doi:10.3772/j.issn.1006-6748.2016.03.012

Design of trotting controller for the position-controlled quadruped robot①

Zhang Guoteng (张国腾)*, Liu Jinchang②**, Rong Xuewen*, Li Yibin*, Chai Hui*, Li Bin*, Zhang Hui*, Zhang Shuaishuai*

(*School of Control Science and Engineering, Shandong University, Jinan 250061, P. R. China)

(**High Technology Research Development Center, Ministry of Science and Technology of the People's Republic of China, Beijing 100044, P. R. China)

Abstract

This work presents a controller designed for position-controlled quadrupedal dynamic locomotion, aiming at simple and robust trotting control. The controller takes the torso attitude angles and velocities into planning foot trajectories. Firstly design of the servo motor actuated quadruped robot is introduced and the kinematic equations are deduced. Then a scheme is presented for controlling the robot torso attitude based on the virtual leg model. Furthermore, it demonstrates the design of the controller which enables the robot to have a wide range of trotting gaits and omni-directional motions. Finally, results of robust trotting in various speeds, path tracking and push recovery in simulation are reported, and results of trotting on real quadruped robots will be studied.

Key words: quadruped robot, position control, torso attitude, foot trajectory, trotting locomotion

0 Introduction

Legged robots have shown great superiority in terms of agility and versatility compared to their wheeled or tracked counterparts. They are able to navigate on much more uneven and rough terrains. These capabilities of legged robots have made it a new hot spot in robots research. Nevertheless, legged robots are more complicated in structure and difficult to control than wheeled or tracked robots. The theories concerned with legged robots are still in the development stage[1].

Legged animals in nature could keep balance during motions even after they suffer from unexpected disturbances. To date, lots of research have been made towards bridging the gap between locomotivity and balancing skills of legged robotic systems and that of real animals. The most well-known intuitive method for controlling legged robots is the three part locomotion algorithms developed by Raibert[2,3]. The most famous quadruped robots built by Boston Dynamics, BigDog[4] and LS3[5] are believed being developed by Raibert's three-part locomotion algorithms[6] though no details about the control methods have been published. Besides, quadruped robots like TITAN-VIII[7] or LittleDog [8] took the ZMP (zero moment point) method in their locomotion control. The CPG-based neural network with reflex feedbacks and interconnections qualified the Tekken robot with impressive trotting skills[9]. The robot HyQ was controlled by a reactive controller consisting of a CPG-based trajectory generator and an attitude controller[10]. And the Autonomous Systems Laboratory combined with virtual model control, PD controller and virtual forces in controlling their quadruped robot StarlETH[11].

Though some achievements have been got in the locomotion and balance control of legged robots, there is still a long way to go. The algorithms mentioned above generally need precise sensors, sophisticated computing, advanced actuators and possess force-control sections. These qualifications are unavailable for the "civilian-style" robots.

In 2012, Center for Robotics of Shandong University developed a servo motor actuated small mammal bionic quadruped robot called LittleCalf. The robot was designed to serve as a platform to study not only its gaits and joint action, the generate and switch of gaits as well as the ability to keep balance, but also the bionic kinematics.

Owning to the limitation of the servo motors in LittleCalf, only position control is available for the robot joints. Traditional position-controlled quadruped robots

① Supported by the National Natural Science Foundation of China (No. 61233014, 61305130), China Postdoctoral Science Foundation (No. 2013M541912) and the Shandong Provincial Natural Science Foundation (No. ZR2013FQ003, ZR2013EEM027).

② To whom correspondence should be addressed. E-mail: liujc@htrdc.com

Received on Apr. 1, 2015

introduce foot trajectory planning in controlling the motion of the robots. Refs[1,12-15] show some kinds of control strategies for position-controlled quadruped robots. Yet those works paid great attention on the motion of the robot feet instead of the overall attitude. And the robust of the robots to external disturbances are not verified.

In this study, considering the peculiarity of the LittleCalf platform, a novel trotting controller for position-controlled quadruped robots is demonstrated. The controller takes the attitude angles and velocities of the torso into planning the foot trajectories and drive the robot to move with trotting gait as well as to maintain the torso attitude.

This paper is organized as follows: Section 1 gives an overview about the LittleCalf robot. Kinematics of the robot are deduced in Section 2. Section 3 gives the torso attitude control algorithm. Section 4 demonstrates the trotting controller designed for the LittleCalf robotic platform and Section 5 reports the experimental results obtained from simulation as well as the physical prototype. Finally, Section 6 contains the conclusions and further work.

1 Overview of LittleCalf

1.1 Mechanism design

As shown in Fig. 1, the robot consists of one torso and four legs. Each leg has a rolling rotary joint in the hip, a pitching rotary joint in the hip and a pitching rotary joint in the knee, which allows the foot to move in a three dimensional workspace around the hip. This reduces the complexity of the quadruped machine, but still enables static walking over rough terrain, robot balance and robust dynamic gaits[16]. The joints are configurated to be centrosymmetric with knees of the front and hind legs pointing to each other, since Zhang et al. have concluded that this kind of configuration is beneficial for slipping-inhibition, and can improve the stability of motion[17].

The joints topology is shown in Fig. 2.

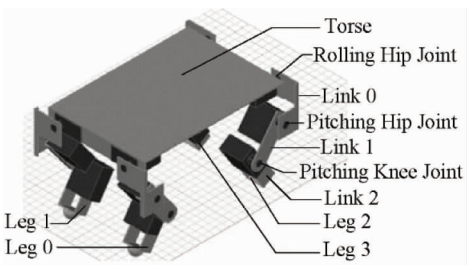


Fig. 1 Three dimensional prototype of LittleCalf

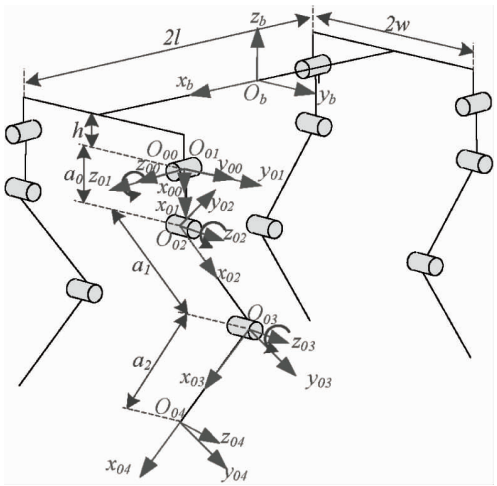


Fig. 2 Joints topology and D-H coordinate frames of LittleCalf

1.2 Control system

Servo motor MG995 is chosen as the actuator of the robot's joint. This kind of servo motor has the advantage of high torque, short response time, light weight and easy to control. A high capacity lithium-ion battery is fixed under the torso to supply the whole

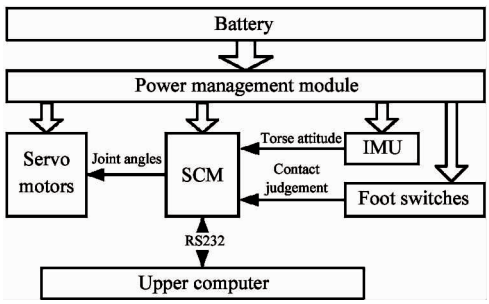


Fig. 3 Block diagram of control system

Table 1 Major technical specifications of LittleCalf

Specifications	Values
Weight (with battery)	1.6kg
Dimensions (fully stretched legs)	230mm × 160mm × 190mm (Length × Width × Height)
Torso size	$2l$ = 200mm $2w$ = 110mm h = 10mm
Leg Length	a_0 = 30mm a_1 = 50mm a_2 = 60mm
DOFs per leg	3 active (rotary)
Joint range of motion	90 degrees
Max rotate speed of joint	0.13s/60degrees
Max torque of joint	1.3N · m

system. Onboard control circuit is based on the single chip computer (SCM) MC9S12XS128 from Freescale, coupled with the power management module, motor drive module, environmental perception module (IMU to detect attitude of the torso, and foot switch to detect whether the foot touches the ground) and communication module.

Block diagram of the control system is shown in Fig. 3. Table 1 shows the major technical specifications of LittleCalf.

2 Kinematic equations

2.1 D-H coordinate frames and link parameters

As shown in Fig. 2, the origin of the body fixed coordinate frame $\{O_b\}$ locates in the geometric center of the torso, and x_b points to the forward direction, z_b points opposite to the gravity direction. And y_b axis is confirmed using right-hand rule. There are four coordinate frames noted as $\{O_{i0}\}$ ($i=0, 1, 2, 3$) fixed on the four corners of the torso respectively. The coordinate frames fixed on link k noted as $\{O_{ik}\}$ of four legs are established according to D-H rules[14]. Fig. 2 shows coordinate frames of leg 0. Coordinate frames of the other three legs are establish accordingly. Since the mechanical configurations of the four legs are extremely identical, the coordinate frames and transformation matrices of four legs are identical too. The only difference is the posture between fore legs and rear legs. The link parameters of LittleCalf are illustrated in Fig. 2 and listed in Table 2.

Table 2 Link parameters of LittleCalf

link k	a_{k-1}	α_{k-1}	joint variables		theoretical range of θ_{ik}	
			d_{ik}	θ_{ik}	fore legs	hind legs
1	0	0°	0	θ_{i0}	(−45°, 45°)	
2	a_0	−90°	0	θ_{i1}	(0°, 90°)	(−90°, 0°)
3	a_1	0°	0	θ_{i2}	(−90°, 0°)	(0°, 90°)
4	a_2	0°	0	/	/	/

2.2 Forward kinematic equations

Since the four legs of LittleCalf have the same D-H coordinate frames and link parameters, they have same forward kinematic equations from $\{O_{i4}\}$ to $\{O_{i0}\}$. The transformations from $\{O_{i0}\}$ to $\{O_b\}$ for four legs can be expressed by the following transformation matrix with different values of δ and λ:

$$ {}^{b}\boldsymbol{T}_{i0} = \begin{bmatrix} 0 & 0 & 0 & \delta l \\ 0 & 1 & 0 & \lambda w \\ -1 & 0 & 0 & -h \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

where l, w and h are geometry parameters of the torso specified in Fig. 2, δ and λ are sign flags which are defined as

$$ \delta = \begin{cases} 1 & i = 0, 1 \\ -1 & i = 2, 3 \end{cases} \tag{2} $$

$$ \lambda = \begin{cases} 1 & i = 0, 3 \\ -1 & i = 1, 2 \end{cases} \tag{3} $$

The coordinates of one foot with respect to $\{O_{i0}\}$ corresponding to the same leg can be obtained easily through homogeneous transformations. It is given by

$$ {}^{i0}\boldsymbol{T}_{i4} = \begin{bmatrix} c_0c_{12} & -c_0s_{12} & -s_0 & a_0c_0 + a_1c_0c_1 + a_2c_0c_{12} \\ s_0c_{12} & -s_0s_{12} & c_0 & a_0s_0 + a_1s_0c_1 + a_2s_0c_{12} \\ -s_{12} & -c_{12} & 0 & -a_1s_1 - a_2s_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

where $c_k = \cos\theta_{ik}$, $s_k = \sin\theta_{ik}$, $s_{12} = \sin(\theta_{i1} + \theta_{i2})$, $c_{12} = \cos(\theta_{i1} + \theta_{i2})$.

Eq. (4) is the forward kinematic equation for four legs. The coordinates of four feet in $\{O_{i0}\}$ can be solved respectively from it. Furthermore, the coordinates of four feet in $\{O_b\}$ can be obtained by premultiplying transformation matrix ${}^{i0}T_{i4}$ with ${}^{b}T_{i0}$ respectively. And the coordinate $\{x_i, y_i, z_i\}^T$ of one foot in $\{O_b\}$ can be gained by

$$ \begin{cases} x_i = -a_1s_1 - a_2s_{12} + \delta l \\ y_i = a_0s_0 + a_1s_0c_1 + a_2s_0c_{12} + \lambda w \\ z_i = a_0c_0 + a_1c_0c_1 + a_2c_0c_{12} - h \end{cases} \tag{5} $$

And the jacobian matrix for the leg is

$$ \boldsymbol{J} = \begin{bmatrix} 0 & -a_0c_1 - a_2c_{12} & -a_2c_{12} \\ c_0(a_0 + a_1c_1 + a_2c_{12}) & -s_0(a_1s_1 + a_2s_{12}) & -a_2s_0s_{12} \\ s_0(a_0 + a_1c_1 + a_2c_{12}) & c_0(a_1s_1 + a_2s_{12}) & a_2c_0s_{12} \end{bmatrix} \tag{6} $$

2.3 Inverse kinematic equations

The inverse kinematic analysis is necessary for motion planning and controlling. Although the forward kinematic equations for four legs from $\{O_{i4}\}$ to $\{O_{i0}\}$ are identical, the inverse kinematic equations of joint variables θ_{i1} and θ_{i2} are not the same since the fore legs and the rear legs are indifferent postures. The feet of

LittleCalf are simplified to four points in this paper. If given the coordinates of one foot with $\{x_i, y_i, z_i\}^T$ in $\{O_b\}$, the corresponding joint variables θ_{ik} can be resolved from the inverse kinematic equations. Moreover, its coordinates in other frames can be obtained by forward and inverse homogenous transformation. Ref. [18] gives the detailed derivation for inverse kinematic equations of a quadruped robot. This study gives the results directly as

$$\begin{cases} \theta_{i0} = \arctan\left(\dfrac{y_i - \lambda w}{-z_i - h}\right) \\ \theta_{i1} = \left(\varphi - \arctan\left(\dfrac{x_i - \delta l}{-a_0 - (z_i + h)/\cos\theta_0}\right)\right) \cdot \delta \\ \theta_{i2} = \left(\arccos\left(\dfrac{a_1^2 + a_2^2 - \xi^2}{2a_1 a_2}\right) - \pi\right) \cdot \delta \end{cases} \tag{7}$$

Among them

$$\xi = \sqrt{\left(a_0 + \frac{z_i + h}{\cos\theta_0}\right)^2 + (x_i - \delta l)^2} \tag{8}$$

$$\varphi = \arccos\left(\frac{a_1^2 + \xi^2 - a_2^2}{2a_1\xi}\right) \tag{9}$$

2.4 Workspace of the foot

Given the mechanical parameters of LittleCalf and rotation range of the joints, using Eq. (5), the foot workspaces of the robot could be got. The workspace can help us in the motion planning of each foot. Take leg 0 as example, the foot workspace could be got show in Fig. 4. And the trajectory of left front foot in this workspace could be planned.

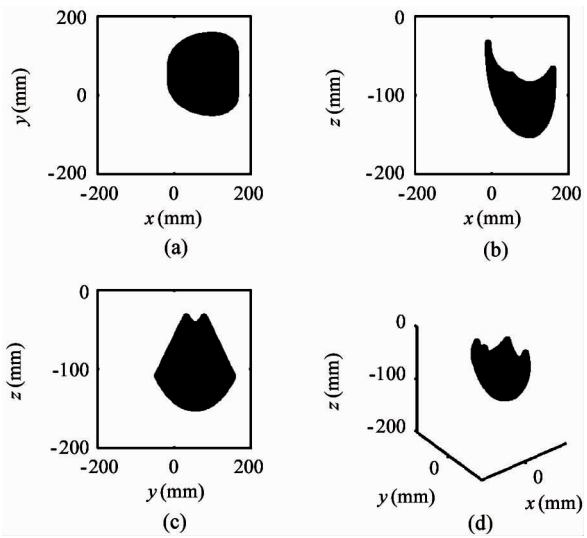


Fig. 4 Workspace of the foot of leg 0 with respect to torso fixed frame $\{O_b\}$: (a), (b) and (c) show the projection of the workspace on the *x*-*y* plane, *x*-*z* plane and *y*-*z* plane, (d) shows foot workspace in 3D view

Point $\{x_{0,ini}, y_{0,ini}, z_{0,ini}\}^T = \{100, 55, -130\}^T$ is chosen as the initial foot position of leg 0, since there are rather big space for the foot to move near this point. Initial position of the other three feet are confirmed similarly.

3 Control torso attitude

In nature, quadruped animals employ different kinds of gaits to move. The trotting is the kind of gait that the diagonal legs move together, which exhibits good energy efficiency over a wide range of running speed, showing no significant pitch or roll motion during each stride and therefore is often seen in nature[19]. Many famous quadruped robots such as HyQ[20], BigDog[4], LS3[5], have selected trotting gait as their primary gait. Thus this work majors in controlling the trotting gait of the robot.

The trot is a kind of dynamic gait and in support phase the diagonal standing legs synchronously support the body. Thus the torso would easily rotate about the body diagonal line in trotting. And attention must be paid on controlling the torso attitude.

The concept of the virtual leg is invoked to simplify the control algorithms of the quadruped running. Since the trotting gait pairs their diagonal legs, and the diagonal feet will move almost the same way from their hips and exert equal forces on the ground, their behavior is precisely equivalent to the behavior of the virtual leg[3,21].

Fig. 5 shows the correspondences between the trot gait and the equivalent virtual model. The motion of the trotting quadruped robot can be simplified to a virtual biped one and further to the one with single leg. Then the motion of one-leg model are furthers simplified to the two-dimensional plane. Fig. 6 shows the planar graph of the simplified robot model and the coordinate system built. All the model variables in Fig. 6 are defined in Table 3.

Since LittleCalf is a position – controlled robot, the joint torques can not be directly regulated to control the torso attitude. But the foot velocity can be programmed so as to make the hip exert equivalent torque.

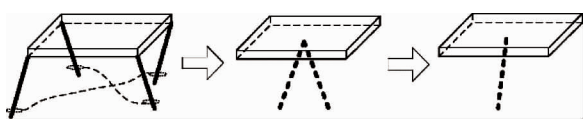


Fig. 5 Correspondences between the trot gait and the equivalent virtual model

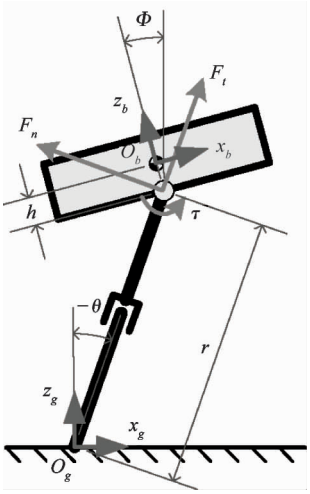


Fig. 6 Planar graph of the simplified robot model

Table 3 Model variables in Fig. 6 and the descriptions

Variables	Descriptions
O_g	coordinate frame attached to the ground
O_b	coordinate frame attached to the COM of the torso
g	gravitational acceleration
M	torso mass
I	torso inertia
θ	leg angle w. r. t. vertical
ϕ	torso attitude angle
ϕ_d	desired torso attitude angle
h	height between the torso COM and the hip
r	leg length
τ	torque exerted by the hip
F_t, F_n	forces acting at the hip between the leg and torso F_t acts tangent to the leg, and F_n acts perpendicular to the leg
$({}^g x_h, {}^g z_h)$	hip position in frame O_g
$({}^g x_m, {}^g z_m)$	position of the torso COM in frame O_g
$({}^b x_s, {}^b z_s)$	position of the support foot in frame O_b

This model makes the simplifying assumptions of negligible frictional losses, zero leg mass, and a total center of mass (COM) located at the torso. Dynamic equations for the model are derived as

$$ {}^g\ddot{x}_h = -\ddot{r}\sin\theta - r\ddot{\theta}\cos\theta + r\dot{\theta}^2\sin\theta - 2\dot{r}\dot{\theta}\cos\theta \tag{10} $$

$$ {}^g\ddot{z}_h = \ddot{r}\cos\theta - r\ddot{\theta}\sin\theta - r\dot{\theta}^2\cos\theta - 2\dot{r}\dot{\theta}\sin\theta \tag{11} $$

$$ {}^g\ddot{x}_m = {}^g\ddot{x}_h + h(\dot{\phi}^2\sin\phi - \ddot{\phi}\cos\phi) \tag{12} $$

$$ {}^g\ddot{z}_m = {}^g\ddot{z}_h - h(\dot{\phi}^2\cos\varphi - \ddot{\phi}\sin\varphi) \tag{13} $$

$$ {}^g\ddot{x}_m M = -F_t\sin\theta - F_n\cos\theta \tag{14} $$

$$ {}^g\ddot{z}_m M = F_t\cos\theta - F_n\sin\theta - Mg \tag{15} $$

$$ I\ddot{\theta} = \tau + F_t h\sin(\phi-\theta) - F_n h\cos(\phi-\theta) \tag{16} $$

$$ \tau = -F_n r \tag{17} $$

Assume the controller could maintain the robot torso relatively level and ϕ is closed to 0. The simplification that $\sin\phi \approx 0$, $\cos\phi \approx 1$, $\sin(\phi-\theta) \approx \sin(-\theta)$ and $\cos(\phi-\theta) \approx \cos(-\theta)$ can be got. Then Eqs(12) (13) and (16) can translate to

$$ {}^g\ddot{x}_m = {}^g\ddot{x}_h - \ddot{\phi}h \tag{18} $$

$$ {}^g\ddot{z}_m = {}^g\ddot{z}_h - \dot{\phi}^2 h \tag{19} $$

$$ I\ddot{\theta} = \tau - F_t h\sin\theta - F_n h\cos\theta \tag{20} $$

Combine Eqs(10) (11) (14) (15) (17) (18) (19) (20) and eliminate ${}^g\ddot{x}_m$, ${}^g\ddot{z}_m$, F_t, F_n and τ, the following can be got:

$$ \ddot{\phi} = \frac{{}^g\ddot{x}_h M(h + {}^g z_h) + {}^g x_h M(g - h\dot{\phi}^2 + {}^g\ddot{z}_h)}{I + Mh^2 + {}^g z_h Mh} \tag{21} $$

It is hoped that the z coordinate of the foot should remain unchanged so as to avoid the torso vibration. Thus ${}^g_h$ will be equal to zero and Eq. (21) can be simplified to

$$ \ddot{\phi} = \frac{{}^g\ddot{x}_h M(h + {}^g z_h) + {}^g x_h M(g - h\dot{\phi}^2)}{I + Mh^2 + {}^g z_h Mh} \tag{22} $$

Since Eq. (22) is a nonlinear differential equation, it is difficult to obtain the universal analytic solutions. But the qualitative analysis of it can be taken. Integrate Eq. (22) with respect to time, yields:

$$ \dot{\phi} = \frac{({}^g\dot{x}_h - {}^g\dot{x}_{h0})M(h + {}^g z_h) + \int^g x_h M(g - h\dot{\phi}^2)dt}{I + Mh^2 + {}^g z_h Mh} + \dot{\phi}_0 $$
$$ = A\,{}^g\dot{x}_h + B \tag{23} $$

where subscribe 0 indicates the initial value, and

$$ A = \frac{M(h + {}^g z_h)}{I + Mh^2 + {}^g z_h Mh} \tag{24} $$

$$ B = \frac{\int^g x_h M(g - h\dot{\phi}^2)dt - {}^g\dot{x}_{h0}M(h + {}^g z_h)}{I + Mh^2 + {}^g z_h Mh} + \dot{\phi}_0 \tag{25} $$

Note that A is always positive. So if given the desired torso attitude angle φ_d and

· Suppose $\phi > \phi_d$ and $B < 0$, as long as ${}^g\dot{x}_h < 0$ then will surely be negative, which will drive ϕ to ϕ_d.

· Suppose $\phi > \phi_d$ and $B > 0$, if ${}^g\dot{x}_h < -B/A$ then will be negative and drive ϕ to ϕ_d.

· Suppose $\phi < \phi_d$ and $B > 0$, as long as ${}^g\dot{x}_h > 0$ then will surely be positive, which will drive ϕ to ϕ_d.

· Suppose $\phi < \phi_d$ and $B < 0$, if ${}^g\dot{x}_h > -B/A$ then will be positive and drive ϕ to ϕ_d.

It can be found that so long as ${}^g\dot{x}_h$ and $(\phi - \phi_d)$

are of opposite sign and the absolute value of ${}^{g}\dot{x}_h$ is larger than a certain value, the angular velocity will be inverse with the offset of torso attitude angle hence drive the torso to the desired state. Given these qualitative conclusions, it can be set that ${}^{g}\dot{x}_h$ is inversely proportional to $(\phi - \phi_d)$. And to shorten the system response time, a differential section is added. Thus a PD controller is involved in the hip horizontal velocity:

$$ {}^{g}\dot{x}_h = -k_\phi(\phi - \phi_d) - k_{\dot\phi}(\dot\phi - \dot\phi_d) \tag{26} $$

Under the assumption of negligible frictional losses, the feet do not move with respect to the ground during the stance phase and the backward motion of a support foot with respect to the body coordinate is equal to the forward motion of the body with respect to the ground. That is

$$ \begin{aligned} {}^{g}\dot{x}_s &= -{}^{g}\dot{x}_h\cos(\phi) + {}^{g}x_h\dot\phi\sin(\phi) \\ &\quad - {}^{g}\dot{z}_h\sin(\phi) - {}^{g}z_h\dot\phi\cos(\phi) \\ &\approx -{}^{g}\dot{x}_h - {}^{g}z_h\dot\phi \\ &= k_\phi(\phi - \phi_d) + (k_{\dot\phi} - {}^{g}z_h)(\dot\phi - \dot\phi_d) \\ &= k_p(\phi - \phi_d) + k_d(\dot\phi - \dot\phi_d) \end{aligned} \tag{27} $$

where k_p and k_d are gains and $k_p = k_\phi$, $k_d = k_{\dot\phi} - {}^{g}z_h$. Eq. (27) shows that it would be able to control the torso attitude by involving a PD section of the torso attitude angle into the foot's horizontal velocity. k_p and k_d are experimentally evaluated and can be tuned according to the level of responsiveness that is desired from the robot.

4 Trotting controller

The control of the rectilinear motion of the robot is divided into the support phase control and flight phase control. Both of them are designed based on the virtual leg model and later on translated to the quadruped model. Then the yaw controller is involved to modify the foot position and enable the robot to rotate around the yaw axis.

For simplicity, all of the following coordinates in this paper are defined in the body frame unless noted otherwise.

4.1 Support Phase

For the foot in the support phase, the z coordinate of the foot should remain unchanged as mentioned before. Meanwhile the x coordinate would customarily be function of the starting point, the velocity and time. See

$$ \begin{cases} x_s(t) = x_{s0} - \int_0^t \dot{x}dt \\ z_s(t) = z_0 \end{cases} \tag{28} $$

where

x_s and z_s are the coordinates of the support foot.

$\dot{x}$ is the moving velocity of the robot.

x_{s0} is the x coordinate of the starting point, i. e., the vertical position of the support foot at time 0.

t is the current time.

z_0 is the initial vertical coordinate of support foot.

Generally the moving velocity $\dot{x}$ is equal to the desired velocity of the robot. That is, $\dot{x} = \dot{x}_d$, where $\dot{x}_d$ indicates the desired velocity. Nevertheless in Section 3 a controller is designed to adjust the torso attitude through the motion of the support foot. PD section is involved in the velocity of the foot to servo the torso attitude to the desired value as in Eq. (27). Thus the moving velocity should be

$$ \dot{x} = \dot{x}_d + \dot{x}_\varphi = \dot{x}_d + k_p(\phi - \phi_d) - k_d(\dot\phi - \dot\phi_d) \tag{29} $$

And the foot trajectory of the support phase will change to

$$ \begin{cases} x_s(t) = x_{s0} - \int_0^t (\dot{x}_d + k_p(\phi - \phi_d) - k_d(\dot\phi - \dot\phi_d))\,\mathrm{d}t \\ z_s(t) = z_0 \end{cases} \tag{30} $$

4.2 Flight Phase

Motion trajectory of the flight foot should go through the lift-off point (x_{f0}, z_0), and the touch-down point (x_{fT}, z_0). Obviously the lift-off point is the final status of the last support phase, and the touch-down point will be the initial status of the next support phase.

For a legged robot moving with constant speed $\dot{x}_d$ and support period T_s, the foot trajectory should be sym-metric to the vertical line through the hip. Thus the touch-down point of the flight phase would be $x_{fT} = (\dot{x}_d T_s)/2$. Nevertheless, according to Eq. (29) the moving velocity $\dot{x}$ would hardly be equal to $\dot{x}_d$. Raibert, et al. [2] indicated that the forward speed could be adjusted by using a linear function of the error in forward speed to find a displacement for the foot. Thus the touch-down point should be

$$ x_{fT} = \frac{\dot{x}_d T_s}{2} + k_{\dot{x}}(\dot{x} - \dot{x}_d) \tag{31} $$

In order to minimize the contact forces between ground and the foot, it is generally considered that the foot trajectory should meet the demand that the vertical velocity becomes zero at the time of touch-down, lift-off and maximum foot height.

In addition, it is hoped that there are no steps with the position curve and velocity curve of the foot. According to Eq. (29), the velocity of the support foot

(also velocity of the torso) would not be a constant value and the coordinate of the foot while lifting off would be variable. Thus the foot trajectory of the flight phase should be state based.

The requirements for the flight phase trajectory can be summarized as

$$\begin{cases} x_f(0) = x_{f0}, \dot{x}_f(0) = \dot{x}_{f0}, x_f(T_f) = x_{fT}, \dot{x}_f(T_f) = \dot{x} \\ z_f(0) = z_0, \dot{z}_f(0) = 0, z_f(T_f) = z_0, \dot{z}_f(T_f) = 0 \\ z_f(T_f/2) = z_0 + H_f, \dot{z}_f(T_f/2) = 0 \end{cases} \tag{32}$$

where

x_f and z_f are the coordinates of the flight foot.

H_f indicates the step height.

T_f is the length of the flight phase time. For trotting gait with duty factor $\beta = 0.5$, $T_f = T_s$.

Thus the equations for the flight phase foot trajectory are

$$x_f(t) = \frac{\dot{x}T_f + \dot{x}_{f0}T_f + 2x_{f0} - 2x_{fT}}{T_f^3}t^3 + \frac{3x_{fT} - 3x_{f0} - 2\dot{x}_{f0}T_f - \dot{x}T_f}{T_f^2}t^2 + \dot{x}_{f0}t + x_{f0} \tag{33}$$

$$z_f(t) = \begin{cases} z_0 + H_f \times \left(-\frac{16}{T_f^3}t^3 + \frac{12}{T_f^2}t^2\right) \\ \qquad \text{for } 0 \leqslant t < \frac{T_f}{2} \\ z_0 + H_f \times \left(\frac{16}{T_f^3}t^3 - \frac{36}{T_f^2}t^2 + \frac{24}{T_f}t - 4\right) \\ \qquad \text{for } \frac{T_f}{2} \leqslant t < T_f \end{cases} \tag{34}$$

The next step is to transform the virtual leg model back to the quadruped robot. In the derivation mentioned above trajectories that lie in the $x - z$ plane of the virtual leg model are generated, but this can be easily extended to any orientation and eventually to the quadruped robot. Equations for the foot of the real quadruped robot are given in Appendix A.

4.3 Yaw Control

The control algorithm discussed earlier can drive the robot to move in longitudinal or lateral directions, while yaw control would enable it to spin or make turns.

In our controller, foot placement is used to generate a yaw motion of the robot. As shown in Fig. 7, if the feet are positioned to rotate the line connecting the feet about the center of mass, the robot body will spin around the yaw axis (z_b axis in Fig. 2). The resulting couple is used to manipulate the yaw orientation of the quadruped without disturbing its rectilinear motions. Coordinates for the feet will be

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \begin{bmatrix} \cos(\psi(t)) & -\sin(\psi(t)) & 0 \\ \sin(\psi(t)) & \cos(\psi(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} \tag{35}$$

where

(x_i, y_i, z_i) are the coordinates of foot i in $\{O_b\}$ programmed for the rectilinear motions.

(X_i, Y_i, Z_i) are the modified coordinates of foot i.

$\psi(t)$ is the angle between the line through the hips and the line through the diagonal feet, as shown in Fig. 7.

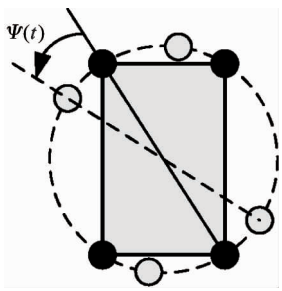


Fig. 7 Control of turning about the yaw axis. The diagram shows the quadruped viewed from above, indicating how the placement of the feet can be used to generate a spinning motion of the robot. The filled circles indicate the location of the hips. Tile open circles indicate the placement of the feet

Similar to the previous part, $\psi(t)$ of the support feet and flight feet are programmed separately with the desired yaw rate. For the support feet:

$$\psi_s(t) = \psi_{s0} - \int_0^t \dot{\psi} \mathrm{d}t \tag{36}$$

And for the flight feet:

$$\psi_f(t) = \frac{2\dot{\psi}T_f - 2\int_0^t \dot{\psi}\mathrm{d}t}{T_f^3}t^3 + \frac{3\int_0^t \dot{\psi}\mathrm{d}t - 3\dot{\psi}T_f}{T_f^2}t^2 + \dot{\psi}t + \psi_{f0} \tag{37}$$

where ψ_{s0} and ψ_{f0} are the initial values of $\psi_s(t)$ and $\psi_f(t)$. $\dot{\psi}$ is the desired yaw rate.

Block diagram of the trotting controller is shown in Fig. 8.

5 Experiments

For the sake of verifying kinematic equations and the performance of the trotting controller, and getting some important parameters, experiments are conducted both in simulation and on the real robot. The mobile robotics simulate software Webots is used for the simulation tests. This section demonstrates successful trotting in simulation and on the real LittleCalf robot with

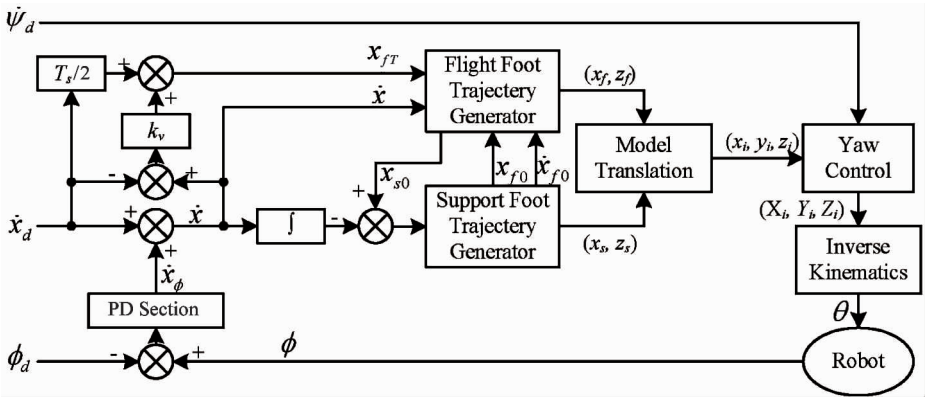


Fig. 8 Block diagram of the trotting controller

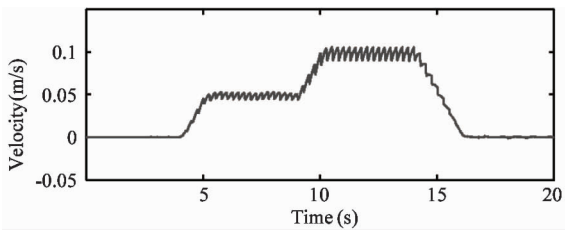
the same controller, while it also shows how the system responds to unexpected perturbations in simulation.

Some parameters used in the simulation and the physical prototype test are listed in Appendix B.

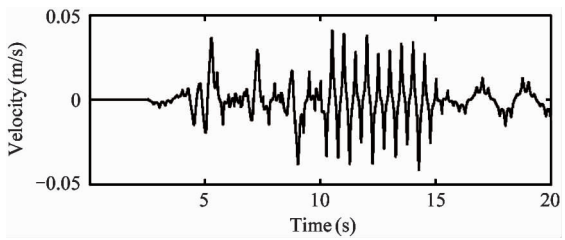
5.1 Trotting in simulation

The initial implementation and testing cycle have been performed in simulation. There it is able to test and tune a number of different controller parameter sets and obtain exhaustive data about the motions.

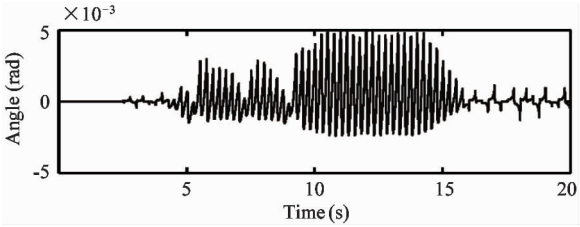
The trotting controller is capable of trotting in place where it maintains the body attitude very close to zero. The desired forward velocity of the robot is ramped up to 0.05m/s and then to 0.1m/s and finally ramp down to zero. In Fig. 10, a run of this procedure is presented. Fig. 9(a) and Fig. 9(b) present the robot velocity while Fig. 9(c) and Fig. 9(d) show the torso attitude throughout this simulation trial. Example snapshots of the robot trotting in simulation are available in Fig. 10. And the leg motions during the 0.1m/s trot in world frame are plotted by stick sequence for two gait cycles in Fig. 11.



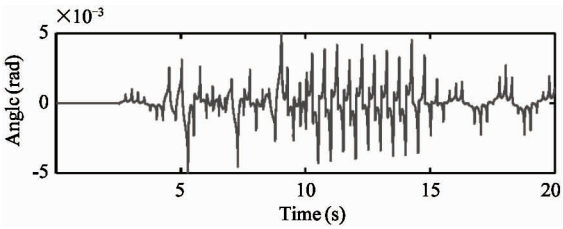
(a) Forward velocity of the robot torso through the simulation run



(b) Lateral velocity of the robot torso through the simulation run



(c) Roll angle of the robot torso through the simulation run



(d) Pitch angle of the robot torso through the simulation run

Fig. 9 Plots of the velocity and attitude of the robot throughout the simulation run. The robot was trotting at place at first and then the desired forward velocity ramped up to 0.05m/s and then to 0.1m/s and finally ramped down to zero

Fig. 10 Snapshots showing the robot trotting in simulation. From left to right, trotting at a forward velocity of 0m/s, at 0.05m/s and 0.1m/s

Moreover, to test the superiority of the controller compared with traditional position-controlled gait planning method, the composite cycloid foot trajectory in Ref. [1] on our robot is applied. Fig. 12 shows the torso attitude of the robot while the robot trotting forward with velocity of 0.1m/s using the composite cycloid foot trajectory in Ref. [1] as well as using the

trotting controller proposed in this paper. Obviously the controller could maintain the body attitude much more close to zero.

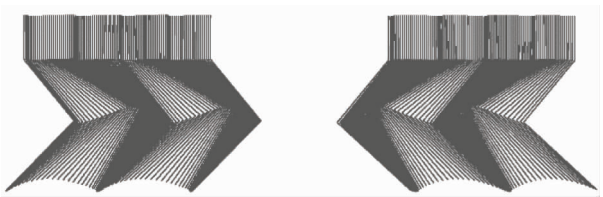
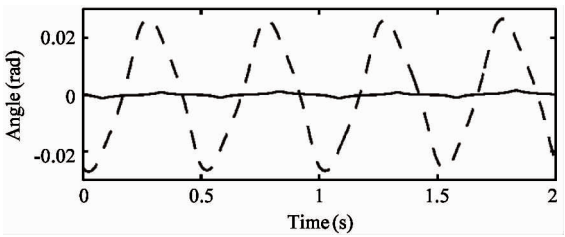
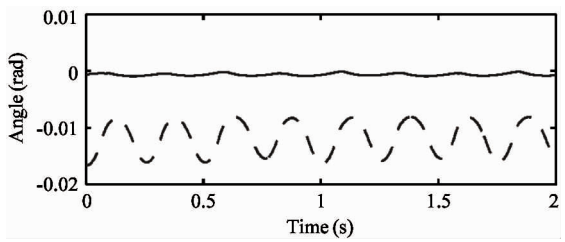


Fig. 11 Stick figure sequence of leg 0 (right) and leg 2 (left) for two gait cycles (from left to right), described in sagittal plane of world frame. The desired forward speed of the robot is 0.1 m/s



(a) Roll angles of the robot torso



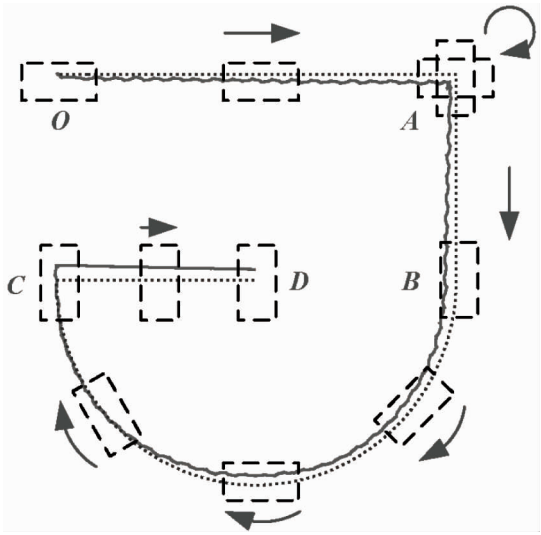
(b) Pitch angles of the robot torso

Fig. 12 Comparison of the torso attitude angles collected in the simulator while the robot trotting with the composite cycloid foot trajectory and with the trotting controller. Dashed lines: trotting with the composite cycloid foot trajectory. Solid line: trotting with the trotting controller. The desired forward velocity are all set to 0.1m/s

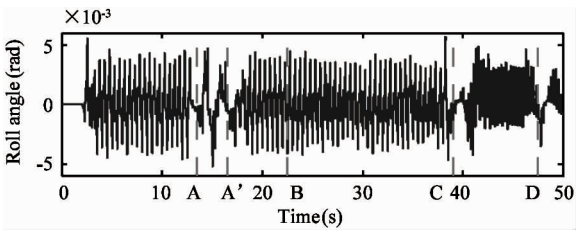
5.2 Path tracking test

If the robot has the ability to actively move forward or sideward, spin or make turns, it can effectively avoid large obstacles in front or track a complex path. In the simulation, the robot is dictated to track a desired path as shown in Fig. 13(a) (dashed line). The starting point is O and the robot moves forward to point A. At A it spins for 90° to right and then move forward to point B. After that, the robot is dictated to make a right turn and draw a semicircle path to point C. Finally it moves right-sideward and arrives at D.

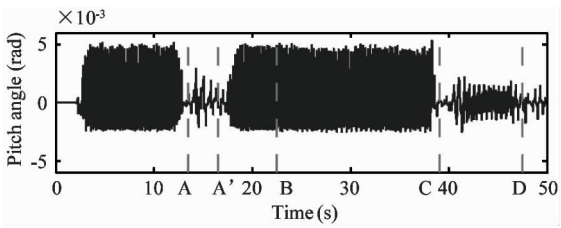
The simulation results for this task are shown in Fig. 13. The solid line in Fig. 13(a) indicates the desired and actual trajectory of the robot's COM. It seems



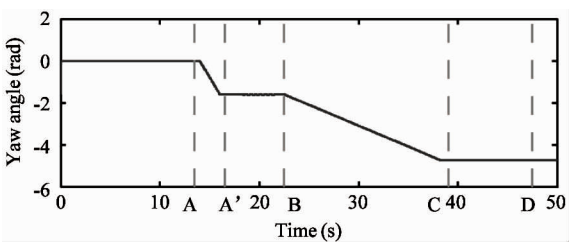
(a) Desired and actual path of the robot



(b) Roll angle of the robot torso



(c) Pitch angle of the robot torso



(d) Yaw angle of the robot torso

Fig. 13 Simulation results of path tracking test. In (a), the solid and dashed lines indicate the actual and desired paths, respectively. (b)(c)(d) shows the torso attitude angles and the vertical dashed lines indicate the moments when the robot arrives at the corresponding points

that the robot tracks the desired path quite well. Fig. 13(b), Fig. 13(c) and Fig. 13(d) show the roll, pitch and yaw angles of the torso during the motion.

The vertical dashed lines labelled A, B, C and D indicate the moments when the robot arrives at point A, B, C and D of the path. And the dates between time A and A' show the spinning motion of the robot at point A. The snapshots of this simulation are exhibited in Fig. 14.

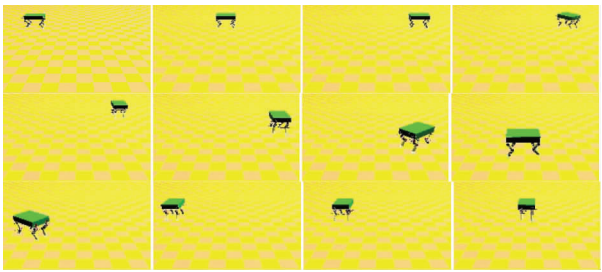


Fig. 14 Snapshots showing the path tracking simulation

5.3 Impact recovery

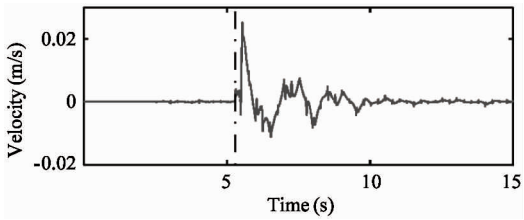
A showcase of the controller robustness is the ability to recover from unexpected perturbations. Due to the morphology of the robot, perturbations along the coronal plane are much harder to accommodate[22]. the response of the controller is tested while trotting in place and while trotting at a constant velocity.

Unexpected impacts are exerted on the broadside of the robot by a pendulum bob in simulation. The impulse acting on the robot's torso is 0.3kg·m/s. When trotting in place and when trotting in the specified velocity range the robot can successfully recover from perturbations. Fig. 15 shows how the controller responds to the impact by presenting the attitude and the velocity of the robot body as it is impacted laterally by the pendulum bob, and how the disturbances are dissipated.

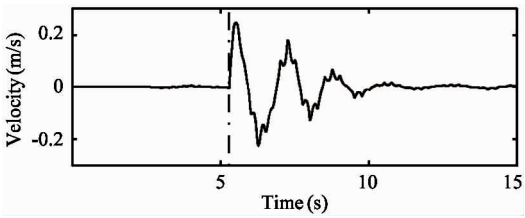
5.4 Test on LittleCalf

The trotting controller is tested on the real quadruped robot with similar success including experiment with the robot trotting in place, trotting forward and backward. Fig. 17 shows a photo of the LittleCalf robot trotting forward with the forward velocity of 0.1m/s. In the same way, the composite cycloid foot trajectory proposed in Ref. [1] is also applied on LittleCalf as comparison. The pitch and roll angles of the torso in these experiments are presented in Fig. 18. And similarly, LittleCalf also demonstrates the ability of omnidirectional moving and the path tracking test is shown in Fig. 19.

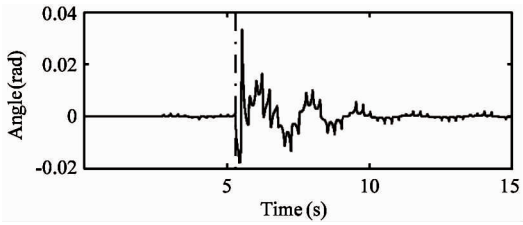
Snapshots from this test are presented in Fig. 16.



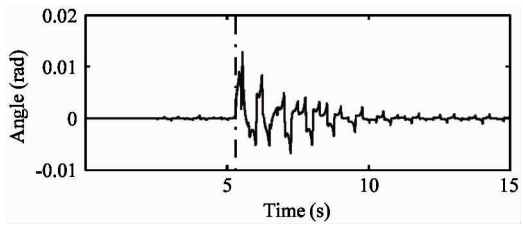
(a) Forward velocity of the robot torso through the lateral impact test



(b) Lateral velocity of the robot torso through the lateral impact test



(c) Roll angle of the robot torso through the lateral impact test



(d) Pitch angle of the robot torso through the lateral impact test

Fig. 15 Plots of the velocity and attitude of the robot collected from the lateral impact test. Vertical dashed line shows the time that the robot torso being impacted laterally by the pendulum bob. The bob is 0.15kg weight and moves at speed of 2m/s before knocking on the robot

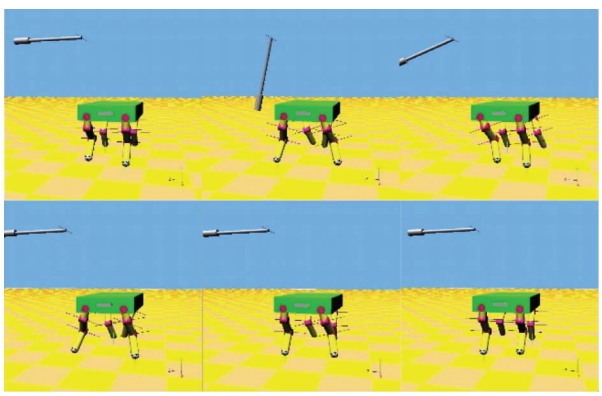


Fig. 16 Snapshots showing the impact recovery of the quadruped robot

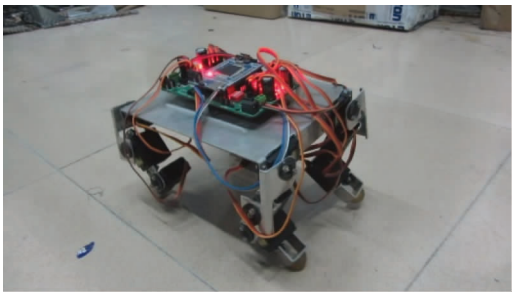
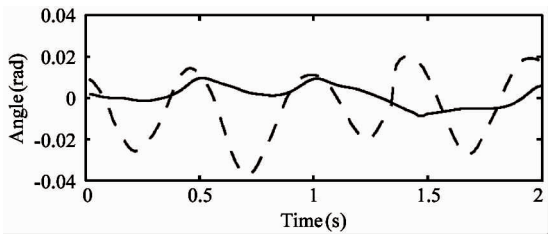
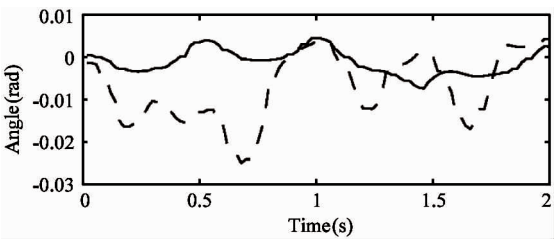


Fig. 17 Photo of the LittleCalf trotting with forward velocity of 0.1m/s



(a) Roll angles of the robot torso



(b) Pitch angles of the robot torso

Fig. 18 Comparison of the torso attitude angles collected through LittleCalf while the robot trotting with the composite cycloid foot trajectory and with the trotting controller. Dashed lines: trotting with the composite cycloid foot trajectory. Solid line: trotting with the trotting controller. The desired forward velocity are all set to 0.1m/s

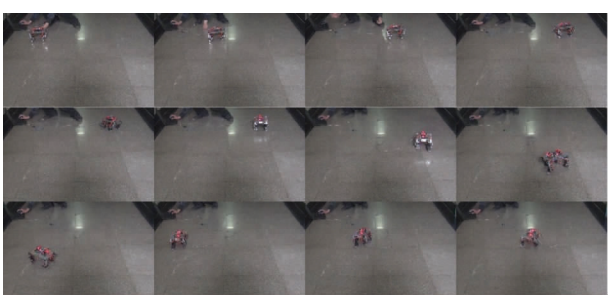


Fig. 19 Snapshots showing the path tracking test on LittleCalf

Objectively speaking, the test done on LittleCalf is less than perfect compared to that in simulation on account of the limitations of the hardware platform. That is the shortcoming of position control which requires high motion precision and accurate sensor values[23]. And in simulation it is possible to decouple the control laws from the limitations of specific hardware platforms. Nonetheless, the controller is still quite effective on LittleCalf and shows superiority in maintaining the torso attitude.

6 Conclusion

Based on the actuation mode and sensors of the robot LittleCalf, this paper elaborates a trotting controller for the position controlled quadruped robot, which utilizes the torso attitude control, moving velocity control and yaw control. It has shown how the controller is organized and how the different tasks are implemented. Main advantage of the controller is that the complex dynamic calculation of the robot and complicated force control are avoided yet the robot could move steadily and stably. How this trotting controller is able to trot at varying speeds and varying directions and how it can robustly dissipate unexpected perturbations in simulation are presented. Moreover, Effectiveness of the trotting controller is also verified by experiments on LittleCalf.

However, every coin has two sides. Shortcomings of our trotting controller are quite obvious, which requires high motion precision and accurate attitude sensor values. And collision between the feet and the ground may be strong since it does not employ force control thus this method may not be suitable to be directly applied to some large robots. Nevertheless, our controller is still quite effective in improving the stability and robustness of quadruped trotting and it is believed it will provide a new idea for the quadruped researchers.

In the future work, the controller should be improved so as to enable the robot to move on different types of terrain. There would be more simulations and experiments.

Appendix A: Correspondence between the virtual leg model and the quadruped robot

If leg i is in support phase, the coordinate of the foot in $\{O_b\}$ should be

$$x_{i,s} = x_{i,s0} - \int_0^t (\dot{x}_d + k_{px}(\phi_p - \phi_{pd}) - k_{dx}(\dot{\phi}_p - \dot{\phi}_{fd}))dt + x_{i,ini} \quad (38)$$

$$y_{i,s} = y_{i,s0} - \int_0^t (\dot{y}_d + k_{py}(\phi_r - \phi_{rd}) - k_{dy}(\dot{\phi}_r - \dot{\phi}_{rd}))dt + y_{i,ini} \quad (39)$$

$$z_{i,s} = z_{i,ini} \quad (40)$$

where $(x_{i,ini}, y_{i,ini}, z_{i,ini})$ indicates the initial point of foot i declared in the last part of Section 2. ϕ_p and ϕ_r

are the pitch and roll angles of the robot torso, ϕ_{pd} and ϕ_{rd} are the desired values.

If leg i is in flight phase, the coordinate of the foot in in $\{O_b\}$ is

$$x_{fT} = \frac{\dot{x}_d T_s}{2} + k_{\dot{x}}(\dot{x} - \dot{x}_d) \tag{41}$$

$$x_{i,f}(t) = \frac{\dot{x}T_f + \dot{x}_{f0}T_f + 2x_{f0} - 2x_{fT}}{T_f^3}t^3 + \frac{3x_{fT} - 3x_{f0} - 2\dot{x}_{f0}T_f - \dot{x}T_f}{T_f^2}t^2 + \dot{x}_{f0}t + x_{f0} + x_{i,ini} \tag{42}$$

$$y_{fT} = \frac{\dot{y}_d T_s}{2} + k_{\dot{x}}(\dot{y} - \dot{y}_d) \tag{43}$$

$$y_{i,f}(t) = \frac{\dot{y}T_f + \dot{y}_{f0}T_f + 2y_{f0} - 2\dot{y}_{fT}}{T_f^3}t^3 + \frac{3y_{fT} - 3y_{f0} - 2\dot{y}_{f0}T_f - \dot{y}T_f}{T_f^2}t^2 + \dot{y}_{f0}t + y_{f0} + y_{i,ini} \tag{44}$$

$$z_{i,f}(t) = \begin{cases} z_{i,ini} + H_f \times \left(-\frac{16}{T_f^3}t^3 + \frac{12}{T_f^2}t^2\right) & \text{for } 0 \leqslant t < \frac{T_f}{2} \\ z_{i,ini} + H_f \times \left(\frac{16}{T_f^3}t^3 - \frac{36}{T_f^2}t^2 + \frac{24}{T_f}t - 4\right) & \text{for } \frac{T_f}{2} \leqslant t < T_f \end{cases} \tag{45}$$

Note that Eqs(38)(39)(40)(42)(44)(45) do not involve the yaw control.

Appendix B: Experiments parameters

Table 4 Control parameters of the simulated model and LittleCalf

Control parameters	Simulated model	LittleCalf
T_s	0.25s	0.25s
T_f	0.25s	0.25s
H_f	10mm	10mm
$\boldsymbol{\Phi}_{pd}$	0	0
$\boldsymbol{\Phi}_{rd}$	0	0
K_{px}	1mm/(rad · s)	1mm/(rad · s)
k_{dx}	0mm/rad	0mm/rad
k_{py}	−15mm/(rad · s)	−20mm/(rad · s)
k_{dy}	−0.01(mm/rad)	−0.05(mm/rad)
$\boldsymbol{k}_{\dot{x}}$	0.1	0.15
$\boldsymbol{k}_{\dot{y}}$	0.05	0.05

Reference

[1] Rong X, Li Y, Ruan J, et al. Design and simulation for a hydraulic actuated quadruped robot. *Journal of mechanical science and technology*, 2012, 26(4): 1171-1177

[2] Raibert M H. Legged robots that balance. Cambridge, MA: MIT press, 1986

[3] Raibert M H. Trotting, pacing and bounding by a quadruped robot. *Journal of biomechanics*, 1990, 23: 79-98

[4] Raibert M, Blankespoor K, Nelson G, et al. Bigdog, the rough-terrain quadruped robot. In: Proceedings of the 17th World Congress The International Federation of Automatic Control, Seoul, Korea, 2008. 10822-10825

[5] Michael K. Meet Boston dynamics' LS3-the latest robotic war machine. *Research Online*, 2012

[6] Xie H, Ahmadi M, Shang J, et al. An intuitive approach for quadruped robot trotting based on virtual model control. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 2015, 229(4): 342-355

[7] Kurazume R, Yoneda K, Hirose S. Feedforward and feedback dynamic trot gait control for quadruped walking vehicle. *Autonomous Robots*, 2002, 12(2): 157-172

[8] Kalakrishnan M, Buchli J, Pastor P, et al. Learning, planning, and control for quadruped locomotion over challenging terrain. *The International Journal of Robotics Research*, 2011, 30(2): 236-258

[9] Fukuoka Y, Kimura H, Cohen A H. Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts. *The International Journal of Robotics Research*, 2003, 22(3-4): 187-202

[10] Barasuol V, Buchli J, Semini C, et al. A reactive controller framework for quadrupedal locomotion on challenging terrain. In: Proceedings of the IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 2013. 2554-2561

[11] Gehring C, Coros S, Hutter M, et al. Control of dynamic gaits for a quadrupedal robot. In: Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 2013. 3287-3292

[12] Yi S. Reliable gait planning and control for miniaturized quadruped robot pet. *Mechatronics*, 2010, 20(4): 485-495

[13] Kim H, Kang T, Loc V G, et al. Gait planning of quadruped walking and climbing robot for locomotion in 3D environment. In: Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005. 2733-2738

[14] Sakakibara Y, Kan K, Hosoda Y, et al. Foot trajectory for a quadruped walking machine. In: Proceedings of the IEEE International Workshop on Intelligent Robots and Systems, IROS '90, Ibaraki, Japan, 1990. 315-322

[15] Dong H, Zhao M, Zhang J, et al. Gait planning of quadruped robot based on third-order spline interpolation. In: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 2006. 5756-5761

[16] Semini C, Tsagarakis N G, Guglielmino E, et al. Design of HyQCa hydraulically and electrically actuated quadruped robot. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 2011: 831-849

[17] Zhang X, Zheng H, Guan X, et al. A biological inspired quadruped robot: structure and control. In: Proceedings of the IEEE International Conference on Robotics and Biomimetics, Hong Kong, China, 2005. 387- 392

[18] Rong X, Li Y, Ruan J, et al. Kinematics analysis and simulation of a quadruped robot. *Applied Mechanics and Materials*, 2010, 26: 517-522

[19] Nanua P, Waldron K J. Energy comparison between trot, bound, and gallop using a simple model. *Journal of biomechanical engineering*, 1995, 117(4): 466-473

[20] Semini C. HyQIDesign and development of a hydraulically actuated quadruped robot: [Ph. D dissertation]. Genoa: University of Genoa, 2010. 45-86

[21] Raibert M, Chepponis M, Brown Jr H B. Running on four legs as though they were one. *IEEE Journal of Robotics and Automation*, 1986, 2(2): 70-82

[22] Havoutis I, Semini C, Buchli J, et al. Quadrupedal trotting with active compliance. In: Proceedings of the 2013 IEEE International Conference on Mechatronics, Vicenza, Italy, 2013. 610-616

[23] Zhang G, Chai H, Rong X, et al. An impact recovery approach for quadruped robot with trotting gait. In: Proceedings of the 2014 IEEE International Conference on Information and Automation, Hailar, China, 2014. 819-824

Zhang Guoteng, born in 1989. He is currently a PhD student in Shandong University, China. He received his B. S degree from Shandong University in 2007. His research interests include robotics, intelligent control, etc.