision handler, but the ground reaction force of the AB model is never computed explicitly. Because the character is (at least in part) driven by kinematics, the ground reaction force is not defined. This issue arises due to the kinematic-driven links which act as irresistible influences that allow the dynamics to pull and push on them in an unbounded fashion. However, the effect of the ground can be inferred based on the accelerations of the character. The character sees finite linear and angular accelerations. From these we determine the aggregate generalized forces acting on the character and use these to determine $\mathbf{F_r}$ for the purposes of the RB model. In our implementation, we compute $\mathbf{F_r}$ in this manner and further threshold its value such that its up-normal component is greater than zero and its horizontal component is less than the friction coefficient scaled by the normal.

The output of the RB model is a new position and angular acceleration for the rigid body based on the described force and torque influences. The RB state is integrated with a simple Euler step to determine new velocities $\dot{\check{c}}$ and $\dot{\check{\theta}}$ and then the new positions $\check{c}$ and $\check{\theta}$. We feed this update to the rest of the system through the IK module which modifies the current reference frame to reflect the new state Note, this modified IK output, $\tilde{q}$, replaces $\bar{q}$ in Equation 1. As in the AB model, after the disturbance is over, the RB model moves back to kinematics by setting $\check{c} = \bar{c}$ and $\check{\theta} = \bar{\theta}$ after a smooth interpolation over a small time window. When $\mathbf{F_e} > 0$, the RB dynamics is initiated instantly.

4 Inverse Kinematics

The RB model state is passed to the IK system which is specialized for its specific task. At a high level, the IK modifies the original motion to follow the RB state through a series of simple modifications. The rotation is computed as a desired angular offset from the previous frame, taken from the state, q, of the AB model to the new desired RB state. We employ a pseudo-inverse Jacobian similar to [Coros et al. 2010] to compute the desired rotation. Next, the IK updates the CM position based on the RB state change. In order to do this, it starts with a selected support foot as the root of the chain for the kinematics. It uses a simple gradient descent to modify the angles of the leg up the chain from the root to the the pelvis in order to shift the original motion's CM to the target CM. Finally, the position of the second foot is set to its desired position and a short IK chain is found analytically for the leg of this foot. The desired position is set based on the reference motion.

Note, we opt to employ this custom IK solution because it is streamlined for our needs and runs very quickly. As described, our system runs at interactive rates on a modern processor. More general approaches, such as the reconstruction process proposed by Ye [Ye and Liu 2010], may lead to higher quality results in applications which are less time critical. With our focus on contributing to game technology, such an approach would be too slow, and instead, the IK we describe allows us to keep the system running quickly.

5 Implementation details

We use ODE (www.ode.org) for the dynamics and its collision handler to manage all collisions. Based on the influences of the environment, the character's body parts move repeatedly from kinematics to dynamics and back. And, as such, we employ the approach described in [Nguyen et al. 2010] to seamlessly merge the dynamics and kinematics models. To prevent the dynamics articulation from growing too quickly, we induce a time delay such that the set $\mathbf{j^d}$ does not change quickly. This protects from shocks that might lead to full dynamics when the system can "manage" the existing disturbance with a more modest articulation. After a small time window (100 ms) it adds additional articulated joints to the dynamics as needed.

Because our system has complete control we can easily put desired limits on the various targets, the number of articulations, and/or prevent the support body (or bodies) from switching to dynamics. We tested many scenarios before arriving upon the one shown in the videos. A few important factors that we opted to include are as follows: 1) the support leg (defined by the one closest to projected CM) must remain kinematically driven; 2) limits on the RB state offset from the reference motion is capped, both in translation and rotation; 3) twist rotation about the vertical axis for the RB model is ignored by the IK to allows the IK to be computed more easily. Although different results of such choices can change the realism of the system, control over the character ultimately is left in the hands of the game developers and this is a primary goal for this effort.

We add a stepping routine that monitors the whole-body state as defined by the RB model and computes steps if it is required. Reactive stepping utilizes the stepping technique outlined by Wu and Zordan [Wu and Zordan 2010]. A supervisor initiates a step based on the RB state and selects the foot and its placement as well as constructs new trajectories for the CM and the stepping foot. In our system, the trajectory of the CM replaces that of the RB model and we add the constructed foot trajectory as an additional IK objective.

6 Conclusions

We present a technique that is unique in its treatment of dynamics and kinematics to produce the appearance of controlled, physical response in characters (See Figure 3). Notably, when there is no physical interaction, the character is fully kinematic and follows a reference motion perfectly. But, when a disturbance is introduced, the system automatically adjusts the number of physically

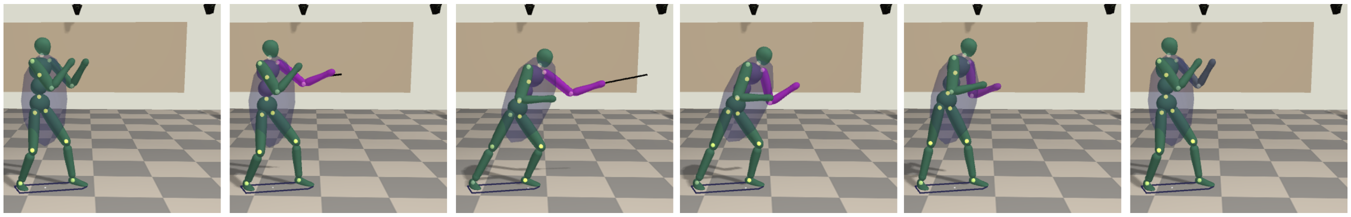

Figure 3: *Response to a sustained force interaction. Note, here and in the video, the ID estimate is visualized through the color saturation of the joint, bright yellow indicates that a strong torque is present in the joint.*

simulated joint articulations through an inverse dynamics analysis and produces a response that is physically based in both gross-body movement and locally using the chosen dynamics articulation. Taken as a whole, our system provides several innovations over previous work. Our novel effort in creating an adaptive dynamic articulation minimizes the complexity of the dynamics and keeps the character control as simple as possible. One contribution here is in presenting a straightforward solution for determining the appropriate articulation of the character automatically. The result is simplicity and savings in computation both in the dynamics computation and in the control. Further the character can remain in complete control unless full-body (or ragdoll) dynamics is ultimately necessary. In this fashion, this solution is aligned with the basic requirements of most game engines. Finally, we offer a contribution through our unique hybrid response. To our knowledge, no other technique has combined two physical controllers that separate the dynamic effects of an impact into their global and local parts. In our system, we do not explicitly compute control for an underactuated full-body character, but instead create the appearance of such. We are enthusiastic about the implications of this form of simplistic control for games and other commercial applications and we hope to explore advances in this direction in our future work.

Acknowledgments

We acknowledge Nkenge Wheatland for her assistance in early developments of the project, and GAANN, Award No: P200A090077 for funding the first author.

References

ABE, Y., DA SILVA, M., AND POPOVIĆ, J. 2007. Multiobjective control with frictional contacts. In *SIGGRAPH/Eurographics Symposium on Computer Animation*.

ARIKAN, O., FORSYTH, D. A., AND O'BRIEN, J. F. 2005. Pushing people around. In *SIGGRAPH/Eurographics Symposium on Computer Animation*, 59–66.

COROS, S., BEAUDOIN, P., AND VAN DE PANNE, M. 2010. Generalized biped walking control. *ACM Trans. on Graphics 29*, 4 (July), 130.

DE LASA, M., MORDATCH, I., AND HERTZMANN, A. 2010. Feature-based locomotion controllers. *ACM Trans. on Graphics 29*, 4 (July), 131.

ISHIGAKI, S., WHITE, T., ZORDAN, V., AND LIU, K. C. 2009. Performance-based control interface for character animation. *ACM Trans. on Graphics 28*, 3.

LEE, Y., KIM, S., AND LEE, J. 2010. Data-driven biped control. *ACM Trans. on Graphics 29*, 4 (July), 129:1–129:8.

MACCHIETTO, A., ZORDAN, V., AND SHELTON, C. 2009. Momentum control for balance. *ACM Trans. on Graphics 28*, 3, 80.

MANDEL, M., 2004. Versatile and interactive virtual humans: Hybrid use of data-driven and dynamics-based motion synthesis. *Master's Thesis, Carnegie Mellon University*.

NGUYEN, N., WHEATLAND, N., BROWN, D., PARISE, B., LIU, C. K., AND ZORDAN, V. 2010. Performance capture with physical interation. In *2010 ACM SIGGRAPH / Eurographics Symposium on Computer Animation*.

PRATT, J., DILWORTH, P., AND PRATT, G. 1997. Virtual model control of a bipedal walking robot. In *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*, vol. 1, IEEE, 193–198.

REDON, S., GALOPPO, N., AND LIN, M. C. 2005. Adaptive dynamics of articulated bodies. *ACM Trans. on Graphics 24*, 3 (July), 936–945.

SHAPIRO, A., PIGHIN, F., AND FALOUTSOS, P. 2003. Hybrid control for interactive character animation. In *Pacific Graphics*.

SOK, K., YAMANE, K., LEE, J., AND HODGINS, J. 2010. Editing dynamic human motions via momentum and force. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 11–20.

WU, J.-C., AND POPOVIĆ, Z. 2010. Terrain-adaptive bipedal locomotion control. *ACM Trans. on Graphics 29*, 4 (July), 72.

WU, C., AND ZORDAN, V. 2010. Goal-directed stepping with momentum control. In *SIGGRAPH/Eurographics Symposium on Computer Animation*, 113–118.

YE, Y., AND LIU, C. K. 2008. Responsive characters with dynamic constraints in near-unactuated coordinates. *ACM Trans. on Graphics 27*, 5 (Dec.).

YE, Y., AND LIU, C. K. 2010. Optimal feedback control for character animation using an abstract model. *ACM Trans. on Graphics 29*, 4 (July), 74.

YIN, K., PAI, D., AND VAN DE PANNE, M. 2005. Data-driven interactive balancing behaviors. *Pacific Graphics*.

YIN, K., LOKEN, K., AND VAN DE PANNE, M. 2007. Simbicon: Simple biped locomotion control. *ACM Trans. on Graphics 26*, 3, 105.

ZORDAN, V., AND HODGINS, J. 2002. Motion capture-driven simulations that hit and react. In *SIGGRAPH Symposium on Computer Animation*, ACM Press, 89–96.

ZORDAN, V., MAJKOWSKA, A., CHIU, B., AND FAST, M. 2005. Dynamic response for motion capture animation. *ACM Trans. on Graphics 24*, 3 (July), 697–701.