Multiple Balance Strategies for Humanoid Standing Control

XING Deng-Peng[1] LIU Xu[1]

Abstract Full-state feedback parametric controllers are proposed for standing balance of humanoid robots in response to impulsive and constant pushes. Multiple robot models are used to approach multiple strategies in human standing balance. For each model, we design a parametric controller acting on each state variable and optimize controller parameters for different push sizes, directions, and locations. The performance of each controller is shown in response to different external pushes. By comparing the capabilities of handling disturbances in each strategy, the contributions of every joint to standing balance are also explored.

Key words Parametric controller, optimization, multiple balance strategy, humanoid robot, impulsive and constant push

DOI 10.3724/SP.J.1004.2011.00228

Research in balancing a robot is fundamental for the stability of humanoid robot and is also valuable to the robust motion planning in complex environment. Such investigations also facilitate to give a deep understanding of how human reacts to disturbances and pave the way to applying such ideas in helping elders and disables.

Four strategies are revealed of human to compensate for perturbations[1]: an ankle strategy, behaving like an inverted pendulum for small perturbations; a hip strategy, employing ankle and hip actuators when the disturbance increases; a squat strategy, to flex knees and hips to lower the center of mass (CoM); and a step strategy, if the above strategies are not sufficient to keep balance. To imitate these behaviors, Atkeson et al.[2] use multiple strategies for standing balance arising from the same optimization criterion. Usually the control policies are discussed treating arms as a part of torso, neglecting the balance roles of upper limbs. In fact, arm swinging is important in reacting to sagittal disturbances and is also a common behavior for human perturbed standing balance. This paper focuses on the ankle, hip, squat, and arm swinging strategies without stepping, and uses them to approach human behaviors.

Many policies have been proposed in perturbed standing balance. Integral balance controller is proposed by using linear inverted pendulum model (LIPM) to allow humanoid to recover from large disturbances[3]. A biomechanically motivated two-phase strategy, reflex and recovery phases, is presented for robust balance maintenance under disturbance[4]. An approach using random sampling state in dynamic programming is investigated in several simulated robotics problems[5]. By using differential dynamic programming (DDP), a trajectory library of optimized control policy is generated for standing balance on adaptive grid of initial conditions[6]. Kuo uses optimal control and state estimation to explain selection of control strategies used by humans, in response to small perturbations to stable upright balance[7]. A linearized model and linear quadratic regulators (LQR) are used for optimization. Park et al.[8] test human postural responses in terms of a feedback control system with feedback gains, which are gradually scaled with perturbation size and can accommodate biomechanical constraints.

Compared with other methods, parametric controllers can produce feedback responses with a small number of parameters, with little need for storage of possible postures and pushes; with less computational cost, these controllers can be applied to complex models and problems. Standing balance involves geometry constraints and a set of gains might not handle all possible pushes. This paper is engaged in optimizing the postural feedback parameters to external perturbations with a variety of sizes, locations, and directions.

Most research focuses on instantaneous perturbations and designs various controllers in response to the impulses and recovering to the upright state. While, in reality, some perturbations may act for a period, during which the push size, direction, and location may also change. Often policies for impulses cannot handle constant pushes, so the controllers for both instantaneous and constant disturbances are needed. This paper uses multiple models to approach multiple strategies in standing balance, designs feedback controllers which act on each state error for impulsive and constant pushes, respectively, uses an optimization approach to access the optimal parametric controllers of each model for each push sizes, locations, and directions, and compares the performance of each strategy to explore the contributions of each joint in standing balance.

1 The one-link model

With small external pushes, human usually takes the ankle strategy for balance, fixing other joints and actuating the ankles. This study uses a one-link inverted pendulum model, as shown in Fig. 1, with an actuator at the ankle joint to approach the ankle strategy. The model is facing to the right, with 1.314 m height and 48.97 kg weight. We assume that when the center of pressure is at the center of the foot. The joint limit is $-0.52 < \theta_a < 0.79$ rad, with $\theta_a = 0$ standing for the upright state, and the torque is bounded by $\pm 50\,\mathrm{N{\cdot}m}$ to prevent the foot from tilting. Robot parameters in this paper are taken from a preliminary design of a planned humanoid. We use this example to describe our optimization approach.

1.1 Balance controller

Define the state as ankle angle and angular velocity, and the robot dynamics are

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \tau, F, r) \tag{1}$$

where $\boldsymbol{x} = (\theta_a, \dot{\theta}_a)^{\mathrm{T}}$ is the state vector, $\tau = \tau_a$ represents the ankle torque, and F and r are the push size and location. We assume no slipping or other change of contact state during the perturbation.

Consider a feedback controller which acts on the error in each state variable

$$\tau_a = -k_1 \Delta\theta_a - k_2 \Delta\dot{\theta}_a \tag{2}$$

Manuscript received June 7, 2010; accepted June 8, 2010

1. Shanghai Chest Hospital Affiliated to Shanghai Jiao Tong University, Shanghai 200030, P. R. China

where $\Delta\theta_a = \theta_a - \theta_{a\_d}$ and $\Delta\dot{\theta}_a = \dot{\theta}_a - \dot{\theta}_{a\_d}$ represent the error between actual state and the desired state for the current push. The torque output from the controller is limited as described earlier.

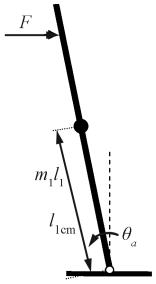


Fig. 1 The one-link robot model

1.2 Optimization approach

The one step optimization cost function is defined as a weighted sum of the squared state error and the joint torque magnitude

$$L = T(\Delta\theta_a^2 + \Delta\dot{\theta}_a^2) + 0.02\,T\tau_a^2 \tag{3}$$

where T is the time step of the simulation (0.01 s) and 0.02 weights the torque penalty relative to the state error. The total cost is the sum of the one step cost function over time. We use this cost function as an optimization criterion to find the minimum total cost for each push, optimizing the parameters of the feedback controller.

The controller parameters are optimized for both impulsive and constant pushes, respectively, with a number of sizes and locations. All pushes are assumed horizontal, as vertical pushes have little effect. An impulsive push exists for a short period of time (0.1 s) and upright state is the desired state; with the given push location, we choose as simulation perturbations a set of push sizes which are between zero and the given magnitude in optimization, rather than optimizing from one push size. For a constant push, based on each push size and location, we first calculate an equilibrium state, which is the posture where the robot leans into the push and the torque at the ankle is zero. We use this equilibrium state as the desired state, rather than the posture of standing straight up. Instead of optimizing from one initial state of standing vertically, we choose a set of initial states which lie in the range between the state of straight up and the desired state. For this strategy, the equilibrium state is

$$\theta_{a\_d} = \arctan\frac{Frl_1}{m_1 g l_{1cm}} \tag{4}$$

SNOPT[9] is a general purpose system for constrained optimization, using sequential quadratic programming (SQP). We employ it to optimize controller parameters for each push, using the LQR gain for standing upright as initial values for our controller and optimizing the next push size with the optimized gains of the previous case. For an impulsive push, with the given location, we choose five pushes as the simulation perturbations including the given magnitude. For a constant push, with the given size and location, we choose five initial states for each joint including the posture where the robot stands straight. The cost of the trajectory from each push size (or initial state) is combined, with a penalty added for cases violating state constraints. Optimized parameters can be generated for a wide range of pushes. The robot looks up the desired state and feedback gains based on the push force and location.

1.3 Control scheme

Fig. 2 shows the diagram of balance control in robot to respond to external disturbances, by looking into the push table and finding the best entry. Online push estimation continuously observes the robot state to estimate the current push information. Since the robot dynamics model is nonlinear, an extended Kalman filter[10] is employed to use the current state and torque to estimate the push size and location. The desired state and feedback gains are determined by looking into the push table and finding the appropriate entry according to the estimated push information. For the impulsive pushes, the controller chosen for the last detected push is applied for a period of time. With the state error and feedback gains, joint torques are then generated by the controller and limited.

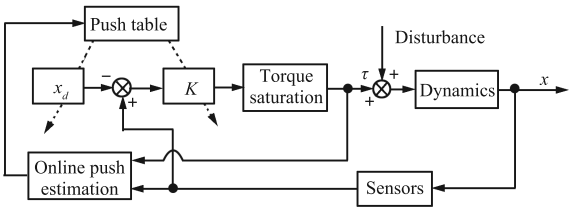
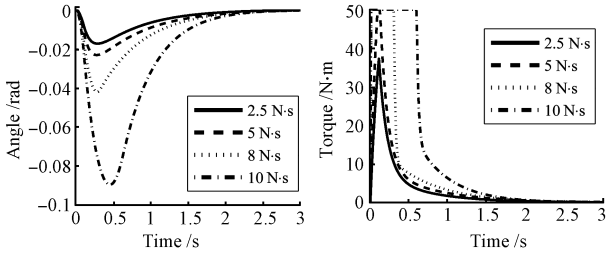


Fig. 2 Diagram of feedback balance control

1.4 Results

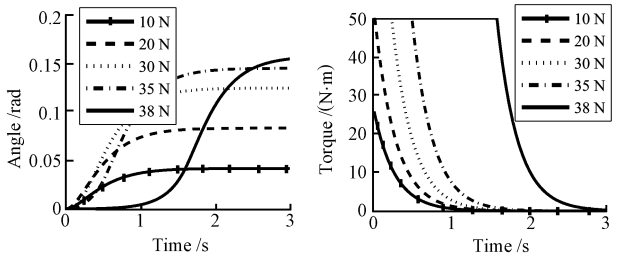
Fig. 3 shows the angle trajectories and the corresponding torques of the proposed controller for a range of impulsive push sizes located at the head. Angles and torques roughly scale with push magnitude, except for the torque saturation to larger perturbations. The joints initially grow in negative direction and approach to zero after reaching the maximum negative displacements. The torques initially saturate at 50 N·m if that torque is reached, and then smoothly decrease to zero. The maximum handling ability of the controller to impulsive perturbations is 10 N·s, and for higher impulses the robot will fall down.



(a) Ankle angle (b) Ankle torque

Fig. 3 Angle trajectories for a range of impulsive perturbation sizes at the head (2.5, 5, 8, 10 N·s)

Fig. 4 shows the angle trajectories and the corresponding torques for a set of constant push sizes at the head. With the small push sizes, angles gradually scale with push magnitude, quickly reaching to the corresponding equilibrium states. With the large push sizes, the robot initially takes small displacements and gradually approaches to the desired states. As the push grows larger, the initial ankle displacement becomes smaller and the torque saturation persists longer. For the constant push of 38 N and higher at the head, the controller falls down.



(a) Ankle angle (b) Ankle torque

Fig. 4 Angle trajectories for a range of constant perturbation sizes at the head (10, 20, 30, 35, 38 N)

2 The two-link model

For bigger pushes, simple ankle strategy is not suitable for balance and human gets support from hip joint, taking advantage of large hip flexing. The two-link model simulates this hip strategy, as shown in Fig. 5. The upper link is the torso, with 0.661 m height and 29.5 kg weight, and the lower link is the leg, with 0.653 m height and 19.5 kg weight. The hip angle limit is $-2.18 < \theta_h < 0.52$ rad and its torque is bounded by ± 157 N·m. The angle limit and torque boundary of the ankle is the same as in the previous model.

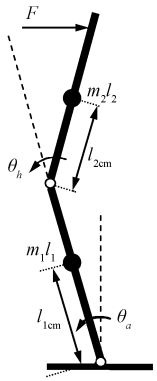


Fig. 5 The two-link robot model

2.1 Controller and optimization

Define the state as ankle and hip angles and angular velocities, and the feedback controller acts on state errors

$$\begin{aligned}\tau_a &= -k_1\Delta\theta_a - k_2\Delta\theta_h - k_3\Delta\dot{\theta}_a - k_4\Delta\dot{\theta}_h \\ \tau_h &= -k_5\Delta\theta_a - k_6\Delta\theta_h - k_7\Delta\dot{\theta}_a - k_8\Delta\dot{\theta}_h\end{aligned} \tag{5}$$

where

$$\begin{aligned}\Delta\theta_a &= \theta_a - \theta_{a\_d}, \qquad \Delta\theta_h = \theta_h - \theta_{h\_d} \\ \Delta\dot{\theta}_a &= \dot{\theta}_a - \dot{\theta}_{a\_d}, \qquad \Delta\dot{\theta}_h = \dot{\theta}_h - \dot{\theta}_{h\_d}\end{aligned}$$

represent the error between ankle and hip actual state and the desired state for the current push.

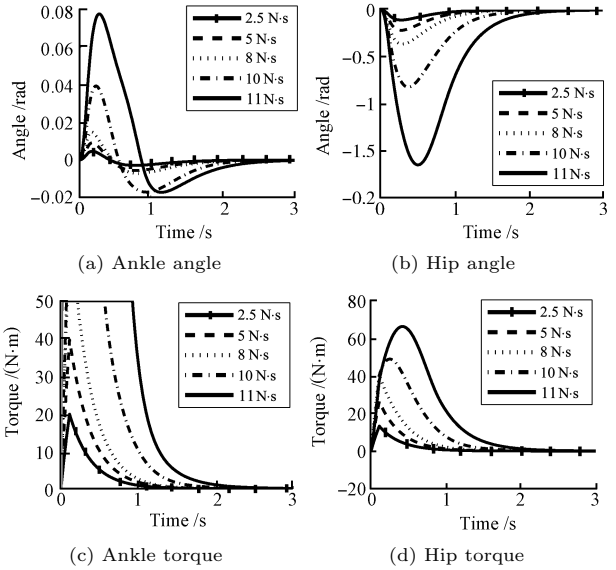
Define the one step optimization cost function of two-link model as

$$L = T(\Delta\theta_a^2 + \Delta\theta_h^2 + \Delta\dot{\theta}_a^2 + \Delta\dot{\theta}_h^2) + 0.02\,T(\tau_a^2 + \tau_h^2) \tag{6}$$

where T is the time step of the simulation (0.01 s) and 0.02 weights the torque penalty relative to the state error. We use the same optimization approach to acquire the controller parameters, according to different push sizes and locations, as described in the above strategy.

2.2 Results

Fig. 6 shows the angle trajectories and the corresponding torques of the proposed controller for a range of impulsive push sizes located at the head. Angles and torques are also roughly scaled with push magnitude, except ankle torques saturate for larger perturbations. The hip angle is entirely negative, reaching the desired state after reaching the maximum displacement, and the hip torques smoothly decrease from positive to zero for the small perturbations, whereas initially grow and then decline to zero for the large perturbations. Different from the one-link model behaviors, the ankle joint of hip strategy initially grows in positive direction, changes to negative shortly after the hip reaches its maximum, and returns back to zero. With the help of large hip flexing, hip strategy extends the impulsive handling to 11 N·s, where hip joint is close to its negative limit.



(a) Ankle angle (b) Hip angle

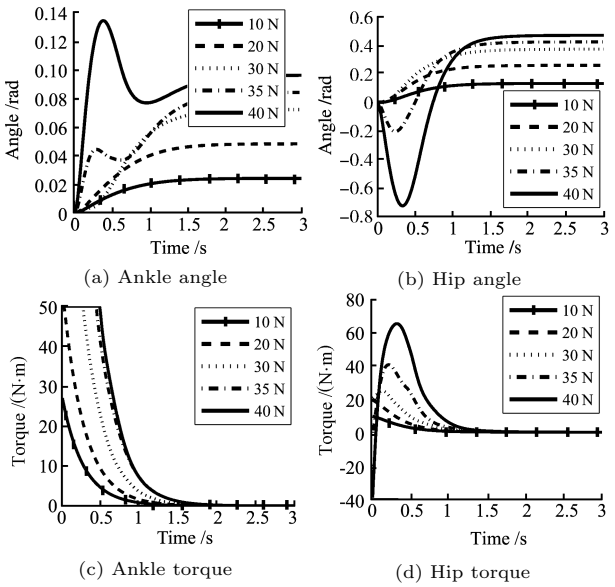
(c) Ankle torque (d) Hip torque

Fig. 6 Angle and torque trajectories for a range of impulsive perturbation sizes at the head (2.5, 5, 8, 10, 11 N·s)

Fig. 7 shows the angle trajectories and the corresponding torques for a set of constant push sizes at the head. With the small push sizes, ankle and hip angles gradually scale with push magnitude, reaching to the corresponding equilibrium states; ankle and hip torques smoothly decrease to zero, with ankle saturation. With the large push sizes, hip angle responds negatively at first and moves to the positive desired posture; compared with the one-link trajectories, the ankle reacts initially to a big displacement and moves smoothly to the corresponding desired state; the ankle torque saturation persists longer. The maximum displacements of the hip negative angle and the corresponding ankle angle are proportional to the push size. For perturbations of 47 N and higher at the head, the robot can also keep balance, but the equilibrium state is out of the joint limit.

We have designed an LQR controller that optimizes the same criterion on the two link model. For small perturbations, the performance of LQR controller is almost the same as the proposed controller; for large push sizes, the behaviors exhibit much more differences. Fig. 8 shows the trajectory comparison between the LQR and the proposed controllers with a range of pushes at the head (the dashed lines stand for the trajectories of the LQR controller, and the solid lines for the proposed controller). The horizontal

and vertical axes are ankle and hip angles, respectively. In response to perturbations, the robot starts from the initial state x_0, the straight up posture, progressing to the desired states over each time step. Starting from 20 N push, the trajectories of controllers begin to show differences, which become more significant as the push size increases. The trajectories of the LQR controller have the same trend as before, which starts with negative ankle angles and positive hip angles and moves to the equilibrium state after the ankle reaches its maximum displacement. For 35 N, the largest hip angle is closed to its state constraint. The trajectories of the proposed controller present more changes: for smaller pushes, it has a similar trend as the LQR controller while exhibiting less joint displacements; for bigger pushes, the robot moves in an opposite direction, taking advantage of large hip flexing. For the constant pushes at the head, the LQR controller can handle [−35 N, 35 N] and falls down for higher perturbations, whereas our controller can extend to [−40 N, 46 N] and even can keep the robot stable for higher perturbations although the equilibrium states are outside the joint limits.



(a) Ankle angle (b) Hip angle

(c) Ankle torque (d) Hip torque

Fig. 7 Angle and torque trajectories for a range of constant perturbation sizes at the head (10, 20, 30, 35, 40 N)

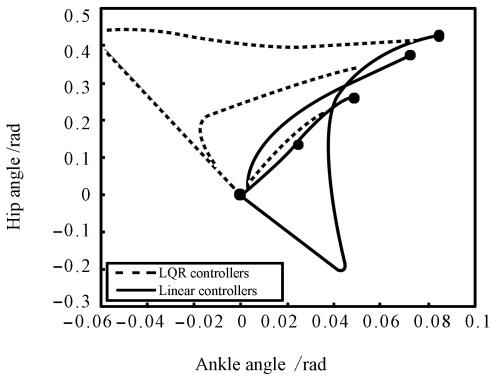


Fig. 8 Joint angle trajectories of the LQR and the proposed controllers for a series of constant pushes at the head (10, 20, 30, 35 N)

3 The three-link model

With increasing pushes, human employs the squat strategy to lower the CoM, by actuating the knee joints. The three-link model is used here to approach this squat strategy, as shown in Fig. 9. The leg is divided into the thigh, 0.33 m high and 13.7 kg weight, and the shin, 0.33 meters high and 5.78 kg weight. The knee joint is limited as $-0.01 < \theta_k < 1.87$ rad and the torque bound is ± 245 N·m. The small negative knee angle limit is used to expand the optimization region. The angle limits and torque boundaries of other joints are the same as described before.

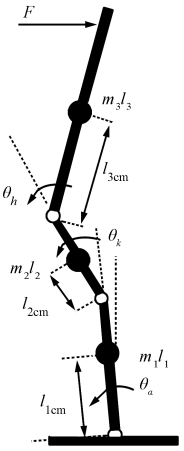


Fig. 9 The three-link robot model

Define the state as ankle, knee, and hip angles and angular velocities, and a feedback controller is considered as acting on the each state error:

$$\begin{aligned} \tau_a &= -k_1 \Delta\theta_a - k_2 \Delta\theta_k - \cdots \\ \tau_k &= -k_7 \Delta\theta_a - k_8 \Delta\theta_k - \cdots \\ \tau_h &= -k_{13} \Delta\theta_a - k_{14} \Delta\theta_k - \cdots \end{aligned} \tag{7}$$

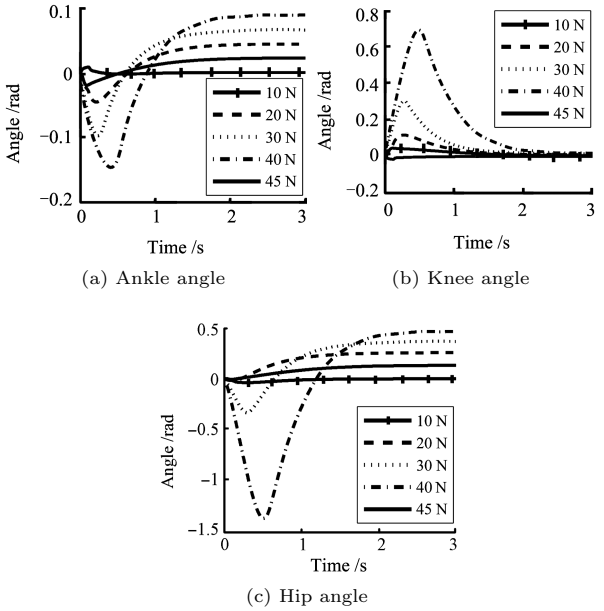
where $\Delta\theta_i = \theta_i - \theta_{i\_d}$ and $\Delta\dot{\theta}_i = \dot{\theta}_i - \dot{\theta}_{i\_d}$ represent the error between ankle, knee, and hip actual state and the desired state for the current push. The torque outputs from the controller are limited.

The one step optimization cost function of the three-link model is defined as

$$\begin{aligned} L = \; & T(\Delta\theta_a^2 + \Delta\theta_k^2 + \Delta\theta_h^2 + \Delta\dot{\theta}_a^2 + \Delta\dot{\theta}_k^2 + \\ & \Delta\dot{\theta}_h^2) + 0.02\, T(\tau_a^2 + \tau_k^2 + \tau_h^2) \end{aligned}$$

where T is the time step of the simulation (0.01 s) and 0.02 weights the torque penalty relative to the state error. The same optimization approach in the ankle strategy is also applied here to obtain the parameters.

Fig. 10 shows the angle trajectories for a set of constant push sizes at the head. The angles are roughly scaled with the push magnitude. Knee angles firstly increase to positive values and then approach to the small equilibrium states; compared with the two-link model, hip angles have the same trajectory trend and ankle angles are different: decrease to negative displacements and move back to positive desired states. The maximum push handling of the squat strategy is 14 N·s for the impulse and 46 N for the constant push.



(a) Ankle angle (b) Knee angle

(c) Hip angle

Fig. 10 Angle trajectories for a range of constant perturbation sizes at the head (10, 20, 30, 40, 45 N)

4 The four-link model

Without taking a step, the arm swinging strategy helps people dealing with the biggest possible disturbances in the sagittal plane. It uses appropriate arm swinging motion with adequate squatting postures to balance external perturbations. We use a four-link model, including ankle, knee, hip, and shoulder joints shown in Fig. 11, to explore this more complex and human-like strategy. The arm is separated from the torso, with 0.67 m high and 8.54 kg weight; the torso changes to 20.96 kg weight and the location of CoM also changes. The leg parameters are the same as three-link model. The shoulder torque limit is $\pm 150\,\mathrm{N{\cdot}m}$.

Define the state as each angle and angular velocity, and a feedback controller is considered as acting on the each state error:

$$\begin{aligned}\tau_a &= -k_1\Delta\theta_a - k_2\Delta\theta_k - \cdots\\ \tau_k &= -k_9\Delta\theta_a - k_{10}\Delta\theta_k - \cdots\\ \tau_h &= -k_{17}\Delta\theta_a - k_{18}\Delta\theta_k - \cdots\\ \tau_s &= -k_{25}\Delta\theta_a - k_{26}\Delta\theta_k - \cdots\end{aligned} \tag{8}$$

where $\Delta\theta_i = \theta_i - \theta_{i\_d}$ and $\Delta\dot{\theta}_i = \dot{\theta}_i - \dot{\theta}_{i\_d}$ represent the error between ankle, knee, hip, and shoulder actual states and the desired states for the current push.

The one step optimization cost function is defined as a weighted sum of the squared state error and on the joint torque

$$\begin{aligned}L = T(&\Delta\theta_a^2 + \Delta\theta_k^2 + \Delta\theta_h^2 + \Delta\theta_s^2 + \Delta\dot{\theta}_a^2 + \Delta\dot{\theta}_k^2 +\\ &\Delta\dot{\theta}_h^2 + \Delta\dot{\theta}_s^2) + 0.02\,T(\tau_a^2 + \tau_k^2 + \tau_h^2 + \tau_s^2)\end{aligned} \tag{9}$$

where T is the time step of the simulation (0.01 s) and 0.02 weights the torque penalty relative to the state error. We employ the same optimization approach for both impulsive and constant pushes.

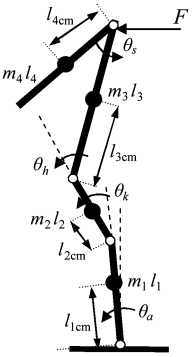
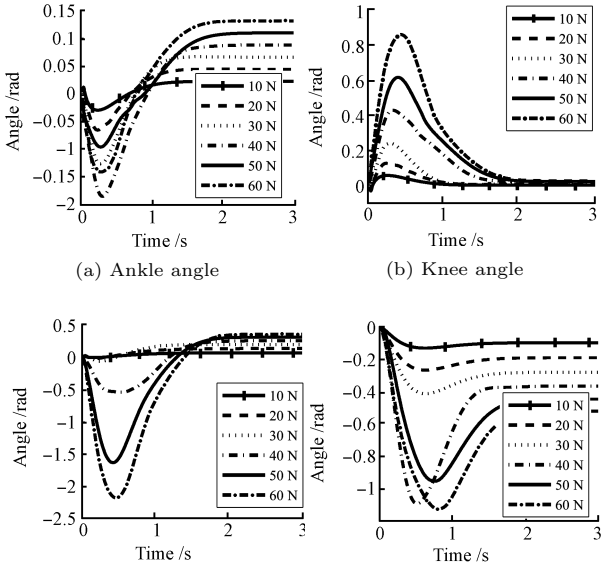


Fig. 11 The four-link robot model



(a) Ankle angle (b) Knee angle

(c) Hip angle (d) Shoulder angle

Fig. 12 Angle trajectories for a range of constant perturbation sizes (10, 20, 30, 40, 50, 60 N)

Fig. 12 shows the angle trajectories for a set of constant push sizes at the head. For small pushes, shoulder angles gradually direct to the negative desired states; for large pushes, the arm moves to large negative displacements and then approaches to the desired positions. The other joints have the same trends as the three-link model behaviors, except for different magnitudes. Compared with the squat strategy, the arm swinging policy can handle much larger push sizes.

5 Discussions

We have described the trajectory responses of each model and compared the joint behaviors among each strategy. We compare the maximum push handling of optimized parametric controllers for each strategy, with push location at the head, to explore the significance of each joint. Table 1 shows the possible impulsive and constant pushes to the right of each model. With one actuator at the ankle, the one-link model can handle up to the 38 N constant push and the 10 N·s impulse. For bigger pushes, the robot will fall down. Adding the hip joint, the hip strategy can extend the maximum push handling to the 12 N·s impulse and the 46 N constant push, by taking advantage of large hip flex-

ing. With contributions of the knee joint, the squat strategy has little effect on the constant push case but expands the impulse handling region to the 14 N·s, showing the influence of lowering the CoM. For larger constant pushes, the hip strategy and squat strategy can also keep robot stable, but the desired state is out of the joint limit. Due to the arm swinging compensation, the four-link model can handle up to the 19 N·s impulse and the 100 N constant push. With the arm separated from the torso and the vertical desired state of the shoulder joint for constant pushes, the four-link model can reach the desired state for large pushes without any feedforward torques.

Table 1 The maximum pushes of optimized controller for each model

Push types	One-link	Two-link	Three-link	Four-link
Impulsive	10 N·s	12 N·s	14 N·s	19 N·s
Constant	38 N	46 N	46 N	100 N

From Table 1 and the above analysis, every joint contributes in handling external disturbance. Hip joints change the trajectory of the CoM by varying the relative posture between the upper and lower bodies, and the large forward hip flexing improves the handling capability of the hip strategy. Knees lower the CoM and their big backward bending expands the handling of instantaneous disturbances. The addition of shoulder joints separates arms from the torso, and balances bigger perturbations by adequately swinging arms to generate reverse moments. For even bigger pushes which the arm swinging strategy cannot handle, the robot (human) will have to take a step. It coincides with the observation of human′s responses to disturbances: ankles apply torque to the ground; hips and arms generate horizonal ground forces; knees and hips squat[2]. The above analysis and the trajectory comparison between different models show that in standing balance arms have a significant effect, enhancing system stability and increasing stability margin, and should not be ignored simply as a mass balance.

6 Conclusions and future work

In this paper, full state feedback parametric controllers in the sagittal plane are designed for standing balance in response to both impulsive and constant pushes. Multiple models are used to approach multiple balance strategies. SNOPT is used to optimize the parametric controllers for different push sizes, locations, and directions. The performance of each strategy is shown with optimized parametric controllers, and we also illustrate the roles of ankle, knee, hip and shoulder joints in standing balance.

In future, this work will be extended to test the results on a robot platform, which needs to deal with imperfect sensor information and a compliant floor. To compare with human behaviors is another task; we will also explore the balance strategy with stepping as a possibility.

References

1 Runge C F, Shupert C L, Horak F B, Zajac F E. Ankle and hip postural strategies defined by joint torques. *Gait and Posture*, 1999, **10**(2): 161−170

2 Atkeson C G, Stephens B. Multiple balance strategies from one optimization criterion. In: Proceedings of the 7th IEEE-RAS International Conference on Humanoid Robots. Pittsburgh, USA: IEEE, 2007. 57−64

3 Stephens B. Integral control of humanoid balance. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robotics and Systems. San Diego, USA: IEEE, 2007. 4020−4027

4 Abdallah M, Goswami A. A biomechanically motivated two-phase strategy for biped upright balance control. In: Proceedings of the IEEE International Conference on Robotics and Automation. Washington D. C., USA: IEEE, 2005. 1996−2001

5 Atkeson C G, Stephens B J. Random sampling of states in dynamic programming. *IEEE Transactions on System, Man, Cybernetics, Part B: Cybernetics*, 2008, **38**(4): 924−929

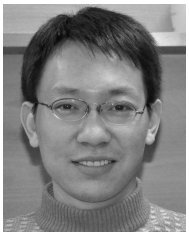
6 Liu C G, Atkeson C G. Standing balance control using a trajectory library. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. St. Louis, USA: IEEE, 2009. 3031−3036

7 Kuo A D. An optimal control model for analyzing human postural balance. *IEEE Transactions on Biomedical Engineering*, 1995, **42**(1): 87−101

8 Park S, Horak F B, Kuo A D. Postural feedback responses scale with biomechanical constraints in human standing. *Experimental Brain Research*, 2004, **154**(4): 417−427

9 Gill P E, Murray W, Saunders M A. SNOPT: an SQP algorithm for large-scale constrained optimization. *SIAM Journal on Optimization*, 2002, **12**(4): 979−1006

10 Brown R G, Hwang P Y C. *Introduction to Random Signals and Applied Kalman Filtering (Third Edition)*. New York: Wiley, 1997



XING Deng-Peng Postdoctor at Shanghai Chest Hospital Affiliated to Shanghai Jiao Tong University. He received his bachelor and master degrees from Tianjin University in 2002 and 2006, respectively, and Ph. D. degree from Shanghai Jiao Tong University in 2010. His research interest covers robotics, optimization, intelligent control, and medical simulator. E-mail: xingdengpeng@hotmail.com

LIU Xu Professor at Shanghai Chest Hospital Affiliated to Shanghai Jiao Tong University. His research interest covers atrial fibrillation and catheter ablation. Corresponding author of this paper.
E-mail: xkliuxu@126.com