Joint-Torque Control of Character Motions

Active Animations

Ben Kenwright

Abstract—We want to go beyond "passive rag-doll like" simulated characters towards more "active" intelligent self-driven solutions. The "puppet on strings" approach lacks dynamic interactive properties for engaging realistic and immersive virtual environments. This paper focuses on physics-based "self-driven characters" (e.g., character's own joint-torques to control and move the limbs to accomplish specific tasks, such as walking) that can react in a life-like manner using physical properties (e.g., limb-strength, ground contacts, and mass). We start by explaining the kinematic principles of motion-control of articulated figures. The two types of motion-control for virtual characters are 'position-based' and 'torque-based'. The position-based approach focuses on the linear inverse kinematic model, while the force-based approach extends the principle to generalized joint-torques. Primarily, the relationship between end-effectors (e.g., fingers and feet) current and desired targets to joint orientations and torques. We exploit the highly complex and ambiguous nature of an articulated character to produce primary and secondary cases. For example, projecting joint-torques onto the null-space transformation. This discusses and explains the principles and advantages of active torque controlled character animation solutions, including their dynamics and limitations.

Index Terms—joints, torque, force, motion, procedural, ragdoll, puppet, interactive, balancing, character, animation, physics-based, responsive, adaptive, dynamic, 3D, video-games, inverted-pendulum, joint-torque, controllable

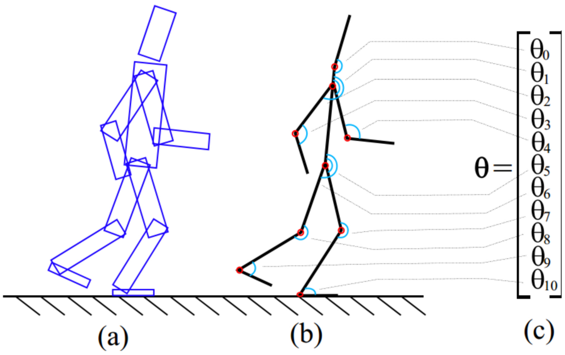


Fig. 1. **Virtual Character Joints** - Decomposition of a virtual character into rigid body joints (e.g., cubes) and relative angular orientations (i.e., parent-joint angle).

1 INTRODUCTION

The days of repetitive pre-recorded key-framed animation solutions are coming to an end. Research across multiple fields, such as biomechanics, robotics, and graphics, are now developing more intelligent self-driven techniques based-on physical principles to mimic human movements in a new life-like way, to produce a fresh generation of immersive, engaging, and interactive character solutions.

For example, a crucial movement that is indispensable in multiple fields, such as graphics and robotics, is simulating balance recovery in a life-like and realistic way. Nevertheless, creating interactive virtual characters that react to random unforeseen disturbances and are able to recover from them (i.e., regain their balance) in a human-like way that mimics the real-world is challenging and interesting [22], [8], [12], [23], [1].

- *Ben Kenwright*
  E-mail: bkenwright@xbdev.net

The human-muscle is analogous to a spring-damper system [7]. We can calculate specific gain and damping coefficients to mimic specific muscle responses while actively controlling joint-torque movement by means of current and desired angular error.

1.1 Control

One specific problem for virtual characters is the kinematic issues associated with control of the final motions (i.e., task specific movements, such as placing the foot at a precise location at a certain time while following a specified trajectory).

For physics-based character solutions, uncomplicated angular-springs are used (i.e., active-joint torque servos), which affect the orientation and angular velocity of the character's body. Determining a specific set of joint torques and trajectories as inputs to the angular-spring servos to solve the specified task is a crucial issue.

1.2 Inverse Kinematic Solution

Inverse kinematics allows us to analyze, design, simulate, and plan virtual character movements. However, due to the complex nature of human based virtual characters it can be computationally expensive and complex to accomplish in real-time systems, such as games. For instance, a common situation that requires inverse kinematics regularly in character based games is the adaptation of pre-canned animations to accommodate certain tasks to appear more realistic (e.g., feet striking the ground and not slipping and arms touching object they are holding).

As a side note, while we focus on the global iterative approach (i.e., the Jacobian technique), there are, however, other techniques, such as, the cyclic coordinate descent (CCD) technique [5], that can solve IK problems faster and are more viable for real-time environments. The CCD technique is one such popular alternative, that works in a heuristic manner to provide a computationally fast, algorithmically simple, and straight-forward technique for generating IK solutions for interactive systems (e.g., gaming environments).

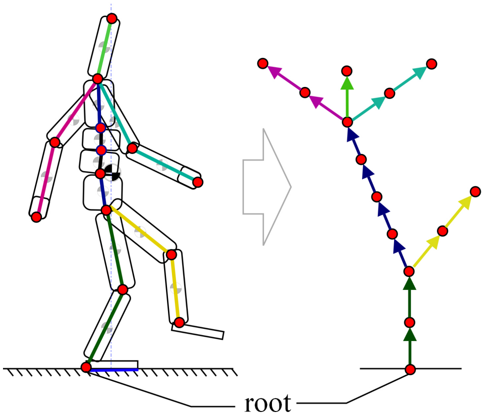


Fig. 2. **Articulate Character Tree Structure** - The interconnected limbs of the character form a hierarchical tree-like structure.

1.3 State-of-the-art

Virtual character-based control mechanics exploit techniques from numerous disciplines, such as robotics, biomechanics, and mathematics. The concept in this paper focuses on joint-orientation and position control based on kinematic principles that are extended to include torque-based motion control techniques. Precise control of joint-torques and transition speeds are essential. This task, however, is highly challenging due to the non-linearities, computational complexity, numerical errors, and approximation assumptions.

Despite the numerous challenges, we can achieve a dynamic solution that overcomes the many limitations. In particular, joint control for controlling body torques. The most popular approach is with Proportional-Derivative (PD) controllers. PD controllers, however, ignore dynamic interaction between joints and disturbance rejection depends upon the use of large gains and damping coefficients. Nevertheless, they are intuitive to understand and computationally fast; hence, ideal for real-time interactive simulations, such as games, where we are interested in visual aesthetic results compared to numerical accuracy.

Physics-based virtual character joint-torque control is indispensable not only for attaining highly dynamic solutions, but is also important for solving interactive situations. Joint-torque control techniques combined with active solutions (e.g., balanced life-like motions) are an emergent means of extending character capabilities so they are engaging, interactive, and believable.

1.4 Trends (Today and Tomorrow)

We have seen recent trends to create character animation techniques that are more autonomous and intelligent based on smart solutions. Active research in the area of computer graphics suggests that a new generation of animated virtual characters will soon be appearing (e.g., procedural biomechanically inspired self-driven avatars). These emerging technologies with new inspiring capabilities will focus on physics-based approaches with active joint-torque controlled motions to produce natural life-like characters.

1.5 Challenges

So why is it so to reproduce life-like human movements in real-time 'and' without key-frame data? Why has it eluded us for so long? We give the main reasons here:

Realism is particularly difficult, as a particular character model gives rise to a large set of possible motions with different styles. Even if robust and stabilizing control laws can be found, it is challenging to construct those that reproduce the intricate and agile movements we observe in nature.

Then there is model complexity, since a character can have an extremely high number of degrees of freedom, making the search for the appropriate control parameters hard (e.g., adult human body has over 200 hundred bones). Although continuous numerical optimizations can cope with large search spaces, the *stringent demands of interactive applications* make it clear that optimization cannot solely be performed at the time control is needed.

The discontinuous non-linear character work-space (e.g., joint limits and contacts) restricts movement within a certain region of three-dimensional space; these constraints are difficult to maintain in real-time simulation systems, such as games. Furthermore, frequent ground contacts create a highly discontinuous search space rendering most continuous controller synthesis methods ineffective at planning over longer time horizons.

Dynamically simulated characters are difficult to control because they have no direct control over their global position and orientation (i.e., underactuation). Even staying upright is a challenge for large disturbances. In order to succeed, a control law must plan ahead to determine actions that can stabilize the body [15].

2 Related Work

How have these challenging problems of generating life-like interactive characters been solved before? Which methods have made us 'sit-up' and take notice and why? This is what we aim to address in this section.

To begin with, there are, the manually designed physics-based biped balance controllers [25], [20], which can be optimized for robust behaviors [23], [1], or combined with other techniques (e.g., motion capture data) to produce hybrid solutions with life-like dynamic responses [4]. While controller-based approaches are often intuitive and computationally fast and robust, they can be cumbersome to tune (i.e., the different parameters) and produce only simple motions (e.g., balanced standing and walking). Then again, solutions to these problems have been proposed, such as an automatic search-based algorithms to tune controller parameters automatically [22] based on velocity tracking of motion capture data.

Data-driven solutions are able to adapt character poses so they transition seamlessly [17], mix animation sequences [24] or modify sequences so they account for changing environments and disturbances (e.g., pushes) [21]. However, data-drive methods cannot generate unique motions, and highly depend on the input motion capture data.

This paper adopts a 'non data-driven' technique with a controller-based solution, that is a procedural physics-based controller approach. Where the physics-based model ensures the movement is physically plausible, the controller gives the model a goal (e.g., balanced stepping, walking, steering), and the procedural aspect provides intermediate transition solutions, such as behavioral aspects that include how the character is walking, standing, and looking around.

It is interesting to note, that if we taking a look at biomechanical research, in the area of balance, there is a reaction time for humans responding loss of balance [14], [13], [16].

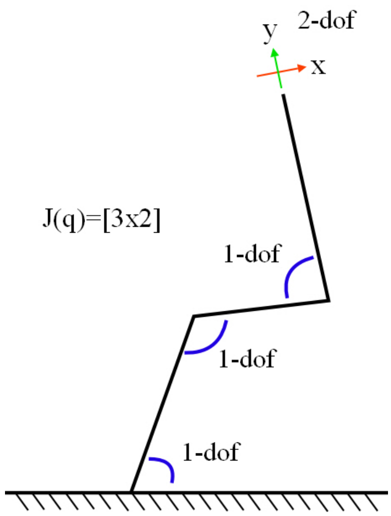


Fig. 3. **Degrees-of-Freedom Example** - Example showing the Jacobian dimensions based on the degrees-of-freedom for a simple interconnected set of links.

Going beyond basic stepping for balance recovery, it should be mentioned, that there are other methods of re-gaining balance, that include, grasping, stretching, and squatting strategies demonstrated in other work [2], [19], [3]. However, this paper demonstrates exploits stepping and ankle torque as a means of balance recover and control [9], [6].

Pratt et al. [18] presents a clear and interesting explaination of how virtual actuator control can be used to manipulate and control end-effectors.

Khatib and Burdick [11] explained motion control principles; later Khatib [10] also discusses force-based motion control for robot manipulators.

3 Position/Orientation Kinematic Control

3.1 Forward Kinematic

A virtual human biped character has a large number of degrees-of-freedom with the inverse kinematic problem being computationally complex. The problem is solved by linearizing the kinematic problem for small changes. This enables us to form a relationship between a change in joint-angles and the associated overall body end-effectors (e.g., feet and hands), as given below in Equation 1

$$\Delta x = J(q)\Delta q \tag{1}$$

where Δx represent a change in end-effector positions (and/or orientations), Δq represents the associated change in orientation of the joints, and $J(q)$ is the Jacobian matrix. For an *n-degree-of-freedom* articulated character with each end-effector operating in an *m-dimensional* space, $J(q)$ is an $n \times m$ matrix. For example, in 2D, if we have an end-effector with just position (i.e., x-y movement) 2-dof, while the interconnected body is composed of a singly-linked chain of connected elements (e.g., three-links), the Jacobian dimensions would be $[3 \times 2]$, as shown in Figure 3

3.2 Inverse Kinematic

For small changes Equation 1 forms a linear relationship between changes in end-effector position and joint-angles. We can calculate the skeleton posture and motion control solution by re-arranging

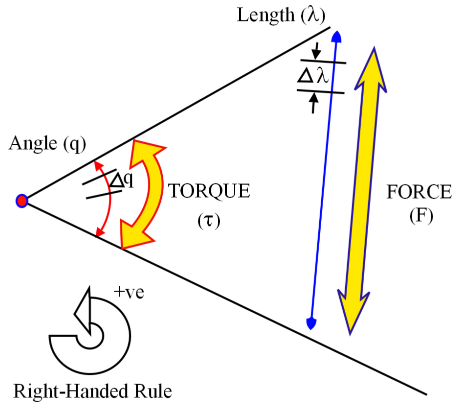


Fig. 4. **Lever Example** - We can easily associate small changes in joint angle with small changes in end-effector position (right-handed convention).

the equation (i.e., calculating the inverse). The inverse of Equation 1 produces a linear relationship, given below in Equation **??**.

$$\Delta q = J^{-1}(q)\Delta x \tag{2}$$

The trajectories for continuous movements of the articulated joints is achieved by incrementally moving the current target end-effector position to specific locations (i.e., $q + \Delta q$).

Note, in a non-redundant system, $n = m$, however, this is really never the case.

3.2.1 *Exploiting Inverse Kinematic Ambiguity and Redundancy*

The highly complex nature of articulated characters can produce an infinite number of solutions to specific inverse kinematic problems. We can, however, use this to our advantage to solve multiple problems with varying priority. This can be accomplished by means of the "pseudo-inverse", which can be calculated using a general inverse algorithm, as shown below in Equation **??**.

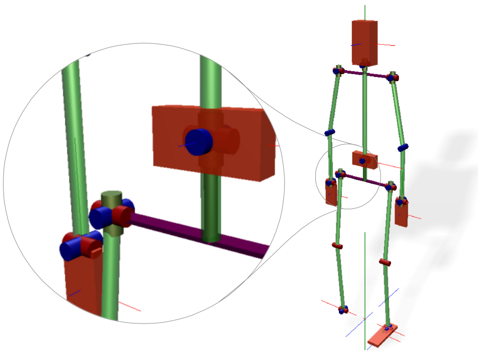
$$\Delta q = J^{+}(q)\Delta x + [I - J^{+}(q)J(q)]\Delta q_g \tag{3}$$

where I is an identity matrix with the same dimensions as $J(q)$, q_g represents an secondary goal arbitrary vector with the dimensions the same as q. The matrix $[I - J(q)^{+}J(q)]$ defines the "null-space" region associated with the "pseudo-matrix" $J^{+}(q)$, while $[I - J^{+}(q)J(q)]\Delta q_g$ represents the "zero-deviation" of position and orientation for the end-effector target. The additional secondary null-space calculation allows us to prioritize motions based on some "minimizing criteria" (e.g., obstacle avoidance, balance, or comfort).

4 Force/Torque-Based Kinematic Control

The inverse kinematic problem is represented by a set of linear equations that we solve using computationally straightforward linear system techniques (e.g., general matrix inverse approaches). We can, however, extend the approach to encapsulate torque-based control problems. Whereby, end-effector positions/orientations are replaced with forces/torques, and the generalized system of connected joint-angles is replaced by a set of joint-torques. The basic relationship between end-effector forces and joint-torques is given below in Equation 4.

$$\mathcal{T} = J^{T}(q)F \tag{4}$$



Joint	DOF
Head	3
Shoulder	3
Elbow	1
Hand	2
Pelvis	3
Hip	3
Knee	1
Foot	2
Total	**30**

Fig. 5. **Degrees-of-Freedom** - A simplified example of a character with joints. Show the degrees-of-freedom for each limb. The right foot is the base of the character hierarchy.

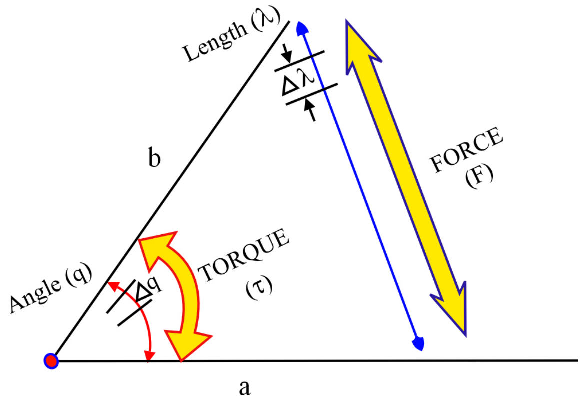


Fig. 6. **Lever Force-Torque Relationship** - We can relate joint-torque and end-effector force.

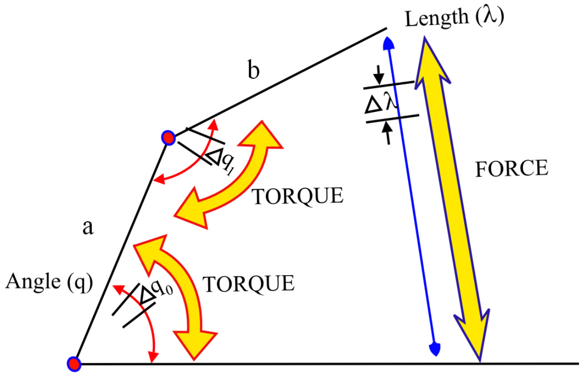


Fig. 7. **Two-Lever** - Analagous to the simple single degree-of-freedom level we can see that multiple joints can be associated with a single change in end-effector position/force.

4.1 Exploiting Ambiguity and Redundancy

Similar to Section 3.2.1 that exploited redundancy for multiple inverse kinematic problems to achieve a priority-based solutions, we can accomplish the same goal with torque-based inverse kinematic system. For articulated systems where $m > n$ we can have an infinite number of displacement solutions. We can, however, use the redundant null-space associated with the Jacobian inverse to ensure the end-effector goal is met, while secondary goals are accomplished. Hence, an infinite number of joint-torque deviations can be applied without effecting the end-effector forces/torques. These secondary joint-torques act along the directions within the null-space.

The virtual joint displacement is:

$$\Delta q = J^{+}(q)\Delta x + [I - J^{+}(q)J(q)]\Delta q_g \tag{5}$$

Knowing that the virtual work displacement:

$$\Delta W = \Delta W_0 + \Delta W_1 \tag{6}$$

with

$$\Delta W_0 = [J^{+T}(q)\mathcal{T}]^T \Delta x \tag{7}$$

and

$$\Delta W_1 = ([I - J^{+}(q)J(q)]^T \mathcal{T})^T \Delta q_g \tag{8}$$

where W_0 is the end-effector virtual work, and W_1 is the null-space virtual work for the joint-torque displacement associated with the pseudo-inverse $J^{+}(q)$. Hence, the relationship between end-effector forces and joint-torques is shown below in Equation 9.

$$\mathcal{T} = J^T(q)F + [I - J^T(q)J^{+T}(q)]\mathcal{T}_g \tag{9}$$

where $\mathcal{T}_g$ is an arbitrary set of joint-torques.

While a vector F controls the end-effector goal, the joint-torque vector $\mathcal{T}_g$ controls secondary internal joint motions. We can, for example, make $\mathcal{T}_g$ the gradient potential with its minimum at the desired postural position.

5 Conclusion and Discussion

Multiple factors, such as computational speed, naturalness, realism, and robustness all contribute towards what makes an animation system a viable solution. Creating an animation system that tackles each of these challenges is difficult and important. In practice, as addressed in this paper, the challenges can be subdivided into individual problems (e.g., balanced stepping, arm movement, and behavioral random motions) that are solved separately. This makes a modular solution that gives the end user the ability to swap and change different components and create a system that meets their particular demands.

The control of the characters articulated physics-based posture is important. While inverse-kinematic techniques in conjunction with dynamic torque-based approaches allows complex motions to be generated. Exploiting redundancy through priory based approaches solves complex tasks, such as, end-effector goal location, trajectory tracking, collision-avoidance, joint-limit enforcement.

We have explained the synthesis of control motions using orientation and torque based techniques using kinematic principles. Torque-based approaches use active forces to achieve the final pose. This in conjunction with a physics-based articulated rigid body skeleton produces a unified solution which is important for dynamically coupling the stability and control of the end-effector movements (e.g., feet and hands).

Acknowledgments

A special thanks to readers for taking the time to contact me and provide insightful, invaluable comments and suggestions to help to improve the quality of this article.

References

[1] Stelian Coros, Philippe Beaudoin, and Michiel van de Panne. Robust task-based control policies for physics-based characters. In *ACM*

SIGGRAPH Asia 2009 papers, SIGGRAPH Asia '09, pages 170:1–170:9, New York, NY, USA, 2009. ACM. 1, 2

[2] Alexander CH Geurts, Mirjam de Haart, Ilse JW van Nes, Jaak Duysens, et al. A review of standing balance recovery from stroke. *Gait & posture*, 22(3):267–281, 2005. 3

[3] At L Hof. The equations of motion for a standing human reveal three mechanisms for balance. *Journal of biomechanics*, 40(2):451–457, 2007. 3

[4] Sumit Jain, Yuting Ye, and C. Karen Liu. Optimization-based interactive motion synthesis. *ACM Trans. Graph.*, 28(1):10:1–10:12, February 2009. 2

[5] Ben Kenwright. Inverse kinematics ? cyclic coordinate descent (ccd). *Journal of Graphics Tools*, 16(4):177–217, 2012. 1

[6] Ben Kenwright. Responsive biped character stepping: When push comes to shove. In *Proceedings of the 2012 International Conference on Cyberworlds*, volume 2012, pages 151–156, 2012. 3

[7] Ben Kenwright. Synthesizing balancing character motions. *environments*, 15(3):113–123, 2013. 1

[8] Ben Kenwright, Richard Davison, and Graham Morgan. Dynamic balancing and walking for real-time 3d characters. In *Proceedings of the 4th international conference on Motion in Games*, MIG'11, pages 63–73, Berlin, Heidelberg, 2011. Springer-Verlag. 1

[9] Ben Kenwright, Richard Davison, and Graham Morgan. Dynamic balancing and walking for real-time 3d characters. In *Proceedings of the 2011 Motion in Games Conference*, volume 2011, pages 63–73. Springer Berlin/Heidelberg, 2011. 3

[10] Oussama Khatib. Force-based motion control of robot manipulators. In *Proceedings International Meeting on Advances in Robotic Kinematics, Ljubljana, Yugoslavia*, pages 176–180, Sept 1988. 3

[11] Oussama Khatib and Joel Burdick. Motion and force control of robot manipulators. In *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*, volume 3, pages 1381–1386. IEEE, 1986. 3

[12] Adriano Macchietto, Victor Zordan, and Christian R. Shelton. Momentum control for balance. In *ACM SIGGRAPH 2009 papers*, SIGGRAPH '09, pages 80:1–80:8, New York, NY, USA, 2009. ACM. 1

[13] Lida Mademli, Adamantios Arampatzis, and Kiros Karamanidis. Dynamic stability control in forward falls: postural corrections after muscle fatigue in young and older adults. *European journal of applied physiology*, 103(3):295–306, 2008. 2

[14] RC Miall, DJ Weir, DM Wolpert, and JF Stein. Is the cerebellum a smith predictor? *Journal of motor behavior*, 25(3):203–216, 1993. 2

[15] Uldarico Muico, Yongjoon Lee, Jovan Popović, and Zoran Popović. Contact-aware nonlinear control of dynamic characters. In *ACM SIGGRAPH 2009 papers*, SIGGRAPH '09, pages 81:1–81:9, New York, NY, USA, 2009. ACM. 2

[16] Richard R Neptune, Felix E Zajac, and Steven A Kautz. Muscle mechanical work requirements during normal walking: the energetic cost of raising the body's center-of-mass is significant. *Journal of biomechanics*, 37(6):817–825, 2004. 2

[17] Nancy S Pollard and Victor Brian Zordan. Physically based grasping control from example. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 311–318. ACM, 2005. 2

[18] Jerry Pratt, Ann Torres, Peter Dilworth, and Gill Pratt. Virtual actuator control. In *Intelligent Robots and Systems' 96, IROS 96, Proceedings of the 1996 IEEE/RSJ International Conference on*, volume 3, pages 1219–1226. IEEE, 1996. 3

[19] Xingda Qu and Maury A Nussbaum. Evaluation of the roles of passive and active control of balance using a balance control model. *Journal of biomechanics*, 42(12):1850–1855, 2009. 3

[20] Takaaki Shiratori, Brooke Coley, Rakié Cham, and Jessica K Hodgins. Simulating balance recovery responses to trips based on biomechanical principles. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 37–46. ACM, 2009. 2

[21] Kwang Won Sok, Manmyung Kim, and Jehee Lee. Simulating biped behaviors from human motion data. In *ACM SIGGRAPH 2007 papers*, SIGGRAPH '07, New York, NY, USA, 2007. ACM. 2

[22] Yao-Yang Tsai, Wen-Chieh Lin, Kuangyou B. Cheng, Jehee Lee, and Tong-Yee Lee. Real-time physics-based 3d biped character animation using an inverted pendulum model. In *IEEE Transactions on Visualization and Computer Graphics*, volume 16:2, pages 325–337, 2010. 1, 2

[23] Jack M. Wang, David J. Fleet, and Aaron Hertzmann. Optimizing walking controllers. In *ACM SIGGRAPH Asia 2009 papers*, SIGGRAPH Asia '09, pages 168:1–168:8, New York, NY, USA, 2009. ACM. 1, 2

[24] Yuting Ye and C Karen Liu. Synthesis of responsive motion using a dynamic model. In *Computer Graphics Forum*, volume 29, pages 555–562. Wiley Online Library, 2010. 2

[25] KangKang Yin, Kevin Loken, and Michiel van de Panne. Simbicon: simple biped locomotion control. In *ACM SIGGRAPH 2007 papers*, SIGGRAPH '07, New York, NY, USA, 2007. ACM. 2