Chapter 8
Responsive Motion Generation

Sukwon Lee and Sung-Hee Lee

Abstract In this chapter, we discuss the generation of natural behaviors of humanoids (virtual human characters in particular) responsive to the physical interaction with the user such as push and pull. These physical interactions play an important role for increasing the level of immersion of the user and lay foundations for more advanced level of interactions. One of the key components for physical interaction is the generation of suitable balancing behaviors of humanoids against user inputs. We review three major approaches for humanoid balancing, namely the ZMP-based methods, data-driven methods, and momentum-based methods. For each method, we discuss its basic ideas and principles, exemplar work, as well as important future research directions.

8.1 Introduction

A humanoid takes the form of a virtual human character in computer animation, and the form of a physical humanoid robot in robotics. In both fields of research the generation of humanoid motion remains a central problem, and the techniques developed in one field of research have been smoothly adopted in the other. In this chapter we focus on the motion generation problem for virtual human characters for computer animation, but many techniques discussed here originated from and can be applied to humanoid robots as well.

Rapid advancement of computing technologies allows more and more complex algorithms to be incorporated into real-time applications. In addition, natural user inputs based on gestures, gaze, and physiological signals of the users have been employed increasingly for human–computer interaction. These technical innovations trigger and require the development of computational methods to generate humanoid motions interacting with natural user inputs, which are characterized by their large degrees of freedom as opposed to a predefined set of inputs available from

S. Lee · S.-H. Lee (✉)
Korea Advanced Institute of Science and Technology, Daejon, Korea
e-mail: sunghee.lee@kaist.ac.kr

N. Magnenat-Thalmann et al. (eds.), *Context Aware Human-Robot and Human-Agent Interaction*, Human–Computer Interaction Series,
DOI 10.1007/978-3-319-19947-4_8

conventional user input devices such as keyboards and joysticks. Increased level of interaction between humanoids and users widens the scope for VR content and enhances immersiveness of VR applications.

Among the broad possibilities of interaction between the user and the humanoids, in this chapter we discuss the generation of natural behaviors of humanoids responsive to physical interaction with the user through (virtual) contacts such as push and pull. These physical interactions play an important role in giving an impression to the users that humanoids share the same physical space with users and thus laying the foundations for more advanced level of interactions.

Specifically, in this chapter we present techniques regarding the balancing behaviors of the whole body. Among a number of behaviors that a human can employ to maintain balance, the two most representative strategies are postural balancing and reactive stepping as shown in Fig. 8.1. Postural balancing is usually chosen for relatively short and mild perturbations against which a human can maintain balance in place simply by rotating the ankles, hips, or the whole upper body. On long or strong perturbations, a human should take a reactive stepping, i.e., take one or more steps to prevent falling. Falling is a behavior taken when a human loses her balance. However, this is not a completely passive motion because during the fall a human takes action to protect her body or other valuable objects. Other interesting interactive motions such as those made through the hands (e.g., handshaking) are out of the scope of this chapter.

In this chapter, we review some important approaches for human balancing, namely the ZMP-based methods, data-driven methods, and momentum-based methods. In the ZMP-based methods, which have been used extensively in controlling humanoid robots, humanoids are controlled to move in such a way that the ZMP, an important feature point related with balance, is located in the desired position during standing and walking.

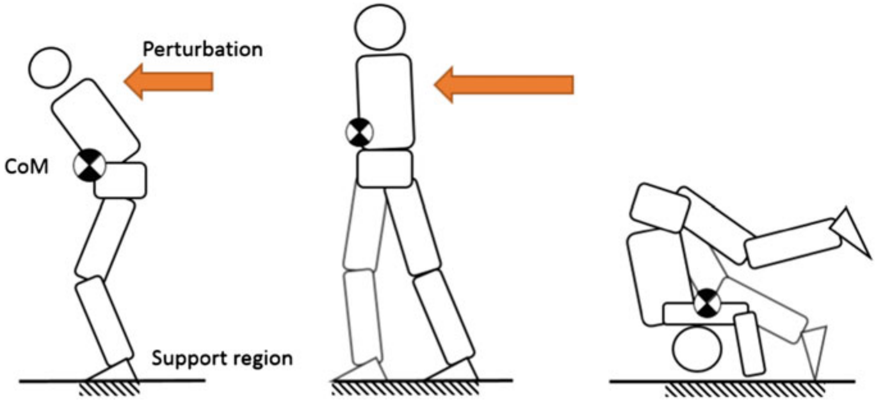


Fig. 8.1 Balance-related behaviors: *left* postural balancing, *middle* reactive stepping, and *right* falling

When only a physics-based controller (e.g., regular ZMP-based method) is used, the resulting motions typically look rather robotic and not humanlike. This would be fine for humanoid robots, but would be problematic for virtual humans that are expected to mimic the real human behavior. To overcome this limitation, data-driven approaches incorporate human motion capture data into the generation of humanoids motion. Usually the motion capture data provides the naturalness prior, while the motions are still generated by physics-based controllers to keep the responsiveness against the unpredictable inputs from the users.

Finally, we introduce the momentum-based approaches that are characterized by controlling both the linear and angular momenta of the whole body. By controlling not only the linear momentum (as typical ZMP-based methods do) but also the angular momentum, this approach enlarges the possible range of actions (i.e., the angular motion as well as the linear motion) for tracking the ZMP.

8.2 ZMP-Based Approaches

Let us first discuss ZMP-based methods. The *zero-moment point* (ZMP) [7, 16] is defined as a *center of pressure* (CoP) of the *ground reaction force* (GRF) and is a key concept regarding balance in humanoids. While ZMP has been used extensively for walking controllers, it has also been widely employed in postural balance controller as well. In fact, walking controller and balance controller are in many cases tightly bound and inseparable. We discuss the ZMP with respect to the postural balance, a main topic of this chapter.

Let us see the definition and computation of the ZMP first. Later, we discuss some methods using ZMP. The pressure on the foot in contact with the ground shows complex distribution over the contact surface. To express the force relationship between the foot and the ground compactly, the ground pressure is integrated into GRF applied to a point around which the horizontal components of the total moment vanish (Fig. 8.2). This point is referred to as ZMP. For simplicity, let us assume the ground is flat and all contact points lie in the XY-plane of the reference frame located on the ground. Suppose $\sigma_z(x, y)$ is the Z-component of the ground pressure at a point $p = [x, y, 0]^T$ as shown in Fig. 8.3. The sum of all vertical components of GRFs is thus

$$f_z = \int_S \sigma_z(x, y) dS, \tag{8.1}$$

where S is the area of contact between the foot and the ground. The moment $\tau_n(p)$ of GRF about a point p due to $\sigma_z(x, y)$ can be calculated as

$$\tau_n(p) = [\tau_{nx}, \tau_{ny}, \tau_{nz}]^T \tag{8.2}$$

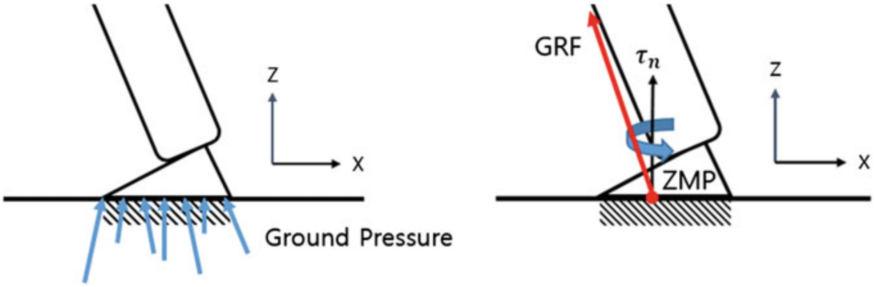
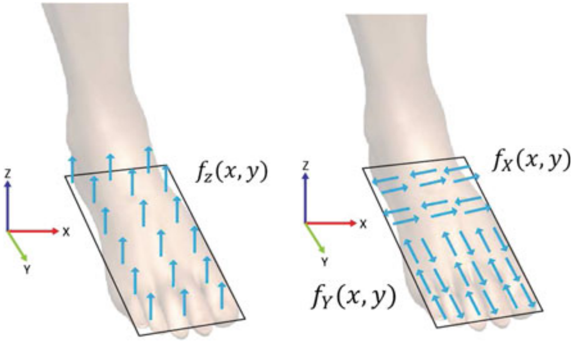


Fig. 8.2 Ground pressure on the base of the foot (*left*) can be equivalently expressed by the ground reaction force (*red arrow*) applied at the zero-moment point (ZMP) and a torque around the vertical axis

Fig. 8.3 *Left* The vertical components of the ground forces. *Right* The horizontal components of the ground forces



$$\tau_{nx} = \int_S (y - p_y) \times \sigma_z(x, y) dS \tag{8.3}$$

$$\tau_{ny} = -\int_S (x - p_x) \times \sigma_z(x, y) dS \tag{8.4}$$

$$\tau_{nz} = 0 \tag{8.5}$$

Since $\tau_{nx} = \tau_{ny} = 0$ at ZMP, Eqs. 8.3 and 8.4 find the position of the ZMP:

$$p_x = \frac{\int_S x\sigma_z(x, y)dS}{\int_S \sigma_z(x, y)dS} \tag{8.6}$$

$$p_y = \frac{\int_S y\sigma_z(x, y)dS}{\int_S \sigma_z(x, y)dS} \tag{8.7}$$

The ZMP $p = [p_x, p_y, 0]^T$ is also called the *Center of Pressure* (CoP), and we use both terms interchangeably in this chapter. Second, let us consider the effect of the horizontal component of the ground forces. Integrating them over the contact surface gives the tangential components of the GRF:

$$f_x = \int_S \sigma_x(x, y) dS \tag{8.8}$$

$$f_y = \int_S \sigma_y(x, y) dS \tag{8.9}$$

The moment $\tau_t(p) = [\tau_{tx}, \tau_{ty}, \tau_{tz}]^T$ due to σ_x and σ_y is

$$\tau_{tx}(p) = 0 \tag{8.10}$$

$$\tau_{ty}(p) = 0 \tag{8.11}$$

$$\tau_{tz}(p) = \int_S \Big\{(x - px)\sigma_y(x, y) - (y - p_y)\sigma_x(x, y)\Big\} dS \tag{8.12}$$

Note that $\tau_{tx}(p)$ and $\tau_{ty}(p)$ are zero because the horizontal components of the GRF cannot produce the moment about the horizontal axes. Nonzero $\tau_{tz}(p)$ means that the horizontal components of the GRF generate the normal component of the moment. Summing all components leads to the GRF and the moment about ZMP as

$$f = [f_x, f_y, f_z]^T \tag{8.13}$$

$$\tau(p) = \tau_n(p) + \tau_t(p) = [0, 0, \tau_{tz}]^T \tag{8.14}$$

Note that the ZMP is not a point where all components of the moment become zero: the vertical moment still exists.

ZMP is located inside the *support polygon*, a convex hull including all the contact points, and cannot get out of it. To see this, suppose the GRF consists of a finite number of forces $f_i := [f_{ix}, f_{iy}, f_{iz}]^T (i = 1 \ldots N)$ on contact points $p_i \in S$. Then the following relations hold:

$$f = \sum_{i=1}^{N} f_i \tag{8.15}$$

$$\tau(p) = \sum_{i=1}^{N} (p_i - p) \times f_i . \tag{8.16}$$

When p equals the ZMP, then $\tau_x(p) = \tau_y(p) = 0$, which leads to

$$p = \frac{\sum_{i=1}^{N} p_i f_{iz}}{\sum_{i=1}^{N} f_{iz}} \tag{8.17}$$

Rearranging this term, we get

$$p = \sum_{i=1}^{N} \alpha_i p_i, \qquad \text{where, } \alpha_i = \frac{f_{iz}}{f_z} \tag{8.18}$$

Since the GRF is unilateral (the foot cannot pull the ground), $f_{iz} \geq 0$. Thus,

$$\begin{cases} \alpha_i \geq 0 \qquad (i = 1 \ldots N) \\ \sum_{i=1}^{N} \alpha_i = 1 \end{cases} \tag{8.19}$$

Equation (8.19) shows that p must lie in the convex hull of the contact point of the support polygon:

$$p = \left\{ \sum_{i=1}^{N} \alpha_i p_i \mid p_i \in S \, (i = 1 \ldots N) \right\} \tag{8.20}$$

Figure 8.4 shows the ZMP for some cases of foot pressure distribution. When the pressure is evenly distributed over the contact surface, ZMP is located at the center of the foot indicating the posture is stably balanced and less likely to topple (Fig. 8.4, left). The quality of balance decreases as the pressure distribution is skewed, and a humanoid starts to topple when ZMP reaches an edge of the support polygon (Fig. 8.4, right).

Therefore, ZMP is regarded an important indicator for the balance of a humanoid, and researchers have put efforts in the development of efficient balance controllers using the ZMP. Most ZMP-based controllers are designed to move a subset of body parts (i.e., ankle or hip) or the whole body to make the ZMP located in the desired position, which is usually set at the center of the support polygon.

The simplest kind of ZMP-based methods would be the ankle control strategy [5, 17], which controls only the ankle torque to bring the ZMP toward the desired position inside the foot base; dorsiflexion of the ankle moves the ZMP backwards and plantar flexion moves it forward.

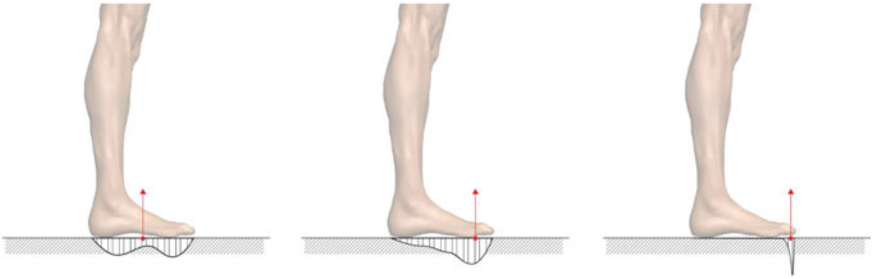


Fig. 8.4 Distributions of the pressure and the corresponding locations of ZMP

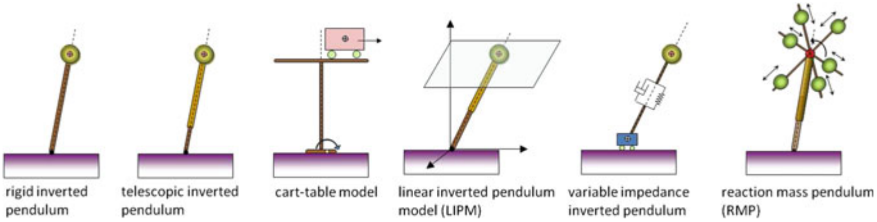

rigid inverted pendulum
telescopic inverted pendulum
cart-table model
linear inverted pendulum model (LIPM)
variable impedance inverted pendulum
reaction mass pendulum (RMP)

Fig. 8.5 Various reduced models for the dynamics of humanoid balance [14]

More advanced ZMP-based controllers operate to move the humanoid to follow the desired CoM position, which is appropriately set to achieve the desired ZMP position. For this, the relation between the CoM and the ZMP needs to be defined, and there are various models describing the core dynamics of the humanoid in terms of CoM and ZMP. Figure 8.5 shows some examples of such models. Note however that these reduced models only capture the important features of the humanoid dynamics, and there are always errors from the true dynamics of a humanoid caused by the simplification. For example, the 3D *linear inverted pendulum model* (LIPM) [9] relates the motion of CoM on a horizontal plane with respect to ZMP as follows:

$$\ddot{y} = \frac{g}{z_c}y, \quad \ddot{x} = \frac{g}{z_c}x, \tag{8.21}$$

where (x, y, z_c) is the position of CoM, and g denotes the acceleration of gravity. Therefore, the desired ZMP position is obtained by controlling the movement of the CoM. Different models provide different relations between the CoM and the ZMP.

Some methods [2, 6] take the strategies where the ground projection of the CoM moves to the support polygon. First, the desired acceleration of the CoM is determined from the desired CoM position.

$$\ddot{c}_d = k_s(c_d - c) - k_d\dot{c}, \tag{8.22}$$

where c_d is the desired CoM position. k_s and k_d are positional and damping gains, respectively, which control the influences of the positional and derivative terms on the desired acceleration. There are many choices on the desired CoM position for a stable configuration, but most opt for the center of the support polygon because in that case the humanoid can cope with perturbations from various directions.

Given the desired motion of the CoM, we need to determine the control input to achieve the goal. One method for this is the virtual force method [4, 6, 15]. The virtual force f_v is calculated from the desired acceleration of the CoM $\ddot{c}_d$

$$f_v = m\ddot{c}_d, \tag{8.23}$$

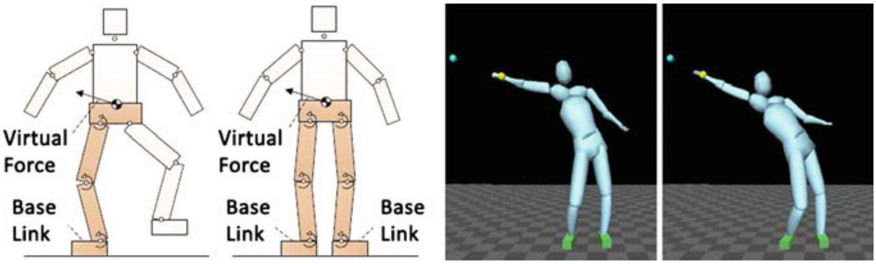


Fig. 8.6 *Left* Virtual forces applied to the CoM [4]. *Right* Reaching to the point while maintaining balance [2]

where m is mass of the humanoid. The virtual force is the imaginary force that would generate $\ddot{c}_d$. Then the virtual force is substituted by the joint torques τ from the relation:

$$\tau = J(p)^T f_v, \tag{8.24}$$

where $J(p)$ is the Jacobian matrix that maps the derivatives of the generalized coordinates to the velocity of the point p, the point on the foot that coincides with the ZMP. Equation (8.24) is a form of Jacobian transpose control that enables the operation space control by computing the corresponding joint torques (Fig. 8.6).

8.3 Data-Driven Approaches

Until now we have discussed ZMP-based approaches for maintaining balance of humanoids. They have shown good performance for maintaining balance, but the resulting motions may look rather robotic and do not closely look like human movements. This is a common downside of physics-based approaches that capture the physical aspect of human motions well but cannot account for other various factors related to human motion, such as aesthetic qualities and personal styles. These features are implicitly contained in motion capture data and thus the data-driven approaches that use recorded human motion data have great potential to generate natural looking motions. Thus, researchers have developed various methods that use recorded human motion data as a human motion prior to create humanoids motions.

Many data-driven approaches concern modifying recorded motion data to match different human figures and environments to increase reusability of the data through interpolating, blending, and rearranging motion clips. Some methods are effective for offline animation while others are good for interactive animation. Some pursue physical plausibility while others ignore it. Here we discuss only those data-driven approaches that generate physically plausible, interactive animation in real-time.

Naturally many methods of this kind employ physical controllers to generate motions or at least use physical principles to create movement kinematically.

A common framework of data-driven approaches for physical interaction is to generate reference motions using motion database and have a physical controller to track the reference motion. A basic tool for tracking is a PD controller, which is modeled as follows:

$$\tau = k_p(\theta_m - \theta_c) + k_d(\dot{\theta}_m - \dot{\theta}_c), \tag{8.25}$$

where θ_m and $\dot{\theta}_m$ denote the joint angles and velocities of the reference motion, and θ_c and $\dot{\theta}_c$ are the current joint angles and velocities of the humanoid. k_p and k_d are the proportional and derivative gains, respectively, which determine the magnitude of control force τ given the position and velocity differences.

The tracking controller is effective enough to follow simple motions such as upper body movement. Sometimes offline procedure for gain tuning achieves good tracking quality [4]. For more complex motions, simple tracking control may fail to achieve the goal. This is mainly because the human model (body dimensions and physical properties) and the environment (shape and contact configurations) at hand are different from those at the motion capture session. In this case motion data should also be modified. In order to create multi-contact motions such as walking and rolling, [11] developed a sampling-based optimization method, in which optimal reference motion is created through sampling so as to maximize the similarity to the original motion and the quality of balance. The latter is measured regardless of whether the CoM-ground projection is located inside the support polygon.

Nam et al. [13] divided the human model into dynamic and kinematic parts. Physical simulation is applied only to the dynamic parts, and the rest are moved kinematically. This approach eases the burden of physical controller while still being able to create responsive motions to user inputs interactively. To determine which part should be simulated, they compare each joint torque resisting the user disturbance with the predefined threshold value. If a joint torque exceeds the threshold, a chain of body parts from the joint to the terminal part that the user interacts with are treated as dynamic and a PD controller is applied to follow the reference motion. This approach provides rather simple means to creating responsive behaviors, but the resulting motions are generally limited to those not deviating much from the original motion.

Some methods create motions kinematically, but employ dynamic analysis or simulation as a tool to select suitable motion data to generate responsive motions against external perturbations. In general, motion data expresses human motions with high realism yet the range of expressible motion is limited by the size and content of the motion database. Therefore, motion data are usually interpolated and blended to make new motions. Yin et al. [19] (Fig. 8.7, left) used a set of motion data that captures responsive behaviors of humans under external push to create humanoid motions. By analyzing offline the momentum rate change of the motion data after push, the magnitude and direction of the external push are estimated and stored per each sample motion clip. Online, when a new external force is applied, the closest

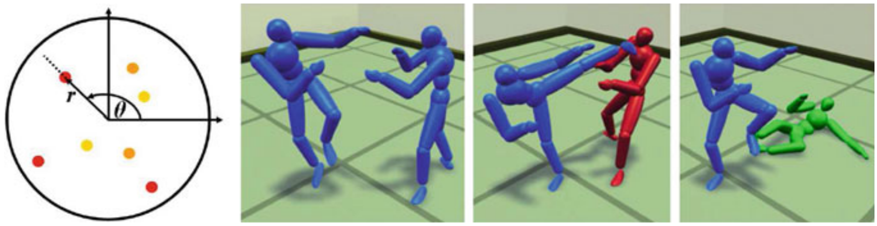


Fig. 8.7 *Left* Parameterization of push response. *Color* represents the type of balancing strategy and the vector represents the direction and the magnitude of the push [19]. *Right* An example of making falling motion due to kicking. Before being hit, the character moves by motion data. Falling motion (*red*) is created by a PD controller that creates smooth transition to a suitable lying motion (*green*) [21]

sample to the given external force is found and the corresponding sample motion is scaled to account for the difference (magnitude and direction) in the input force and the force in the closest sample. The final motion is generated kinematically rather than created by a physics-based controller.

Data-driven approaches have the advantages that they provide rather straightforward methods to create complex motions, albeit of limited scope, which are often difficult to obtain through physics-based approaches. Zordan et al. [21] proposed a method to create natural falling motion using motion data (Fig. 8.7, right). In this method, when a humanoid starts to fall given a high impact, a falling motion is simulated first to approximate plausible falling motion. Then a motion clip that is mostly like to occur after falling is searched from a database. Specifically, a motion clip that has the least difference from the simulated motion for a certain time window is selected. After finding this target motion clip that should occur after falling, the humanoid is re-simulated with a PD-controller activated to make a smooth transition to the target motion.

We have reviewed several data-driven approaches for creating responsive motions. Thanks to the motion capture data, the resulting motions of these approaches are highly natural. By employing physics-based techniques, data-driven approaches can achieve responsiveness to the external inputs. However, the output motions are generally limited to the scope of the recorded motion data.

8.4 Momentum Control Approaches

In Sect. 8.2, we have seen that the GRF and CoP (or ZMP) are important features regarding balance. In fact, GRF and CoP are closely related with the linear and angular momenta of a human. Newton–Euler equations of motion state that the total external force and torque equal the rate of change of the linear and angular momenta, respectively. Consider a standing human with total mass of m, who gets external forces only from the ground f and the gravitational force mg. If we compute a torque generated by the external forces about the CoM r_g of the human, the rate of change of linear momentum $\dot{l}$ and angular momentum $\dot{k}$ is as follows:

$$\dot{l} = mg + f \tag{8.26}$$

$$\dot{k} = (p - r_g) \times f + \tau_n, \tag{8.27}$$

where p denotes the location of CoP and τ_n is the torque about the vertical axis generated by the tangential forces from the ground. The position of the CoP can be calculated by substituting (8.26) into (8.27) and using the conditions $p_z = 0$ and $\tau_{nx} = \tau_{ny} = 0$.

$$p_x = r_{gx} - \frac{1}{\dot{l}_z - mg}(f_x r_{gz} - \dot{k}_y) \tag{8.28}$$

$$p_y = r_{gy} - \frac{1}{\dot{l}_z - mg}(f_y r_{gz} - \dot{k}_x) \tag{8.29}$$

From the equations, one can easily verify that GRF and CoP are uniquely determined by the linear and angular momentum rate change. In particular, GRF has a one-to-one relation with the linear momentum rate change, and CoP is determined by both linear and angular momentum rate changes.

Since a human body can be modeled as an articulated system that consists of a set of rigid bodies connected by joints, the momentum of a human is calculated as the total sum of momentum of individual rigid bodies, i.e.,

$$h_g(q, \dot{q}) = \sum_i^n X_i^T h_i, \tag{8.30}$$

where $h = (k^T, l^T)^T$ is the spatial momentum, a combination of angular and linear momentum, and X_i^T transforms h_i (spatial momentum of part i, expressed in its local frame) to the CoM frame (a reference frame located at CoM with its orientation aligned to the world frame). Note that the momentum is uniquely defined by the generalized coordinates q and their time derivatives $\dot{q}$, which means the GRF and CoP are completely controlled by the joint motions of the humanoid.

The idea of momentum control approaches is that by controlling both the angular and linear momentum of a human, one can control the GRF and CoP. Comparing with the ZMP-based control that usually controls only the CoM motion (hence the linear momentum), the momentum control approaches take angular momentum into account for designing the controller; hence they have higher potential to control the position of the CoP more precisely.

For postural balance control problem, [8] introduced angular momentum control for the whole body control problem. Abdallah and Goswami [1] developed a postural balance controller that controls the rate of change of linear and angular momenta of

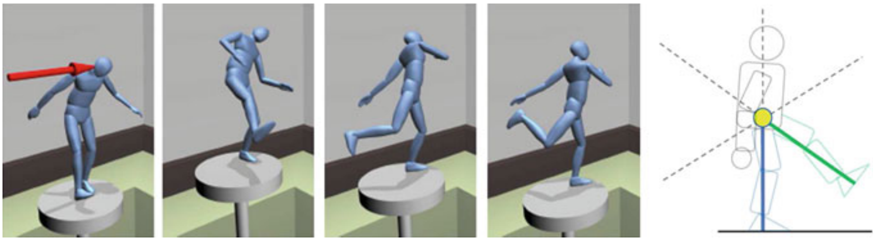


Fig. 8.8 *Left* The character responds to a disturbance [12]. *Right* A planar rimless wheel model (GFPE) [20]

a humanoid robot. More recently, [12] proposed a whole body postural balance controller that identifies the desired CoP as the high level input. (Figure 8.8, left) while [10] extended their method to present a balance controller for a nonlevel and nonstationary ground.

Recently, researchers have applied momentum control approaches to reactive stepping. Wu and Zordan [18] introduced a momentum-based stepping controller that generates parameterized curves for the swing foot and center of mass (CoM) trajectories according to the step position and duration. The method subsequently creates whole body motions to realize the trajectories via joint accelerations optimally calculated from the multiobjective function and joint torques computed by inverse dynamics. Yun and Goswami [20] presented a novel reactive stepping method based on a rimless wheel model. Specifically, they developed the *generalized foot placement estimator* (GFPE) to define the target stepping point. The proposed method is applicable to nonlevel ground (Fig. 8.8, right).

Here, we review a case of momentum control approach proposed in [10] for postural balance. In the work, the behavior of the momentum controller is determined by the set of inputs, the desired linear and angular momentum rate change ($\dot{l}_d$ and $\dot{k}_d$), as well as additional inputs such as the desired joint accelerations for the upper body (θ_d^u), and the desired position and velocity of the swing foot (T_d, v_d). Output is the appropriate joint torques for postural balance. Figure 8.9 shows the overall framework.

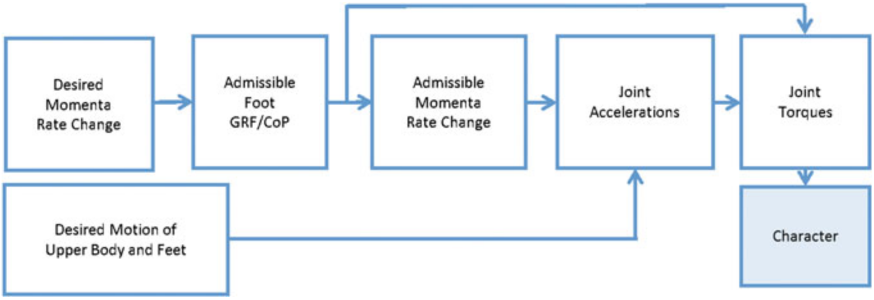

Desired Momenta Rate Change
Admissible Foot GRF/CoP
Admissible Momenta Rate Change
Joint Accelerations
Joint Torques
Desired Motion of Upper Body and Feet
Character

Fig. 8.9 Momentum control module [10]

The desired rate of change of the momenta is determined as

$$\dot{l}_d = \Gamma_{lp} m(r_{Gd} - r_G) + \Gamma_{ld} m(l_d - l) \tag{8.31}$$

$$\dot{k}_d = \Gamma_{kd}(k_d - k), \tag{8.32}$$

where Γ_{lp} and Γ_{ld} are proportional and derivative gains, respectively. Subscript $_d$ denotes the desired value of each property. Postural balance can be maintained if we set $l_d = 0$ and $k_d = 0$ to stabilize the momentum generated by the perturbation. r_{Gd} can be set such that the ground projection of CoM is located at the center of the support polygon.

Given the desired linear and angular momentum rate changes (which are not necessarily physically realizable by a humanoid), the admissible (physically realizable) GRF and CoP are determined such that they can create admissible momentum rate change that is as close as possible to its desired value. If both the desired linear and angular momentum rate changes are not admissible, one should choose either linear or angular momentum to satisfy and sacrifice the other. Here we will consider an easy case where both the desired momenta are admissible. This means that the admissible GRF and CoP can be calculated from (8.26) and (8.28)–(8.29) by substituting $\dot{l}$ and $\dot{k}$ with the desired values computed in (8.31) and (8.32).

After determining the admissible GRF, CoP, and momentum rate changes, the joint accelerations are determined to satisfy the desired values. As there are infinitely many solutions for the joint accelerations given the admissible momentum rate change, we can impose the preferred motion for the upper body and the swing foot as the secondary objective, so that the resulting joint accelerations will satisfy the given momentum rate change while generating the desired motion as close as possible. This is achieved by solving the following optimization problem:

$$\begin{gathered} \ddot{\theta}_a = \text{argmin}_{\ddot{\theta}}\, w_b \|\dot{h}_a - A\ddot{q} - \dot{A}\dot{q}\| + (1 - w_b)\|\ddot{\theta}^u_d - \ddot{\theta}^u\| \\ s.t. \quad J\ddot{q} + \dot{J}\dot{q} = a_d \quad \text{and} \quad \ddot{\theta}_l \leq \ddot{\theta} \leq \ddot{\theta}_u, \end{gathered} \tag{8.33}$$

where the first term on the RHS concerns satisfying the admissible momentum rate change $\dot{h}_a$, which is related to the second derivative of the generalized coordinates from the differentiation of the momentum–velocity relation:

$$h(q, \dot{q}) = A(q)\dot{q}, \tag{8.34}$$

where $A(q)$ maps the derivative of the generalized coordinates to the momentum. Refer to [14] for calculation of the matrix. Note that the generalized coordinates include joint angles, i.e., $\theta \subset q$. $\ddot{\theta}^u_d$ denotes the desired joint accelerations for the upper body. The first constraint term enforces the support foot to be stationary. The Jacobian matrix J maps the derivative of generalized coordinates to the velocity of the foot. The inequality constraints bound the joint accelerations to a feasible region.

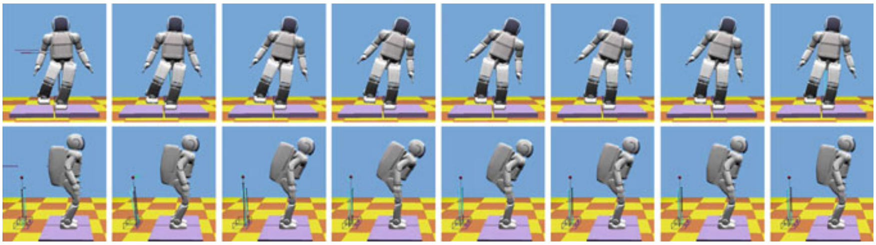


Fig. 8.10 Postural balancing using a momentum-control approach [10]

Given the admissible joint angles, one can use a PD-controller (8.22) to compute the necessary joint torques. A more advanced method is the inverse dynamics [3] that considers the physical properties of the humanoid model to compute the necessary joint torques. Figure 8.10 shows the balancing behavior of a standing humanoid when it is affected by external perturbation from the side and the back.

8.5 Conclusion

In this chapter, we discussed state-of-the-art technologies for generating responsive motions of humanoids in three categories: ZMP-based, data-driven, and momentum-based techniques.

The ZMP-based approach targets to control the position of ZMP by moving a subset of body parts or the whole body. To confront the complex humanoid dynamics, simplified reduced models that capture only the core dynamics of humanoids are used to determine suitable control inputs. Hence, ZMP-based methods are relatively simple and good for real-time applications that have a limited computational budget.

By referring to the motion capture data of real humans, the data-driven approach has great potential to create natural looking motions, which is particularly important for the visual quality of the virtual human characters. By combining with physics-based approaches, one can develop interactive humanoid animation. However, many factors related to the styles of human motions, such as gender, body dimensions, environment conditions, and personal styles, are all implicitly melted into the motion data, so it is challenging to extract such factors and control them at user's will. In general, the resulting motion is heavily dependent on the motion database. Generating diverse motions from a limited size of motion database remains an important future work.

Finally, the momentum-based approach attempts to control both the linear and angular momentum of the humanoids, thus it can control the movement of CoM and CoP more precisely. Postural controller and stepping controllers have been developed in this approach where it is rather straightforward to determine the desired linear and angular momenta. However, it remains an unsolved question as to how the desired momenta should be set for a wider range of human motions, e.g., running; an important topic for future work.

References

1. Abdallah M, Goswami A (2005) A biomechanically motivated two-phase strategy for biped upright balance control. In: Proceedings of the 2005 IEEE international conference on robotics and automation, ICRA '05, pp 1996–2001
2. Abe Y, da Silva M, Popović J (2007) Multiobjective control with frictional contacts. In: Proceedings of the ACM SIGGRAPH Eurographics symposium on computer animation 2007, Eurographics Association, pp 249-258
3. Featherstone R (1987) Robot dynamics algorithm. Kluwer Academic Publishers, Boston
4. Geijtenbeek T, Pronost N, van der Stappen F (2012) Simple data-driven control for simulated bipeds. In: Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation 2012, Eurographics Association, pp 211–219
5. Golliday CL Jr, Hemami H (1976) Postural stability of the two-degree-of-freedom biped by general linear feedback. IEEE Trans Autom Control 21(1):74–79
6. Ha S, Ye Y, Liu CK (2012) Falling and landing motion control for character animation. ACM Trans Graph 31(6):155:1–155:9
7. Kajita S, Harada K, Hirukawa H, Yokoi K (2014) Introduction to humanoid robotics. Springer, New York
8. Kajita S, Kanehiro F, Kaneko K, Fujiwara K, Harada K, Yokoi K, Hirukawa H (2003) Resolved momentum control: humanoid motion planning based on the linear and angular momentum. In: Proceedings of 2003 IEEE/RSJ international conference on intelligent robots and systems, IROS '03, pp 1644–1650
9. Kajita S, Kanehiro F, Kaneko K, Yokoi K, Hirukawa H (2001) The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In: Proceedings of 2001 IEEE/RSJ international conference of intelligent robots and systems, IROS '01, pp 239–246
10. Lee S-H, Goswami A (2012) A momentum-based balance controller for humanoid robots on non-level and non-stationary ground. Auton Robots 33(4):399–414
11. Liu L, Yin K, van de Panne M, Shao T, Xu W (2010) Sampling-based contact-rich motion control. ACM Trans Graph 29(4):128:1–128:10
12. Macchietto A, Zordan VB, Shelton CR (2009) Momentum control for balance. ACM Trans Graph 28(3):80:1–80:8
13. Nam HN, Arista CR, Karen L, Victor BZ (2012) Adaptive dynamics with hybrid response. In: SIGGRAPH Asia technical briefs 5(1–5):4
14. Orin DE, Goswami A, Lee S-H (2013) Centroidal dynamics of a humanoid robot. Auton Robots 35(2–3):161–176
15. Pratt J, Chew C-M, Torres A, Dilworth P, Pratt G (2001) Virtual model control: an intuitive approach for bipedal locomotion. Int J Robot Res 20(2):129–143
16. Vukobratovic M (1990) Biped locomotion. Springer, New York
17. Vukobratovic M, Frank AA, Juricic D (1970) On the stability of biped locomotion. IEEE Trans Biomed Eng 17(1):25–36
18. Wu C-C, Zordan VB (2010) Goal-directed stepping with momentum control. In: Proceedings of the 2010 ACM SIGGRAPH/Eurographics symposium on computer animation, Eurographics Association, pp 113–118
19. Yin K, Pai DK, van de Panne M (2005) Data-driven interactive balancing behaviors. In: Proceedings of the 13th Pacific conference on computer graphics and applications, Pacific Graphics '05, pp 118–121
20. Yun S, Goswami A (2011) Momentum-based reactive stepping controller on level and non-level ground for humanoid robot push recovery. In: Proceedings of 2011 IEEE/RSJ international conference on intelligent robots and systems, IROS '11, pp 3943–3950
21. Zordan VB, Majkowska A, Chiu B, Fast M (2005) Dynamic response for motion capture animation. ACM Trans Graph, pp 697–701