Hopping with Nearly-Passive Flight Phases

Qinghong Guo, Chris Macnab
Department of Electrical and Computer Engineering
University of Calgary
Calgary, AB, Canada, T2N 1N4
{qguo,cmacnab}@ucalgary.ca

Jeff Pieper
Department of Mechanical and Manufacturing
University of Calgary
Calgary, AB, Canada, T2N 1N4
pieper@ucalgary.ca

***Abstract*—The paper presents a novel method which generates hopping gaits for an articulated single leg. Inspired by human running, the flight phase is assumed to be nearly-passive. Consequently, the initial joint velocities of the flight phase can be solved by using a static optimization procedure, provided that the boundary joint angles have been picked in advance. The two hopping phases can then be dynamically optimized, with the velocity jumps of the joints predicted by a simple collision model. The direct single shooting method solves the nonlinear constrained optimization problem. Ground reaction forces and the Zero-Moment-Point serve as constraints. Simulation results show that the proposed approach produces energy-efficient hopping gaits.**

***Index Terms*—One-legged hopping robot; nearly-passive flight phase; dynamical optimization; direct single shooting method.**

I. Introduction

Aiming for a stepping stone to bipedal motion in machines, many have investigated single-legged hopping robots with a knee and ankle. De Man *et al.* developed a kneed hopper, named OLIE, whose hip joint was also located at the CoM of the upper body [1]. Their group then examined a more general hopper model, mimicking human legs [2]. They proposed a simple gait generation algorithm using 5th-order polynomials. Ikeda *et al.* constructed an articulated hopper [3]. Their gait generation algorithm stems directly from sampled data of kangaroo hopping. Hyon *et al.* built a hopper with a knee and a passive ankle, also inspired by kangaroos [4]. An elastic tendon connects the thigh and the foot; thus the kinetic energy can be stored and released in different motion phases. To avoid unwanted oscillations of the springy leg, the control scheme has to be deliberately designed.

In this paper, we present an off-line gait generator for an articulated hopping model without elastic elements. Inspired by human running, we assume that the flight phase is nearly passive (Section III). Based on this fundamental assumption, the flight phase and the stance phase are treated as two-point boundary value problems (TPBVP). An energy-efficient hopping cycle, corresponding to the commanded step length and commanded average forward speed, can be dynamically optimized by using the direct single shooting method (Section II). We describe the Matlab implementation of the proposed approach (Section IV) and show simulation results (Section V).

II. The direct single shooting method

A dynamic optimization (only the Lagrange-type is considered in this paper) minimizes the cost function

$$\mathcal{C} = \int_{t_0}^{t_\mathrm{f}} \mathcal{L}(x(t), u(t), t)dt, \tag{1}$$

subject to

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), t), & (2)\\ x(t_0) &= x_0, & (3)\\ x(t_\mathrm{f}) &= x_\mathrm{f}, & (4)\\ \Psi_\mathrm{x}(x, \dot{x}) &\le 0, & (5)\\ \Psi_\mathrm{u}(u) &\le 0, & (6) \end{aligned}$$

where

- t is the time,
- $x = x(t) \in \Re^n$ is the state vector,
- $u = u(t) \in \Re^m$ is the control vector,
- the equality constraint (2) is the state space model of the dynamic system,
- the Lagrangian $\mathcal{L}(x, u, t)$ will be designed,
- t_0 and t_f are the initial and final time instants,
- x_0 and x_f are the known initial and final states (the two points in the TPBVP),
- $\Psi_\mathrm{x}(x, \dot{x})$ represents the set of all inequality constraints on x and $\dot{x}$,
- and $\Psi_\mathrm{u}(u)$ is the set of all inequality constraints on u.

In the sequel, the set of all equality constraints may be indicated by $\Phi(x, \dot{x})$.

Numerical techniques will produce the solution to the above nonlinear dynamic optimization problem. One can discretize the continuous-time control vector u into piecewise constant control signals on fixed time grid,

$$t_0 < t_1 < \cdots < t_{\mathrm{N}-1} < t_\mathrm{N} = t_\mathrm{f},$$

where N is the number of time intervals.

Let $U = [u(0)^\mathrm{T}, u(1)^\mathrm{T}, \cdots, u(N)^\mathrm{T}]^\mathrm{T}$ be the decision vector, and U_min and U_max be the bounds corresponding to $\Psi_\mathrm{u}(u)$. Plugging an estimate (or a shot) of U, denoted by $\hat{U}$ into (2) yields

$$\dot{x} = f(x, \hat{U}, t). \tag{7}$$

With x_0 known, integrating (7), one obtains a trial version of x, denoted by $\hat{x}$. Let $\hat{x}_\mathrm{f} = \hat{x}(t_\mathrm{f})$ and $e_\mathrm{x} = x_\mathrm{f} - \hat{x}_\mathrm{f}$. For

978–1–4244–1676–9/08/$25.00 © 2008 IEEE

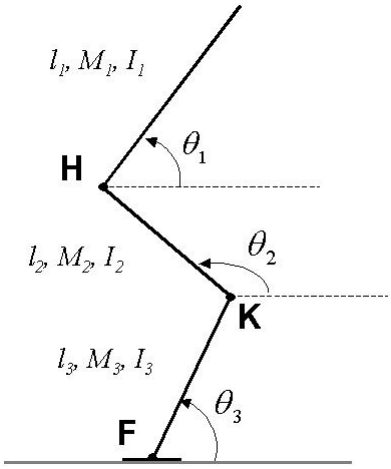

Fig. 1. The model of the articulated hopper

each $\hat{U}$-$\hat{x}$ pair, the cost function $\mathcal{C}(x,u,t) = \mathcal{C}(\hat{x},\hat{U},t)$, $e_{\rm x}$ and inequality constraints $\Psi_{\rm x}(\hat{x},\hat{\dot{x}})$ need to be evaluated. Here the cost function $\mathcal{C}$ may be in fact the discretized version of (1). If the cost function does not reach a local optimum, or one of the constraints is violated, the $\hat{U}$ has to be updated according to some rules. In general, the updating rules of $\hat{U}$ is driven by $e_{\rm x}$ such that $e_{\rm x}$ converges to zero quickly. This shoot-and-update procedure, also known as the *direct single shooting method*, halts when it finds a local optimum of the cost function [5], [6].

III. Modeling of the Hopper

The planar articulated hopper studied here consists of 4 links and 3 frictionless pin joints (Fig. 1). We assume the fourth (foot) link has insignificant mass, inertia, and height. The first three links have length and masses l_i and $M_{\rm i}$ ($i = 1,2,3$). The total mass is $M_{\rm t} = \sum_{i=1}^{3} M_{\rm i}$. Each moment of inertia $I_{\rm i}$ is taken about the CoM of the i^{th} massive link. Each $r_{\rm i}$ is a ratio of the CoM location of the massive link to the link lengths.

The hopping motion is composed of two phases: the stance phase and the flight phase. During the stance phase, we assume the foot does not slip or bounce. During the flight phase, we assume the foot link remains parallel to the ground. The CoM and the foot of the robot are related by

$$P_{\rm g} = P_{\rm f} + f(\Theta), \tag{8}$$

where $P_{\rm g} = (X_{\rm g}, Y_{\rm g})^{\rm T}$ and $P_{\rm f} = (X_{\rm f}, Y_{\rm f})^{\rm T}$ are the positions of the CoM and the foot, respectively, $f(\cdot)$ is a function determined by kinematics, and Θ is defined to be $(\theta_1,\theta_2,\theta_3)^{\rm T}$. Differentiating (8) once gives

$$\dot{P}_{\rm g} = \dot{P}_{\rm f} + \left(\frac{\partial f}{\partial \Theta}\right)^{\rm T} \dot{\Theta}, \tag{9}$$

and twice gives

$$\ddot{P}_{\rm g} = \ddot{P}_{\rm f} + \frac{d}{dt}\left(\frac{\partial f}{\partial \Theta}\right)^{\rm T} \dot{\Theta} + \left(\frac{\partial f}{\partial \Theta}\right)^{\rm T} \ddot{\Theta}. \tag{10}$$

In the stance phase, the foot of the robot firmly grips the ground. Gravity and ground reaction forces (GRFs) act on the robot. In contrast, during the flight phase gravity provides the only external force. When completing the flight phase, the robot foot collides with the ground with a certain velocity. This collision causes an abrupt velocity jump at each joint. We will discuss the flight phase first, followed by a solution to the velocity jump due to the foot/ground collision, and finally go on to explain the stance phase.

A. The nearly-passive flight phase

During the flight phase, the robot has 5 degrees of freedom (DOFs). The generalized coordinate vector, $q^{\rm fl} = (P_{\rm g}^{\rm T}, \Theta^{\rm T})^{\rm T}$, describes the robot's posture in the air.

1) Equations of motion: Using the Euler-Lagrange method leads to the equations of motion (EoM) during the flight phase

$$D^{\rm fl}(q^{\rm fl})\ddot{q}^{\rm fl} + H^{\rm fl}(q^{\rm fl},\dot{q}^{\rm fl})\dot{q}^{\rm fl} + G^{\rm fl}(q^{\rm fl}) = B^{\rm fl}\tau^{\rm fl}, \tag{11}$$

where the superscript fl implies the flight phase, $D^{\rm fl}(q^{\rm fl}) \in \Re^{5\times5}$ is the inertia matrix, $H^{\rm fl}(q^{\rm fl},\dot{q}^{\rm fl}) \in \Re^{5\times5}$ contains the centrifugal and Coriolis terms , and $G^{\rm fl}(q^{\rm fl}) \in \Re^{5}$ is the gravitational torque vector. The vector $\tau^{\rm fl} = (\tau_1^{\rm fl}, \tau_2^{\rm fl}, \tau_3^{\rm fl})^{\rm T}$ contains the torques of the hip, the knee, and the ankle. The term $B^{\rm fl} \in \Re^{5\times3}$ is the constant coefficient matrix of $\tau^{\rm fl}$ and can be determined by using the virtual work principle (based on the selected generalized coordinates.) For the sake of clarity, the superscript fl disappears in the later part of this subsection.

One can partition the generalized coordinates into two parts: $q = (q_1^{\rm T}, q_2^{\rm T})^{\rm T}$, where $q_1 = P_{\rm g}$, and $q_2 = \Theta$. Partitioning the inertia matrix $D(q)$ produces

$$D(q) = \begin{bmatrix} D_1 & 0_{2\times3} \\ 0_{3\times2} & D_2(q_2) \end{bmatrix}, \tag{12}$$

where $D_1 = M_{\rm t} I_{2\times2}$, and $D_2(q_2) \in \Re^{3\times3}$ contains the moment of inertia related to the rotation of the robot's links. Similarly, the partitioned matrices $H(q,\dot{q})$ and B become

$$H(q,\dot{q}) = \begin{bmatrix} 0_{2\times2} & 0_{2\times3} \\ 0_{3\times2} & H_2(q_2,\dot{q}_2) \end{bmatrix}, B = \begin{bmatrix} 0_{2\times3} & B_2 \end{bmatrix}^{\rm T}.$$

The gravity torque vector is $G(q) = (0, M_{\rm t} g, 0, 0, 0)^{\rm T}$, with g being the gravity acceleration. The last three rows of (11) are:

$$D_2(q_2)\ddot{q}_2 + H_2(q_2,\dot{q}_2)\dot{q}_2 = B_2\tau. \tag{13}$$

Simplifying the first two rows of the EoM (11) yields

$$\ddot{X}_{\rm g} = 0, \qquad \ddot{Y}_{\rm g} = -g, \tag{14}$$

which defines the acceleration of the robot's CoM and result in a parabolic trajectory.

Let us denote the time instants at take-off and touchdown by subscripts to and td, respectively, and the step length by $L_{\rm s}$. Assuming that the foot positions at take-off and touchdown are $P_{\rm f,to} = (0,0)^{\rm T}$ and $P_{\rm f,td} = (L_{\rm s},0)^{\rm T}$, respectively, and $\theta_{\rm to}$ and $\theta_{\rm td}$ have been picked in advance, equation (8) produces $P_{\rm g,to}$ and $P_{\rm g,td}$.

In this work, the robot is commanded to hop in a constant average forward speed, $V_{\rm x}$, on an even ground. Hence, the

horizontal velocity $\dot{X}_{\mathrm{g,to}} = \dot{X}_{\mathrm{g,td}} = V_{\mathrm{x}}$. The flight lasts for the duration

$$T^{\mathrm{fl}} = (X_{\mathrm{g,td}} - X_{\mathrm{g,to}})/V_{\mathrm{x}}. \tag{15}$$

The initial and final vertical velocity of the flight phase are

$$\dot{Y}_{\mathrm{g,to}} = (Y_{\mathrm{g,td}} - Y_{\mathrm{g,to}} + \frac{1}{2} g \left(T^{\mathrm{fl}}\right)^2)/T^{\mathrm{fl}}, \tag{16}$$

and

$$\dot{Y}_{\mathrm{g,td}} = \dot{Y}_{\mathrm{g,to}} - gT^{\mathrm{fl}}. \tag{17}$$

2) Angular momentum about the CoM: In the flight phase gravity acts as the only external force at the CoM. Thus the angular momentum of the robot about the CoM is conserved. That is,

$$\Gamma_{\mathrm{g,to}}(\Theta_{\mathrm{to}}, \dot{\Theta}_{\mathrm{to}}) = \Gamma_{\mathrm{g,td}}(\Theta_{\mathrm{td}}, \dot{\Theta}_{\mathrm{td}}), \tag{18}$$

where Γ stands for the angular momentum about the CoM.

3) Energy analysis: The total energy of the robot can be decomposed into three parts: translational kinetic energy, rotational kinetic energy, and potential energy. The total translational kinetic energy plus potential energy is conserved in the flight phase. The rotational kinetic energy of the robot in the flight phase is

$$\Omega_{\mathrm{rot}}(\Theta, \dot{\Theta}) = \frac{1}{2} \dot{q}_2^{\mathrm{T}} D_2(q_2) \dot{q}_2 \tag{19}$$

which depends on the joint torques. Making use of the skew-symmetric property of $\dot{D}_2 - 2H_2$ (see [7]), one can show that $\frac{d(\Omega_{\mathrm{rot}})}{dt} = \dot{q}_2^{\mathrm{T}} B_2 \tau$. During the flight phase, τ is usually not 0, and thus $\frac{d(\Omega_{rot})}{dt} \neq 0$. Therefore, the rotational kinetic energy is not conserved.

However, analysis of human jumping motion [8] reveals that the flight phase consumes much less energy than the stance phase does, and thus it may be viewed nearly passive. This understanding leads to $\Omega_{\mathrm{rot,to}} \approx \Omega_{\mathrm{rot,td}}$. With this in mind, we propose an objective function

$$\mathcal{O}_1 = \gamma \Omega_{\mathrm{rot,to}} + (1-\gamma) \left[\Omega_{\mathrm{rot,to}} - \Omega_{\mathrm{rot,td}}\right]^2, \tag{20}$$

where the constant $\gamma \in [0, 1]$ is a weighting factor. Note that $\mathcal{O}_1$ is a function of $\dot{\Theta}_{\mathrm{to}}$ and $\dot{\Theta}_{\mathrm{td}}$, provided that Θ_{to} and Θ_{td} have been chosen in advance. Minimization of the first term on the right hand side (RHS) of (20) implies a minimal take-off rotational kinetic energy that guarantees the required hopping motion defined by the two commanded parameters L_{s} and V_{x}. Minimization of the second term on the RHS of (20) is inspired by the assumption that the flight phase is nearly passive. In real hopping motion, $|\Omega_{\mathrm{rot,td}} - \Omega_{\mathrm{rot,to}}|$ is usually a small positive quantity. In the case that $\Omega_{\mathrm{rot,to}}$ has been minimized, the minimization of $(\Omega_{\mathrm{rot,to}} - \Omega_{\mathrm{rot,td}})^2$ means that $\Omega_{\mathrm{rot,td}}$ is also minimized. Thus, a second, more concise, novel objective function becomes

$$\mathcal{O}_2 = \gamma \Omega_{\mathrm{rot,to}} + (1-\gamma) \Omega_{\mathrm{rot,td}}. \tag{21}$$

In practice, (21) outperforms (20) in two aspects. It results in more energy-efficient gaits, and also, γ can be chosen in a larger range.

4) Boundary joint velocities: The boundary joint angles are picked manually, but the boundary joint velocities have to be determined. The problem can be stated as:

Solving for $\dot{\Theta}_{\mathrm{to}}$ and $\dot{\Theta}_{\mathrm{td}}$, such that the objective function (21) is minimized. The equality constraints could be

$$\dot{P}_{\mathrm{g,to}} = [V_{\mathrm{x}}, \dot{Y}_{\mathrm{g,to}}]^{\mathrm{T}}, \quad \dot{P}_{\mathrm{g,td}} = [V_{\mathrm{x}}, \dot{Y}_{\mathrm{g,td}}]^{\mathrm{T}}, \quad \Gamma_{\mathrm{g,to}} = \Gamma_{\mathrm{g,td}}.$$

One may also add some inequality constraints, such as the velocity bounds. With this static optimization procedure, $\dot{\Theta}_{\mathrm{to}}$ and $\dot{\Theta}_{\mathrm{td}}$ can easily be solved. In fact, $\dot{\Theta}_{\mathrm{to}}$ is needed, and the true value of $\dot{\Theta}_{\mathrm{td}}$ can then be searched by using dynamic optimization

5) Foot velocity regulation: According to [2], energy loss due to the collision between the foot and the ground is proportional to the foot velocity immediately before the collision. Thus, if the foot velocity were 0, the robot would lose no energy when it collides with the ground. However, this setting may result in a dramatic increase of the control effort in the upcoming stance phase. In this work, the foot velocity at the moment of touchdown is

$$V_{\mathrm{f,td}} = \begin{bmatrix} k_{\mathrm{x}} & 0 \\ 0 & k_{\mathrm{y}} \end{bmatrix} V_{\mathrm{g,td}}, \tag{22}$$

where the constants $k_{\mathrm{x}} \in [0, 1]$ and $k_{\mathrm{y}} \in [0, 1]$ are recommended, trading off efficiency and control effort. A small k_{x} implies acceleration, and a large k_{x} slows down the robot. When k_{x} and k_{y} are both large, the energy loss due to the foot/ground collision is large, but the energy consumption in the stance phase may be small.

B. Collision between the foot and the ground

The collision of the support foot with the ground is assumed to be instantaneous and inelastic. The impulsive force applied by the ground to the robot causes abrupt jump of joint velocities, while joint angles remain unchanged. Equation (8) can be re-arranged into

$$P_{\mathrm{f}} = f_1(q^{\mathrm{fl}}) := P_{\mathrm{g}} - f(\Theta).$$

A Jacobian matrix J is defined as

$$J = \left(\frac{\partial f_1}{\partial q^{\mathrm{fl}}}\right)^{\mathrm{T}}. \tag{23}$$

Let the time instants immediately before and after the collision be denoted by superscripts $-$ and $+$, respectively. According to [9],

$$\dot{q}^+ = \left(I_{5\times 5} - \left(D^{\mathrm{fl}}\right)^{-1} J^{\mathrm{T}} \left(J \left(D^{\mathrm{fl}}\right)^{-1} J^{\mathrm{T}}\right)^{-1} J\right) \dot{q}^-, \tag{24}$$

where $D^{\mathrm{fl}} = D^{\mathrm{fl}}(q)$ and $J = J(q)$ are evaluated at the touchdown. That is, $q = q_{\mathrm{td}}$ is inserted. Similarly, $\dot{q}^- = \dot{q}_{\mathrm{td}}$.

C. The stance phase

During the stance phase, the hopper has 3 DOFs, assuming that the foot is firmly in contact with the ground. We choose the convenient generalized coordinate vector $q^{\mathrm{st}} = \Theta$, which results in the dynamic model of the hopper in stance phase:

$$D^{\mathrm{st}}(q^{\mathrm{st}})\ddot{q}^{\mathrm{st}} + H^{\mathrm{st}}(q^{\mathrm{st}}, \dot{q}^{\mathrm{st}})\dot{q}^{\mathrm{st}} + G^{\mathrm{st}}(q^{\mathrm{st}}) = B^{\mathrm{st}}\tau^{\mathrm{st}}, \tag{25}$$

where $D^{\rm st} \in \Re^{3\times3}$ is the inertia matrix, $H^{\rm st} \in \Re^{3\times3}$ contains centrifugal and Coriolis terms, and $G^{\rm st} \in \Re^3$ is the gravitational torque vector. The matrix $B^{\rm st} \in \Re^{3\times3}$ is the coefficient matrix and depends on the choice of the generalized coordinates.

In the stance phase, the time duration, boundary joint velocities, GRFs, and the Zero-Moment Point (ZMP) have to be determined or bounded. The GRFs include the normal support force of the ground, and the tangential friction force between the foot and the ground.

1) Time duration of the stance phase: We assume that the average forward speed in the stance phase is equal to the horizontal speed in the flight phase ($\bar{V}_{\rm g}^{st} = V_{\rm x}$). Time duration of the stance phase becomes

$$T^{\rm st} = (X_{\rm g,to}^{i+1} - X_{\rm g,td}^{i})/V_{\rm x}, \tag{26}$$

where the superscript i means the i^{th} hopping cycle, and $X_{\rm g,to}^{\rm i+1} - X_{\rm g,td}^{\rm i}$ is the horizontal distance that the robot's CoM travels in the stance phase.

2) Boundary joint velocities: The stance phase is a typical TPBVP. The joint angles at the two boundary points are chosen manually. The boundary joint velocities can be determined by

$$\dot{\Theta}_{\rm initial}^{\rm i,st} = \dot{\Theta}_{\rm td}^{\rm i,+}, \qquad \dot{\Theta}_{\rm final}^{\rm i,st} = \dot{\Theta}_{\rm to}^{\rm i+1},$$

where the superscripts *i* and + have the same meaning as before, and the subscripts *initial* and *final* indicate the two boundary points.

3) The normal GRF: The support force of the ground is always upward. One can formulate this component as

$$F_{\rm y} = M_{\rm t}(g + \ddot{Y}_{\rm g}^{\rm st}) \geq 0, \tag{27}$$

and hence $\ddot{Y}_{\rm g}^{\rm st} \geq -g$.

Since the foot/ground collision is assumed to be instantaneous, during the collision, the resultant normal GRF may be extremely large. To prevent the robot from damage, an active force control could be useful for online gait adaptation. In this off-line gait generator, an upper bound of the normal GRF is considered, $F_{\rm y} \leq F_{\rm y,max}$.

At the end of the stance phase, the normal GRF reduces in amplitude, and vanishes at the take-off of the next flight phase. This implies that $F_{\rm y,final}^{\rm st} = 0$, and

$$\ddot{Y}_{\rm g,final}^{\rm st} = -g. \tag{28}$$

4) The tangential friction force: To ensure that the robot stands on the ground firmly, the horizontal inertial force of the robot must be less than the static friction force between the foot and the ground. That is

$$|\ddot{X}_{\rm g}^{\rm st}| \leq \mu F_{\rm y}/M_{\rm t}. \tag{29}$$

Clearly, if the ground, and hence μ, has been determined, smaller horizontal acceleration is preferred.

5) The ZMP: As assumed, the foot does not rotate during the stance phase, therefore, the ZMP should be located within the foot range. Following the arguments by Popovic *et al.* [10], one can derive the ZMP for the planar hopper as

$$X_{\rm zmp} = \frac{\sum_{\rm i=1}^{3} M_{\rm i}\left[X_{\rm g,i}^{\rm st}\left(\ddot{Y}_{\rm g,i}^{\rm st} + g\right) - Y_{\rm g,i}^{\rm st}\ddot{X}_{\rm g,i}^{\rm st}\right] + \sum_{\rm i=1}^{3} I_{\rm i}\ddot{\theta}_{\rm i}}{M_{\rm t}\left(\ddot{Y}_{\rm g}^{\rm st} + g\right)}. \tag{30}$$

Suppose front and rear lengths of the foot, separated by the ankle, are $l_{\rm f1}$ and $l_{\rm f2}$. One may have $-l_{\rm f2} \leq X_{\rm zmp} \leq l_{\rm f1}$.

IV. Implementations of the Gait Generator

The TPBVP optimization (we use *fmincon()* in Matlab) solves the flight phase and stance phase separately. Each phase has a unique dynamic model, and also differs in constraints. However, the implementation method of the dynamic models, the definitions of the decision vectors, and the cost functions are almost identical for the two hopping phases.

A. Model implementation

The second-order dynamic models (11) and (25) are implemented with Simulink blocks. However, the direct single shooting method can cause the term $H(q,\dot{q})\dot{q}$ to become too large. To avoid crashing the simulation, a saturation block limits the output of this term.

B. The decision vector

The decision vector is defined as

$$U = \left[\tau(0)^{\rm T}, \tau(1)^{\rm T}, \cdots, \tau(N-1)^{\rm T}\right]^{\rm T}, \tag{31}$$

where $\tau(k) = [\tau_1(k), \tau_2(k), \tau_3(k)]^{\rm T}$, and k indicates the k^{th} discretized time interval. $\hat{U}_0$, the initial estimate of U, is taken a zero vector with same dimension as U. The torque limits of the joints form the bounds of the decision vector and thus $U_{\rm min} \leq U \leq U_{\rm max}$.

C. The cost functions

The cost functions in the two hopping phases have the same form

$$\mathcal{C}_1 = \frac{1}{2}U^{\rm T}U\Delta t, \tag{32}$$

where Δt is the time interval with an assumption that the phase duration, $T^{\rm fl}$ or $T^{\rm st}$, is discretized equally. The number of time intervals, N, may be different in the two phases. Clearly, this cost function implies the *least control effort*.

D. Constraints in the flight phase

In the flight phase, the boundary values and (22) are the equality constraints. The physical ranges of the joints constitute the linear inequality constraints

$$\begin{aligned} \theta_{\rm 1,min}^{\rm fl} \leq \theta_1^{\rm fl} \leq \theta_{\rm 1,max}^{\rm fl}, \\ \alpha_{\rm min}^{\rm fl} \leq \theta_2^{\rm fl} - \theta_3^{\rm fl} \leq \alpha_{\rm max}^{\rm fl}, \\ \theta_{\rm 3,min}^{\rm fl} \leq \theta_3^{\rm fl} \leq \theta_{\rm 3,max}^{\rm fl}, \end{aligned}$$

where α is the relative angle between θ_2 and θ_3. One may add new linear inequality constraints, such as joint velocity bounds. All the inequality constraints form the set $\Psi_{\rm x}^{\rm fl}(x,\dot{x})$.

TABLE I
THE CONSTANTS IN THE IMPLEMENTATION

Parameters	Values	Units	Parameters	Values	Units
M_1	8	kg	I_1	0.4	kg-m^2
M_2	2	kg	I_2	0.02	kg-m^2
M_3	0.8	kg	I_3	0.01	kg-m^2
l_1	0.6	m	r_1	0.4	-
l_2	0.35	m	r_2	0.6	-
l_3	0.4	m	r_3	0.6	-
l_{f1}	0.15	m	l_{f2}	0.05	m
g	9.81	m/s^2	-	-	-
$\theta_{1,\min}$	45	deg	$\theta_{1,\max}$	90	deg
$\alpha_{\min}$	0	deg	$\alpha_{\max}$	150	deg
$\theta_{3,\min}$	30	deg	$\theta_{3,\max}$	120	deg
$\tau_{1,\min}$	-40	N-m	$\tau_{1,\max}$	40	N-m
$\tau_{2,\min}$	-30	N-m	$\tau_{2,\max}$	30	N-m
$\tau_{3,\min}$	-30	N-m	$\tau_{3,\max}$	30	N-m

TABLE II
THE PARAMETERS FOR THE SIMULATED CASE

Parameters	Values	Units
L_{s}	0.25	m
V_{x}	0.5	m/s
Θ_{to}	$(80, 110, 75)^{\mathrm{T}}$	deg
Θ_{td}	$(75, 115, 95)^{\mathrm{T}}$	deg
γ	0.3	-
μ	0.6	-
k_{x}	0.5	-
k_{y}	0.5	-
$F_{\mathrm{y,max}}$	$2M_{\mathrm{t}}g$	N

E. Constraints in the stance phase

In the stance phase, the boundary values and (28) are the equality constraints. The linear inequality constraints could be

$$\theta^{\mathrm{st}}_{1,\min} \le \theta^{\mathrm{st}}_1 \le \theta^{\mathrm{st}}_{1,\max},$$
$$\alpha^{\mathrm{st}}_{\min} \le \theta^{\mathrm{st}}_2 - \theta^{\mathrm{st}}_3 \le \alpha^{\mathrm{st}}_{\max},$$
$$\theta^{\mathrm{st}}_{3,\min} \le \theta^{\mathrm{st}}_3 \le \theta^{\mathrm{st}}_{3,\max},$$

and other linear inequality constraints on joint velocities can also be added. The bound values in the stance phase may not be equal to their counterparts in the flight phase, but in our implementations, the corresponding bounds are designed to be identical. Constraints on the normal GRF, the tangential friction force, and the ZMP are the nonlinear inequality constraints. All linear and nonlinear inequality constraints form the set of $\Psi^{\mathrm{st}}_{\mathrm{x}}(x, \dot{x})$.

V. SIMULATION RESULTS AND DISCUSSIONS

In the simulation, the robot hops with a periodic gait on even ground (with parameters listed in Tables I, II). The step length L_{s} and the average forward speed V_{x} are the commanded inputs. In the stance phase, the CoM of the robot behaves like an elastic inverted pendulum (Fig. 2). The joint angle trajectories are intuitively satisfying from the perspective of achieving human-like motion (Figs. 3). The foot/ground collision causes large velocity jump at the knee (Fig. 4). The joint torques fall within reasonable values for achieving in a real robot (Fig. 5).

The ZMP trajectory, the horizontal and vertical accelerations of the robot's CoM, and normal GRF in one stance

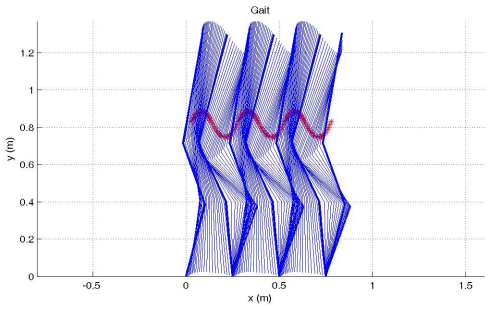

Fig. 2. The synthesized hopping cycles

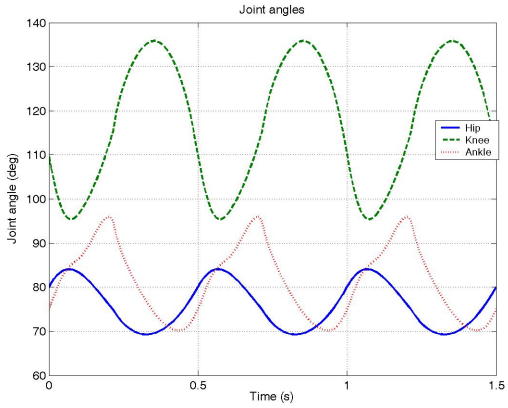

Fig. 3. The joint angles

phase are presented in the three panes of Fig. 6, respectively. Clearly, the ZMP is within the support foot, and hence the robot will not tip over. It is also observed that the vertical acceleration of the CoM, represented by the dashed curve in the middle pane, grows up from $-5.9m/s^2$ to g, and then decreases to $-g$. This implies that the normal GRF (see the third pane) increases from $45.6N$ to $F_{\mathrm{y,max}}$, then decreases dramatically, and finally vanishes at the take-off. Due to (29), the horizontal acceleration of the CoM, specified by the solid curve in the middle pane, becomes 0 at the end of the stance phase. Therefore, the horizontal acceleration of the CoM is continuous over the whole hopping cycle.

The maximum normal GRF, $F_{\mathrm{y,max}}$, is set to be 2 times of the robot weight. The limit seems to be reasonable for a real running robot, according to [11]. When a real running robot is built, the foot has to be specially made such that large impact impulse can be partly absorbed. The static friction coefficient, μ, is chosen to be 0.6, which is a typical value in real world.

If smaller $F_{\mathrm{y,max}}$ and μ have to be used, one may reduce V_{x} to achieve safe hopping.

In each hopping cycle, the energy consumed in the flight phase and the stance phase are 4.2 Joules and 39.0 Joules, respectively, and the flight phase is indeed nearly-passive, as

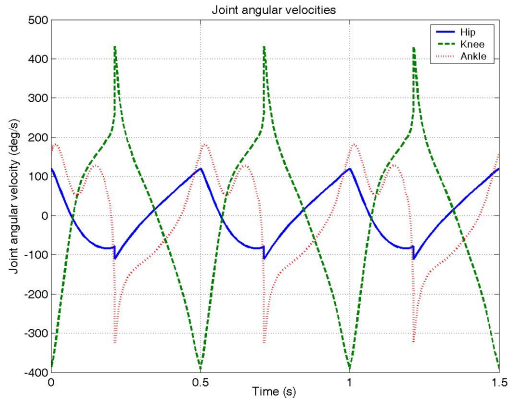

Fig. 4. The joint velocities

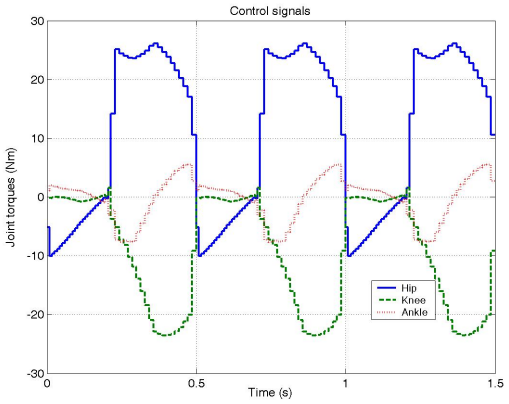

Fig. 5. The joint torques

assumed. The cost of transportation of the robot is about 1.6 J/N-m, and is about half of the Honda Asimo's walking [12].

The use of the ZMP criterion in synthesis of running-like gaits is controversial [13], and it may be too conservative for high-speed motions. Using it here as nonlinear inequality constraints proves that the proposed approach works well, even under such strict constraints. An energy-efficient hopping gait can easily be searched, if the ZMP constraints are not used.

VI. Conclusions

This paper presents an off-line running-like gait generator. Assuming the flight phase is nearly passive, one can solve the boundary joint velocities of the flight phase by a static optimization procedure, provided that the boundary joint angles are pre-determined. Then, the flight phase and the stance phase can be dynamically optimized, with the velocity jumps of the joints being predicted by a simple collision model. The well known direct single shooting method synthesizes the gait. Simulation results show the robot hopping as predicted, with an energy efficient and intuitively satisfying gait.

The proposed method has been extended to deal with hopping up stairs, jumping across an obstacle, and bipedal running in simulations. The synthesized gaits are elegant

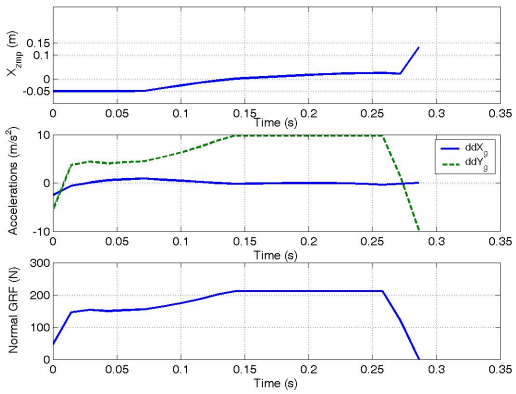

Fig. 6. The ZMP trajectory, horizontal and vertical acceleration of the CoM, and the normal GRF in one stance phase

with satisfactory energy-efficiency in the sense of cost of transportation. The effectiveness of the proposed method needs further verifications by experiments on real robots. This issue is left as our future work.

References

[1] H. De Man, D. Lefeber, and J. Vermeulen, "Control on irregular terrain of a hopping robot with one articulated leg," *ICAR Workshop II: New Approaches on Dynamic Walking and Climbing Machines*, Monterey, California, USA, pp. 72-76, 1997.

[2] J. Vermeulen, *Trajectory generation for planar hopping and walking robots: an objective parameter and angular momentum approach*, PhD Dissertation, Vrije Universiteit Brussel, Belgium, May 2004.

[3] T. Ikeda, Y. Iwatani, K. Suse, and T. Mita, "Analysis and design of running robots in touchdown phase," *ICCA99*, pp. 496-501, Hawaii, USA, Aug. 22-27, 1999.

[4] S. Hyon, T. Emura, and T. Mita, "Dynamics-based control of a one-legged hopping robot," *Journal of Systems and Control Engineering*, 217(2): 83-98, 2003.

[5] L. Roussel, C. Canudas de Wit, and A. Goswami, "Comparative study of method for energy-optimal gait eneration for biped robots," in *International Conference on Informatics and Control*, pp. 1205-1212, St. Petersburg, June 1997.

[6] V. M. Becerra, "Solving optimal control problems with state constraints using nonlinear programming and simulation tools," *IEEE Transactions on education*, Vol. 47, No. 3, pp. 377-384, Aug. 2004.

[7] M. W. Spong, Seth Hutchinson, and M. Vidyasagar, *Robot modeling and control*, John Wiley & Sons, 2006.

[8] A. Meghdari, and M. Aryanpour, "Dynamic modeling and analysis of the human jumping process," *Journal of Intelligent and Robotic Systems*, 37:97-115, 2003.

[9] Y. Fujimoto, "Trajectory generation of biped running robot with minimum energy consumption," *in Proc. of the IEEE International Conference on Robotics and Automation*, pp. 3803-3808, 2004.

[10] M. B. Popovic, A. Goswami, and H. Herr, "Ground reference points in legged locomotion: definitions, biological trajectories and control implications," *International Journal of Robotics Research*, Vol. 24, No. 12, pp. 1013-1032, Dec. 2005.

[11] T. Nagasaki, S. Kajita, K. Yokoi, K. Kaneko, and K. Tanie, "Running pattern generation and its evaluation using a realistic humanoid model," *in Proc. of IEEE International Conference on Robotics and Automation*, pp. 1336-1342, 2003.

[12] S. Collins, A. Ruina, R. Tedrake, and M.Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, Vol. 307. no. 5712, pp. 1082-1085, Feb. 2005.

[13] K. Kondak, and G. Hommel, "Control of online computation of stable movement for biped robots," *IEEE/RSJ Intl. Conf. Intelligent Robots and Systems*, Las Vegas, USA, pp. 874-879, 2003.