

# 基于虚拟模型的四足机器人对角小跑步态控制方法

张国腾, 荣学文, 李贻斌, 柴 汇, 李 彬

(山东大学控制科学与工程学院, 山东 济南 250061)

**摘 要:** 为提高四足机器人对角小跑运动的稳定性, 实现机器人躯干 6 维运动方向控制的解耦, 提出了一种基于虚拟模型的对角小跑步态控制方法. 控制器主要包括支撑相虚拟模型控制和摆动相虚拟模型控制. 在支撑相, 建立了作用于躯干质心的虚拟力与对角支撑腿关节扭矩之间的数学关系, 通过调整躯干虚拟力的大小控制躯干的高度与姿态, 控制机器人前进速度和自转角速度. 在摆动相, 将机器人侧向速度控制引入到足端轨迹规划中, 并通过虚拟的“弹簧—阻尼”元件驱动摆动足沿给定轨迹运动. 此外, 在控制器设计过程中, 引入了状态机, 用于监控机器人各腿的状态, 并输出对角小跑步态相位切换指令. 仿真实验结果表明, 机器人能够以对角小跑步态在平地上进行全方位移动, 跨越不平坦地形, 并能够抵抗外部冲击, 证明了文中控制方法的有效性和鲁棒性.

**关键词:** 四足机器人; 对角小跑步态; 虚拟模型控制

**中图分类号:** TP242

**文献标识码:** A

**文章编号:** 1002-0446(2016)-01-0064-11

## Control of the Quadrupedal Trotting Based on Virtual Model

ZHANG Guoteng, RONG Xuewen, LI Yibin, CHAI Hui, LI Bin

(School of Control Science and Engineering, Shandong University, Ji'nan 250061, China)

**Abstract:** In order to improve the stability of the trotting quadruped robot and to decouple the control of the robot torso motion along six directions, an approach based on virtual model is presented for trot gait control. The controller mainly consists of two main modules: the virtual model control at support phase and the virtual model control at flight phase. During the support phase, the mathematical relationship are mapped between the joint torques of diagonal support legs and the virtual forces acted on the center-of-mass of the torso. And the values of virtual torso forces are regulated to control the torso attitude and height, as well as the forward velocity and the yaw angular velocity of the robot. During the flight phase, lateral velocity is introduced to plan the toe trajectory. And virtual spring-damper sections are implemented to drive the flight toes to track the planned trajectories. In addition, while designing the controller, a state machine is introduced to monitor the legs' states and output phase switching commands for trot gait regulation. The simulations show that the robot is able to trot omnidirectionally on flat ground as well as uneven terrains, even suffering from external impacts. And thus the effectiveness and robustness of the controller are proved.

**Keywords:** quadruped robot; trot gait; virtual model control

## 1 引言 (Introduction)

腿足式机器人是移动机器人领域的一个重要分支, 与常见的轮式或履带式移动机器人相比, 腿足式机器人具有更好的移动灵活性和地形适应能力. 其中四足机器人相对于单足或双足机器人运动更加稳定, 与六足或更多足的机器人相比结构又更加简单, 因此四足机器人成为机器人界的研究热点.

自然界中的四足动物在运动过程中会根据运动目的和地形信息采用不同的步态, 其中对角小跑 (trot) 是一种处于对角线上 2 只脚同时运行的步

态, 这种步态在相当大的运动速度范围内能保证比较高的能量效率, 而且每一步之间躯干姿态并没有特别明显的起伏, 因此对角小跑步态是自然界中最为常见的步态<sup>[1]</sup>, Trot 步态在四足机器人上的应用与控制也成为许多学者们的研究目标.

仿生机器人的控制常常从生物界中寻找灵感, 其中较为具有代表性的是 CPG (central pattern generator) 算法. Kimura 等<sup>[2]</sup>提出了一种基于 CPG 以及响应和反射机制的神经网络控制方法, 并成功地控制 Tekken 机器人以对角小跑步态在崎岖地形上行

基金项目: 国家863计划 (2015AA042201); 国家自然科学基金 (61233014, 61305130); 山东省自然科学基金 (ZR2013FQ003, ZR2013EEM027); 中国博士后科学基金 (2013M541912).

通信作者: 荣学文, rongxw@sdu.edu.cn 收稿/录用/修回: 2015-08-12/2015-10-20/2015-10-29

走。Matos 和 Santos<sup>[3]</sup> 采用 CPG 方法实现了四足机器人的全方位移动，并在 AIBO 四足机器人上进行了相应的实验验证。意大利的 HyQ 机器人在对角小跑步态控制中采用了基于 CPG 的足端轨迹生成方法<sup>[4]</sup>，机器人展示了良好的运动稳定性和越障能力。总之，CPG 算法对于控制四足机器人运动，尤其是复杂地形下的对角小跑步态运动具有较好的效果，然而美中不足的是，该算法较为复杂，且涉及到多种参数关联机制<sup>[5]</sup>。

基于 ZMP（零力矩点）的算法作为双足机器人常用的控制方法，同样被引入到了四足机器人对角小跑步态控制中。Kurazume 等人<sup>[6]</sup> 基于 ZMP 方法提出了躯干的 3 维摆动曲线，并控制 TITAN-VIII 机器人实现了稳定的对角小跑运动。Zhang 等<sup>[7]</sup> 使用 ZMP 稳定判据对足端轨迹进行规划，并分别在仿真环境与物理样机上实现了机器人的对角小跑步态运动。但相对而言 ZMP 方法更适合用于四足机器人静步态的控制，用于对角小跑步态控制会导致机器人运动速度偏慢，且抗扰性较差。

当今最著名的腿足式机器人控制方法当属 Raibert 提出的“三段式”控制理论<sup>[8-9]</sup>，该理论将多足运动等效为单足跳跃模型，并分别控制其前进速度、躯干姿态和弹跳高度。当今最引人瞩目的四足机器人当属波士顿动力公司开发的 BigDog<sup>[10]</sup> 和 LS3<sup>[11]</sup>。虽然未见到具体的介绍控制方法的文献，但研究者们普遍认为其控制是基于 Raibert 的“三段式”方法的。除此之外，另一种较为直接的腿足式机器人控制方法是 Pratt 提出的虚拟模型控制<sup>[12-14]</sup>，通过向机器人某部位施加虚拟力的方法控制机器人运动。一些著名的机器人如 HyQ<sup>[15]</sup>、StarLETH<sup>[16]</sup> 等均在其对角小跑步态运动中采用了虚拟模型控制。遗憾的是，上述研究主要针对机器人的单腿模型或平面模型，对四足机器人的整体 3 维模型的建立和控制研究较少。鄂明成等<sup>[17]</sup> 建立了四足机器人整体动力学模型，将机器人躯干虚拟力矢量分配到各支撑足，采用间歇型对角小跑步态<sup>[18]</sup> 实现了机器人粗糙地形稳定连续行走，但该方法并未实现机器人全方向运动解耦，且需要引入四足支撑相保证运动稳定。

针对上述问题，本文提出了一种简单有效且稳定性较强的四足机器人对角小跑步态控制方法。在支撑相阶段，构建机器人整体 3 维受力模型，推导机器人支撑腿输出力与作用在躯干质心上的虚拟力的映射关系，由躯干运动需求求解出所需虚拟力的大小，并计算出对应支撑腿各关节输出扭矩。在摆

动相阶段，利用作用在摆动腿足端的虚拟元件驱动摆动足沿设定的轨迹运动，并通过落足点的选择控制机器人的侧向运动速度。支撑相与摆动相之间通过状态机进行切换。该方法的有效性与鲁棒性通过在移动机器人仿真软件 Webots 中进行的仿真实验进行验证。

2 四足机器人模型（Model for the quadruped robot）

采用的四足机器人模型基于山东大学研制的液压驱动四足仿生平台 SCalF<sup>[19-20]</sup>，如图 1 所示，该模型包括 1 个躯干与 4 条腿，每条腿含有 1 个横滚髋关节、1 个俯仰髋关节和 1 个俯仰膝关节。模型

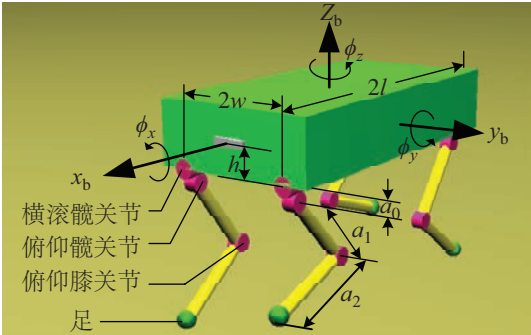


图 1 四足机器人模型

Fig.1 Model for the quadruped robot

表 1 四足机器人模型主要参数

Tab.1 Major parameters of the quadruped model

参数	定义	数值
$l$	前后横滚髋关节距离的一半	0.45 m
$w$	左右横滚髋关节距离的一半	0.175 m
$h$	躯干质心与横滚髋关节垂直距离	0.1 m
$a_0$	髋部连杆长度	0.05 m
$a_1$	大腿连杆长度	0.3 m
$a_2$	小腿连杆长度	0.3 m
$m_0$	髋部连杆质量	2 kg
$m_1$	大腿连杆质量	4 kg
$m_2$	小腿连杆质量	4 kg
$M$	躯干质量	100 kg
$\theta_0$	横滚髋关节转角	/
$\theta_1$	俯仰髋关节转角	/
$\theta_2$	俯仰膝关节转角	/
$\tau_0$	横滚髋关节扭矩	/
$\tau_1$	俯仰髋关节扭矩	/
$\tau_2$	俯仰膝关节扭矩	/
$\phi_x$	躯干横滚角	/
$\phi_y$	躯干俯仰角	/
$\phi_z$	躯干偏航角	/

主要参数列于表 1 中, 躯干与腿部的各连杆均视为刚体, 其质心分别位于各刚体的几何中心上。

在机器人躯干的几何中心上建立躯干坐标系  $\{O_b\}$ , 其  $z_b$  轴垂直于躯干平面指向上方,  $x_b$  轴垂直于  $z_b$  轴并指向机器人正前方,  $y_b$  轴的方向通过右手定则给出, 相应的足端在  $\{O_b\}$  中的坐标  $(x, y, z)$  可以表示为

$$\begin{cases} x = -a_1 s_1 - a_2 s_{12} + \delta l \\ y = a_0 s_0 + a_1 s_0 c_1 + a_2 s_0 c_{12} + \lambda w \\ z = -a_0 c_0 - a_1 c_0 c_1 - a_2 c_0 c_{12} - h \end{cases} \quad (1)$$

其中  $c_k = \cos \theta_k$ ,  $s_k = \sin \theta_k$ ,  $s_{12} = \sin(\theta_1 + \theta_2)$ ,  $c_{12} = \cos(\theta_1 + \theta_2)$ .  $\delta$  和  $\lambda$  为符号变量, 定义如下:

$$\delta = \begin{cases} 1, & \text{左前腿或右前腿} \\ -1, & \text{左后腿或右后腿} \end{cases} \quad (2)$$

$$\lambda = \begin{cases} 1, & \text{左前腿或左后腿} \\ -1, & \text{右前腿或右后腿} \end{cases} \quad (3)$$

对式 (1) 求偏微分可得到相应的雅可比矩阵

$$\mathbf{J} = \begin{bmatrix} 0 & -a_0 c_1 - a_2 c_{12} & -a_2 c_{12} \\ c_0(a_0 + a_1 c_1 + a_2 c_{12}) - s_0(a_1 s_1 + a_2 s_{12}) & -a_2 s_0 s_{12} \\ s_0(a_0 + a_1 c_1 + a_2 c_{12}) & c_0(a_1 s_1 + a_2 s_{12}) & a_2 c_0 s_{12} \end{bmatrix} \quad (4)$$

通过该雅可比矩阵的转置, 已知足端虚拟输出力  $\mathbf{f} = [f_x, f_y, f_z]^T$ , 可以求解出各关节的输出扭矩  $\boldsymbol{\tau} = [\tau_0, \tau_1, \tau_2]^T$ :

$$\boldsymbol{\tau} = \mathbf{J}^T \mathbf{f} \quad (5)$$

由于除某些特殊位姿外该雅可比矩阵满秩, 因此机器人足端 3 个方向上的输出力均为可控的。

### 3 支撑相虚拟模型控制 (Virtual model control at support phase)

#### 3.1 躯干—支撑腿模型

假定机器人腿部质量相对于躯干质量可以忽略不计, 机器人质心集中于躯干几何中心上. 在机器人以对角小跑步态行走过程中, 将机器人看作一个整体并对其进行静力分析, 机器人的前后 2 个支撑足分别受到来自地面的支持力  $\mathbf{f}_F$  和  $\mathbf{f}_H$ , 躯干中心受到垂直向下的重力  $\mathbf{G}$ . 将这 3 个力分别沿  $x_b$ 、 $y_b$  和  $z_b$  轴进行分解, 如图 2 所示, 并由此计算出作用在躯干质心上的虚拟力的大小:

$$F_x = f_{F,x} + f_{H,x} + G_x \quad (6)$$

$$F_y = f_{F,y} + f_{H,y} + G_y \quad (7)$$

$$F_z = f_{F,z} + f_{H,z} + G_z \quad (8)$$

$$T_x = f_{F,z} y_F + f_{H,z} y_H - f_{F,y} z_F - f_{F,y} z_H \quad (9)$$

$$T_y = -f_{F,z} x_F - f_{H,z} x_H + f_{F,x} z_F + f_{H,x} z_H \quad (10)$$

$$T_z = -f_{F,x} y_F - f_{H,x} y_H + f_{F,y} x_F + f_{H,y} x_H \quad (11)$$

其中  $(x_F, y_F, z_F)$  和  $(x_H, y_H, z_H)$  分别为前支撑足与后支撑足在  $\{O_b\}$  坐标系下的坐标。

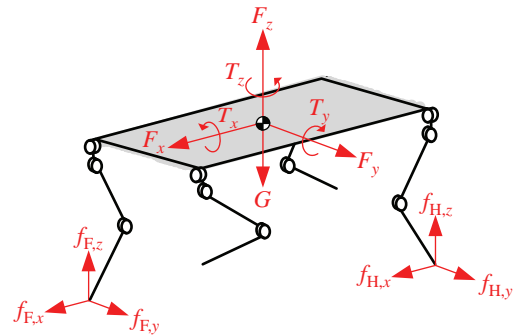


图 2 机器人静态受力图

Fig.2 Static forces acting on the robot

在机器人运动过程中, 一般期望躯干的横滚角  $\phi_x$  为 0, 因此重力在  $y_b$  方向上的分量趋近于 0,  $G_x$ 、 $G_y$  和  $G_z$  的大小可以近似为

$$G_x = Mg \sin \phi_y \quad (12)$$

$$G_y = 0 \quad (13)$$

$$G_z = -Mg \cos \phi_y \quad (14)$$

其中  $\phi_y$  为机器人躯干俯仰角,  $g$  为重力加速度值。

机器人躯干运动控制可以通过控制作用在躯干上的虚拟力的大小来实现. 控制躯干上的虚拟力首先要求得各支撑足所需支撑力的大小, 将式 (6) ~ (11) 改写为矩阵形式, 并结合式 (12) ~ (14), 可以得到

$$\begin{bmatrix} F_x - Mg \sin \phi_y \\ F_y \\ F_z + Mg \cos \phi_y \\ T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & -z_F y_F & 0 & -z_H y_H & 0 & 0 \\ z_F & 0 & -x_F & z_H & 0 & -x_H \\ -y_F & x_F & 0 & -y_H & x_H & 0 \end{bmatrix} \begin{bmatrix} f_{F,x} \\ f_{F,y} \\ f_{F,z} \\ f_{H,x} \\ f_{H,y} \\ f_{H,z} \end{bmatrix} \quad (15)$$

式 (15) 中的  $6 \times 6$  矩阵行列式为 0, 说明该矩阵是不满秩的, 这意味着作用在躯干质心上的虚拟力并非在所有方向上均可控, 因此对躯干某一方向上的运动控制必须被舍弃。

在自然界中, 当某个四足动物希望到达其侧方的某个位置时, 一般会旋转躯干使自己朝向目标前进, 很少采用直接侧向移动的方式. 因此本文在支撑相模型中舍弃对躯干侧向虚拟力  $F_y$  的控制, 躯干的侧向速度通过摆动腿的运动进行控制 (见本文 4.1 节). 给定躯干虚拟力  $[F_x, F_z, T_x, T_y, T_z]^T$ , 求解地面对支撑足的支持力  $\mathbf{f} = [f_{F,x}, f_{F,y}, f_{F,z}, f_{H,x}, f_{H,y}, f_{H,z}]^T$ , 共有 6 个未知数, 而只有 5 个约束条件, 因此人为添加 1 个约束:

$$f_{F,y} = f_{H,y} \quad (16)$$

联立式 (6)、(8) ~ (14) 及 (16) 并写为矩阵形式, 得到

$$\begin{bmatrix} F_x - Mg \sin \phi_y \\ F_z + Mg \cos \phi_y \\ T_x \\ T_y \\ T_z \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & -z_F & y_F & 0 & -z_H & y_H \\ z_F & 0 & -x_F & z_H & 0 & -x_H \\ -y_F & x_F & 0 & -y_H & x_H & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} f_{F,x} \\ f_{F,y} \\ f_{F,z} \\ f_{H,x} \\ f_{H,y} \\ f_{H,z} \end{bmatrix} \quad (17)$$

记式 (17) 中的  $6 \times 6$  矩阵为  $\mathbf{Q}$ , 其行列式的值为  $2(y_F - y_H)(x_H z_F - x_F z_H)$ , 通常情况下该行列式不为 0, 因此可以通过求  $\mathbf{Q}$  的逆来求解各足受到的支持力的大小:

$$\begin{bmatrix} f_{F,x} \\ f_{F,y} \\ f_{F,z} \\ f_{H,x} \\ f_{H,y} \\ f_{H,z} \end{bmatrix} = \mathbf{Q}^{-1} \begin{bmatrix} F_x - Mg \sin \phi_y \\ F_z + Mg \cos \phi_y \\ T_x \\ T_y \\ T_z \\ 0 \end{bmatrix} \quad (18)$$

式中  $\mathbf{Q}^{-1}$  的具体结果请参见附录.

根据作用力与反作用力, 机器人支撑足的输出力与地面对支撑足的支持力大小相等、方向相反. 由式 (4) 分别求得前后支撑腿的雅可比矩阵  $\mathbf{J}_F$  和  $\mathbf{J}_H$ , 要产生期望的躯干虚拟力, 所需前后支撑腿各关节输出扭矩为

$$\begin{bmatrix} \tau_{F,0} \\ \tau_{F,1} \\ \tau_{F,2} \\ \tau_{H,0} \\ \tau_{H,1} \\ \tau_{H,2} \end{bmatrix} = - \begin{bmatrix} \mathbf{J}_F^T & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_H^T \end{bmatrix} \mathbf{Q}^{-1} \begin{bmatrix} F_x - Mg \sin \phi_y \\ F_z + Mg \cos \phi_y \\ T_x \\ T_y \\ T_z \\ 0 \end{bmatrix} \quad (19)$$

### 3.2 躯干运动控制

在机器人以对角小跑步态行走过程中, 需要调整躯干的姿态角和高度, 并控制前进速度和自转角速度. 在本文中, 期望机器人在  $y_b-z_b$  平面上躯干始终处于水平状态, 即横滚角  $\phi_x$  为 0; 而在  $x_b-z_b$  平面上, 躯干始终平行于地面. 为此, 引入“伪俯仰角”概念, 将  $x_b-z_b$  平面上躯干与前后支撑脚的连线之间的夹角记为伪俯仰角  $\psi_y$ , 其求解为

$$\psi_y = \arctan \frac{z_F - z_H}{x_F - x_H} \quad (20)$$

引入伪俯仰角的概念后, 只需伪俯仰角的值为 0, 在  $x_b-z_b$  平面上机器人躯干便会平行于前后脚连线.

躯干高度定义为躯干质心与 2 个支撑脚之间沿  $z_b$  方向的距离, 其求解为

$$h = -\frac{z_F + z_H}{2} \quad (21)$$

机器人躯干的姿态角与高度控制通过在躯干质心处安装虚拟的“弹簧-阻尼”环节进行调节, 即

$$T_x = K_{\phi_x} \phi_x + K_{\dot{\phi}_x} \dot{\phi}_x \quad (22)$$

$$T_y = K_{\psi_y} \psi_y + K_{\dot{\psi}_y} \dot{\psi}_y \quad (23)$$

$$F_z = K_h(h - h_d) + K_{\dot{h}} \dot{h} \quad (24)$$

图 3 给出了机器人矢状面的示意图, 从中可以看出躯干伪俯仰角与高度的获得方法和控制模型.

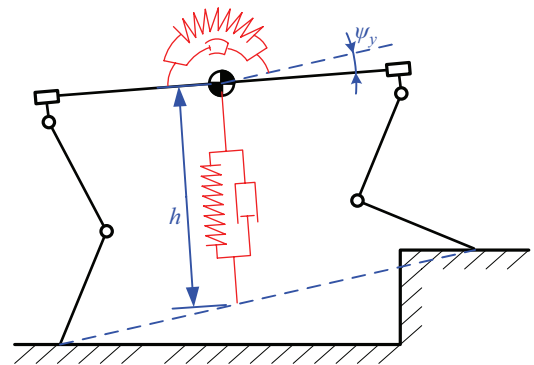


图 3 躯干俯仰角与高度控制示意图

Fig.3 Strategy for the pitch and height control of the torso

而机器人前进速度  $v_x$  和自转角速度  $\omega_z$  的误差通过产生相应的虚拟力进行补偿

$$F_x = K_{v_x}(v_x - v_{xd}) \quad (25)$$

$$T_z = K_{\omega_z}(\omega_z - \omega_{zd}) \quad (26)$$

上述各式中的  $K$  代表增益值, 下标  $d$  表示该变量的期望值. 通过式 (19) 和式 (22)~(26), 便可求出控制机器人躯干运动所需支撑足各关节扭矩的大小.

## 4 摆动相虚拟模型控制 (Virtual model control at flight phase)

### 4.1 摆动足运动轨迹规划

摆动足的运动轨迹规划主要考虑 4 个要素: 摆动周期、最高点高度、起始点、落足点. 对于占空比  $\beta = 0.5$  的对角小跑步态, 一般设定其摆动周期  $T_f$  与支撑周期  $T_s$  相等; 最高点高度  $z_h$  即为足端摆动到最高点时的纵坐标, 其值可根据地面信息进行修改, 本文中将其设为恒值; 摆动轨迹的起始点即为离地点, 可根据支撑相结束时足端位置获得.

Raibert<sup>[8]</sup> 曾经证明, 摆动足落足点的选择可以间接控制机器人躯干的运动速度. 为弥补支撑相模型不能够控制躯干侧向运动的缺陷, 摆动足的落足点的侧向坐标设定为

$$y_T = \frac{v_y T_s}{2} + k_{vy}(v_y - v_{yd}) \quad (27)$$

其中  $v_y$  为机器人实际侧向运动速度,  $v_{yd}$  为期望侧向运动速度,  $k_{vy}$  为比例系数.

由于机器人的前向速度和自转角速度均通过支撑相进行控制, 因此落足点的选择并不受前进速度和自转角速度的限制, 理论上落足点在  $x$  方向上位置可以自由规划, 当机器人在较为崎岖的地面行走时, 可以根据地面信息选择合适的落足位置, 同时不影响对躯干的控制. 本文并未涉及到可落足点规划的内容, 因此在  $x_b$  方向上的落足点位置简单地设定为

$$x_T = \frac{v_{xd} T_s}{2} \quad (28)$$

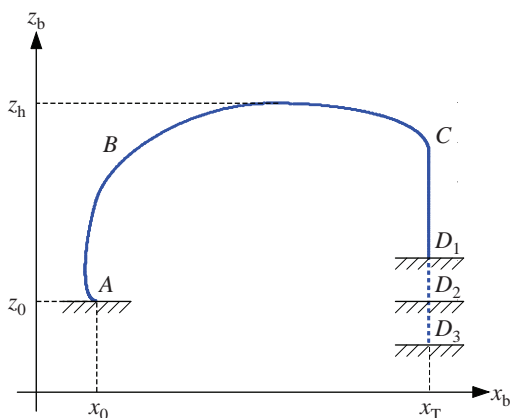


图 4  $x_b$ - $z_b$  平面内足端运动轨迹

Fig.4 The foot trajectory generated in the  $x_b$ - $z_b$  plane

有了摆动周期、摆动高度、起始点和落足点这 4 个要素, 可以设计出成千上万种满足要求的摆动足运动轨迹. 本文采用一种位置曲线和速度曲线连续的足端轨迹, 以  $x_b$ - $z_b$  平面内为例, 足端运动轨

迹如图 4 所示. 从图中的 A 点至 B 点, 摆动足抬起并带有微微后收, 以防止碰到障碍物; 从 B 点至 C 点, 足端向前摆动至期望的落足点水平坐标; 自 C 点以后, 摆动足垂直下落, 直至检测到足端接触到地面为止, 图中  $D_1$ 、 $D_2$ 、 $D_3$  给出了 3 种可能的触地情况. 该足端运动轨迹的数学表达式为

$$x_f(t) = \begin{cases} -\frac{4\dot{x}_0}{T_f}t^2 + \dot{x}_0 t + x_0, & 0 \leq t < \frac{T_f}{4} \\ \frac{-4T_f\dot{x}_0 - 16x_T + 16x_0}{T_f^3}t^3 + \frac{7T_f\dot{x}_0 + 24x_T - 24x_0}{T_f^2}t^2 + \frac{-15T_f\dot{x}_0 - 36x_T + 36x_0}{4T_f}t + \frac{9T_f\dot{x}_0 + 16x_T}{16}, & \frac{T_f}{4} \leq t < \frac{3T_f}{4} \\ x = x_T, & t \geq \frac{3T_f}{4} \end{cases} \quad (29)$$

$$z_f(t) = \begin{cases} \frac{16z_0 - 16z_h}{T_f^3}t^3 + \frac{12z_h - 12z_0}{T_f^2}t^2 + z_0, & 0 \leq t < \frac{T_f}{2} \\ \frac{4z_0 - 4z_h}{T_f^2}t^2 - \frac{4z_0 - 4z_h}{T_f}t + z_0, & t \geq \frac{T_f}{2} \end{cases} \quad (30)$$

其中  $t$  为时间变量,  $(x_0, z_0)$  为摆动轨迹起始点坐标,  $\dot{x}_0$  为摆动足的初始速度, 即支撑相结束时摆动足沿  $x_b$  轴的瞬时速度.  $x_b$ - $z_b$  平面内的足端轨迹规划与此类似.

### 4.2 摆动腿运动控制

摆动腿运动控制通过在摆动足与期望足端位置之间增加虚拟的“弹簧—阻尼”模型实现, 当期望足端位置沿给定的足端轨迹随时间不断变化时, 该模型会拉动摆动足跟随足端轨迹运动. 前后摆动足虚拟模型的数学表达式为

$$f_{F,x} = k_x(x_F - x_{Fd}) + k_{\dot{x}}(\dot{x}_F - \dot{x}_{Fd}) \quad (31)$$

$$f_{F,y} = k_y(y_F - y_{Fd}) + k_{\dot{y}}(\dot{y}_F - \dot{y}_{Fd}) \quad (32)$$

$$f_{F,z} = k_z(z_F - z_{Fd}) + k_{\dot{z}}(\dot{z}_F - \dot{z}_{Fd}) \quad (33)$$

$$f_{H,x} = k_x(x_H - x_{Hd}) + k_{\dot{x}}(\dot{x}_H - \dot{x}_{Hd}) \quad (34)$$

$$f_{H,y} = k_y(y_H - y_{Hd}) + k_{\dot{y}}(\dot{y}_H - \dot{y}_{Hd}) \quad (35)$$

$$f_{H,z} = k_z(z_H - z_{Hd}) + k_{\dot{z}}(\dot{z}_H - \dot{z}_{Hd}) \quad (36)$$

前后摆动腿各关节的扭矩由相应的雅可比矩阵求得



$$\begin{bmatrix} \tau_{F,0} \\ \tau_{F,1} \\ \tau_{F,2} \\ \tau_{H,0} \\ \tau_{H,1} \\ \tau_{H,2} \end{bmatrix} = \begin{bmatrix} \mathbf{J}_F^T & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_H^T \end{bmatrix} \begin{bmatrix} f_{F,x} \\ f_{F,y} \\ f_{F,z} \\ f_{H,x} \\ f_{H,y} \\ f_{H,z} \end{bmatrix} \quad (37)$$

## 5 Trot 步态控制器设计 (Design of the trotting controller)

四足机器人对角小跑步态控制框图如图 5 所示, 主要包括支撑相虚拟模型控制、摆动相虚拟模型控制和状态机 3 部分. 其中  $\mathbf{K}_p$ 、 $\mathbf{K}_d$  分别表示式 (22)~(24) 中的弹性系数和阻尼系数;  $\mathbf{K}_v$  表示式 (25)、(26) 中的速度增益;  $k_p$  和  $k_d$  分别表示式 (31)~(36) 中的弹性系数和阻尼系数.

支撑相虚拟模型用于控制机器人躯干的运动. 根据躯干的姿态、高度与速度的偏差, 计算出作用在躯干质心上的虚拟力向量  $\mathbf{F}$ , 补偿躯干重力后求解出所需地面支持力  $\mathbf{f}$ , 由作用力与反作用力得到支撑腿足端输出力的大小  $\mathbf{f}_s$ , 通过腿部雅可比矩阵的转置求解出支撑腿各关节输出扭矩的大小  $\tau_s$ .

摆动相虚拟模型用于控制机器人摆动腿的运动. 给定离地点向量  $\mathbf{p}_0$ 、最高点  $z_h$  和摆动周期  $T_f$ , 根据运动速度求出落足点位置, 由此规划出足端运动轨迹, 获得期望的足端位置  $\mathbf{p}_d$ , 并通过在足端虚拟“弹簧-阻尼”系统获得摆动腿足端输出力向量  $\mathbf{f}_f$ , 通过腿部雅可比矩阵的转置求解出摆动腿各关节输出扭矩的大小  $\tau_f$ .

当机器人以对角小跑步态行走时, 每条腿的运动状态都会在支撑相和摆动相之间切换. 如果对角小跑步态占空比  $\beta = 0.5$ , 理论上处于对角线上的 2 条腿将会同时抬起并同时落地. 然而一旦机器人躯干发生倾斜或地面不平坦, 则很难保证 2 个摆动足同时触地, 也无法保证摆动足按时触地, 因此无法仅仅根据时间来控制腿部的相位切换. 本文在对角小跑步态控制器中引入基于摆动足摆动时间和触地状态的相位切换机制, 即为状态机.

首先, 给定机器人行走的步态周期  $T$ , 并令  $T_f = T_s = T/2$ . 初始情况下, 机器人 4 条腿均站立于地面, 接着状态机控制处于某一对角线上的 2 条腿 (记为对角腿 A) 抬起作为摆动腿, 控制另外 2 条腿 (记为对角腿 B) 作为支撑腿, 采用摆动相虚拟模型控制对角腿 A 的运动, 采用支撑相虚拟模型驱动对角腿 B 的各关节. 当运行时间  $t > 3/4T_f$  时,

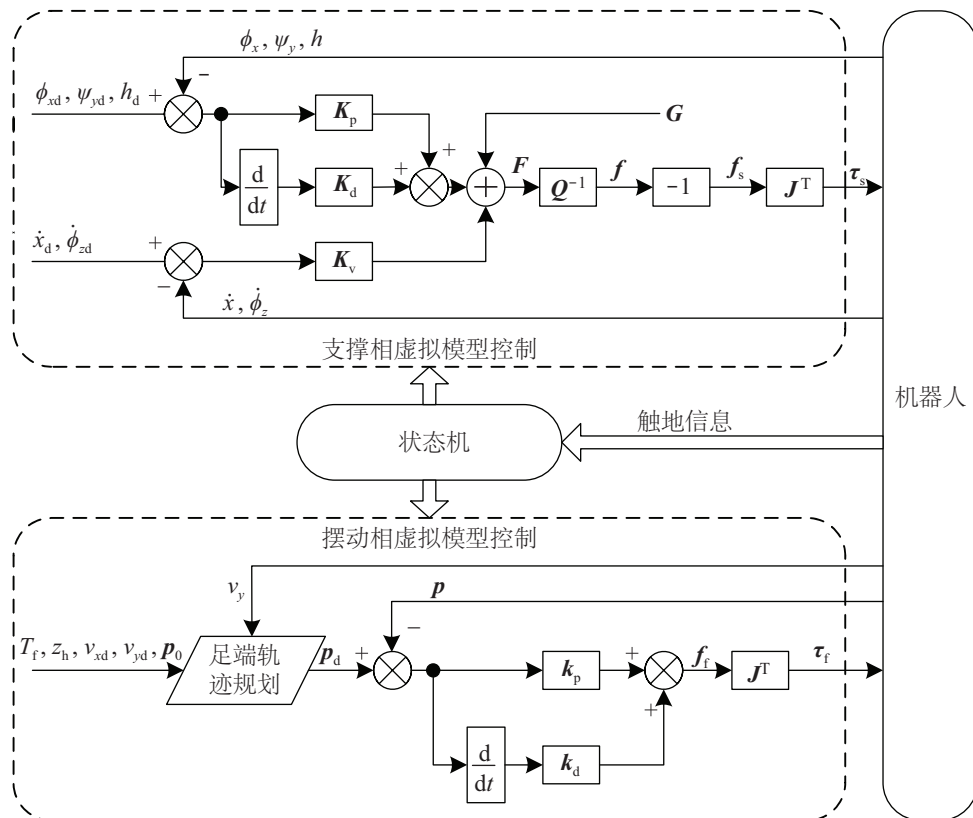


图 5 Trot 步态控制器框图  
Fig.5 Block diagram of the trotting controller

对角腿 A 各自的摆动足均已经到达规划的落足点水平坐标并垂直下落（见本文 4.1 节），此时通过足底传感器或关节扭矩反馈监测腿部触地信息，一旦有 1 个摆动足触地，该足立即停止下落并保持当前位置不变，此时对角腿 B 继续由支撑相虚拟模型进行控制，机器人处于一个短暂的 3 足支撑状态。当检测到另外 1 个摆动足的触地信号后，状态机将运行时间  $t$  清零，并控制各腿进行相位切换，对角腿 A 转换为支撑腿控制躯干运动，对角腿 B 转换为摆动腿跟随足端轨迹，直至  $t > 3/4T_f$  且对角腿 B 的 2 只足均检测到触地信息。此时状态机控制各腿再一次进行相位切换，依此类推。整个流程如图 6 所示。

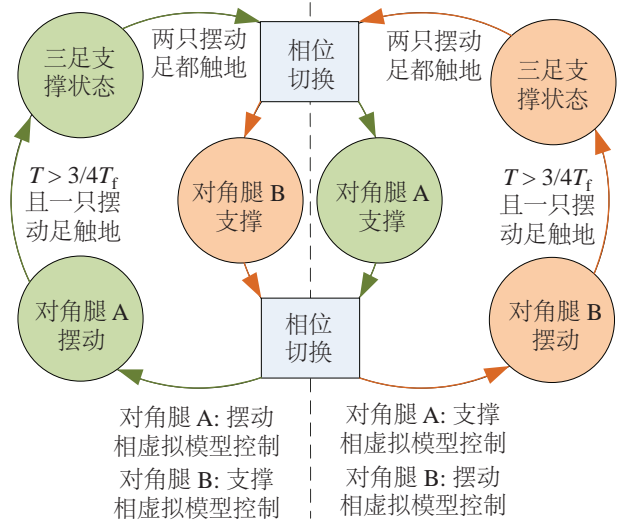


图 6 状态机原理图  
Fig.6 Schematic diagram of the state machine

6 实验 (Experiments)

为验证机器人虚拟模型的正确性，测试文中所设计控制器的性能，本文在 Webots 仿真环境中进行了四足机器人对角小跑步态行走实验。机器人能够在平坦地面上进行全方位移动，顺利地上下台阶和坡道，并能够承受来自侧向的外力冲击，展示了较好的运动平稳性和鲁棒性。仿真实验中的参数除特殊说明外，均采用表 2 中所列出的数据。仿真实验视频可参见网址：[http://v.youku.com/v\\_show/id\\_XMTMwMzEyNTAwNA==.html?from=y1.7-1.2](http://v.youku.com/v_show/id_XMTMwMzEyNTAwNA==.html?from=y1.7-1.2)。

6.1 平地行走实验

本文中第 1 个实验的目的是测试机器人的运动灵活性。初始时刻机器人在平坦地面上以对角小跑步态原地踏步，随后控制器驱动机器人以 0.5 m/s 的速度前进；一段时间后机器人停止前进，改为以 0.25 m/s 的速度向左侧横移；侧向运动测试结束后，机器人在原地绕躯干  $z_b$  轴转动，自转角速度达到

0.25 rad/s。紧接着，将前进、侧行和自转 3 种运动两两进行耦合，机器人实现了斜行、前行转弯和侧行转弯运动。最后，控制机器人同时进行前进、侧行和自转 3 种运动，测试机器人的运动稳定性和各方向运动的解耦性。

表 2 仿真实验所用参数  
Tab.2 Control parameters used in the simulation

参数名称	数值
机器人与地面摩擦因数	0.6
机器人与地面碰撞恢复系数	0.5
控制频率	1000 Hz
$T_f$	0.5 s
$T_s$	0.5 s
$h_d$	0.6 m
$z_h$	-0.5 m
$K_{\phi x}$	10000 N·m/rad
$K_{\phi y}$	800 N·m/(rad/s)
$K_{\psi y}$	10000 N·m/rad
$K_{\psi x}$	800 N·m/(rad/s)
$K_h$	8000 N/m
$K_{\dot{h}}$	800 N/(m/s)
$K_{v x}$	1000 N/(m/s)
$K_{\omega z}$	1000 N·m/(rad/s)
$k_{v y}$	0.05
$k_x$	100 N/m
$k_{\dot{x}}$	100 N/(m/s)
$k_y$	8000 N/m
$k_{\dot{y}}$	100 N/(m/s)
$k_z$	当 $t \leq \frac{3}{4}T_f$ 时，10000 N/m 当 $t > \frac{3}{4}T_f$ 时，1000 N/m
$k_{\dot{z}}$	100 N/(m/s)

实验过程中机器人躯干姿态和速度等记录于图 7 中，从上到下分别为前进速度曲线、侧向速度曲线、自转角速度曲线、躯干横滚角曲线、俯仰角曲线和躯干高度曲线，横坐标均为时间轴。黑色虚线为给定值，红色实线为实际值。从图中可以看出，整个过程躯干姿态平稳，机器人能够非常精确地跟踪前向速度和自转速度的给定值。运动过程中躯干侧向速度在给定值附近小幅振荡，这是因为本文方法并未直接控制作用于躯干质心沿  $y_b$  轴方向的力，躯干侧向速度通过控制摆动腿落脚点间接进行控制，尽管如此，其控制精度仍然可以接受。更重要的是，控制器成功实现了对前向、侧向和自转 3 种运动以及躯干姿态控制之间的解耦。图 8 自

左上至右下分别给出了实验过程中 1 s、10 s、25 s、40 s、55 s、70 s、85 s、105 s、120 s 时刻的实验场景截图。

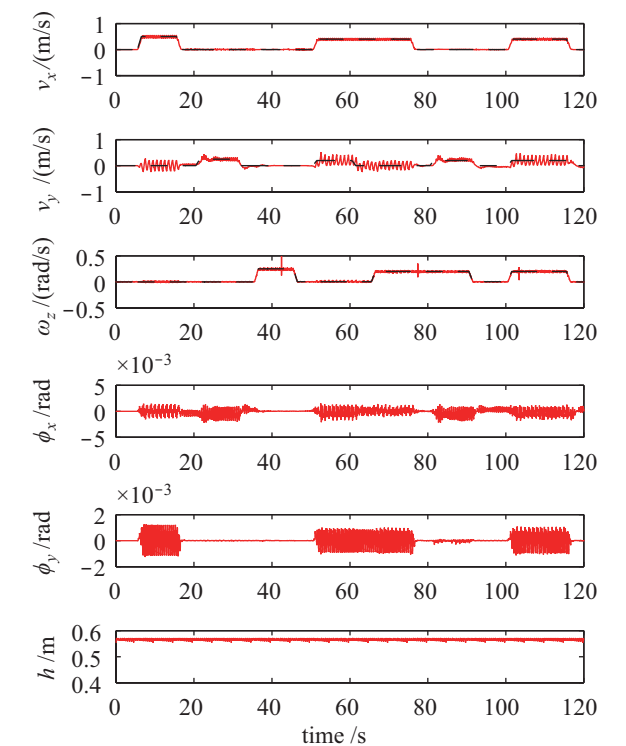


图 7 机器人对角小跑步态平地行走实验主要数据曲线  
Fig.7 Profiles of the key variables when the robot trotting on flat ground

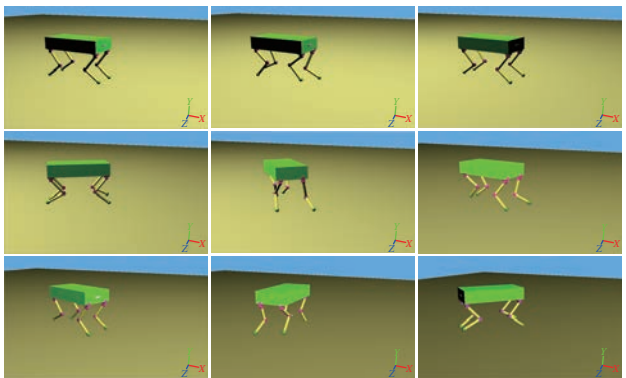


图 8 机器人对角小跑步态平地行走实验截图  
Fig.8 Snapshots showing the robot trotting on flat ground

6.2 不平坦地面行走实验

机器人步态控制器稳定性和鲁棒性的一个表现是其对不同地形的适应能力。实验中在机器人前进的道路上分别布置了 1 个高度为 0.1 m 的台阶和坡度为 0.3 rad 的坡道，台阶高度达到了机器人正常站立时横滚髋关节高度的 20%，坡道包括上坡和下坡。整个实验过程中，机器人没有任何关于地形的先验信息，也未安置任何环境探测传感器，仅仅在其上下台阶过程中人为地将足端摆动轨迹参数  $z_h$  改

为  $-0.45\text{ m}$  以获得足够的抬脚高度登上台阶，机器人行进完全依靠控制器的自主调整。

图 9 的最上方展示了本次实验的地形信息，从第 2 幅图到最下一幅图分别给出了机器人前进速度曲线、侧向速度曲线、自转角速度曲线、躯干横滚角曲线、俯仰角曲线和躯干高度曲线，横坐标均为躯干质心前进距离。黑色虚线为给定值，红色实线为实际值。从图中可以看出机器人成功地走过了仿真环境中设置的路面，躯干平稳性和速度跟随性也较好。实验过程截图请参见图 10。

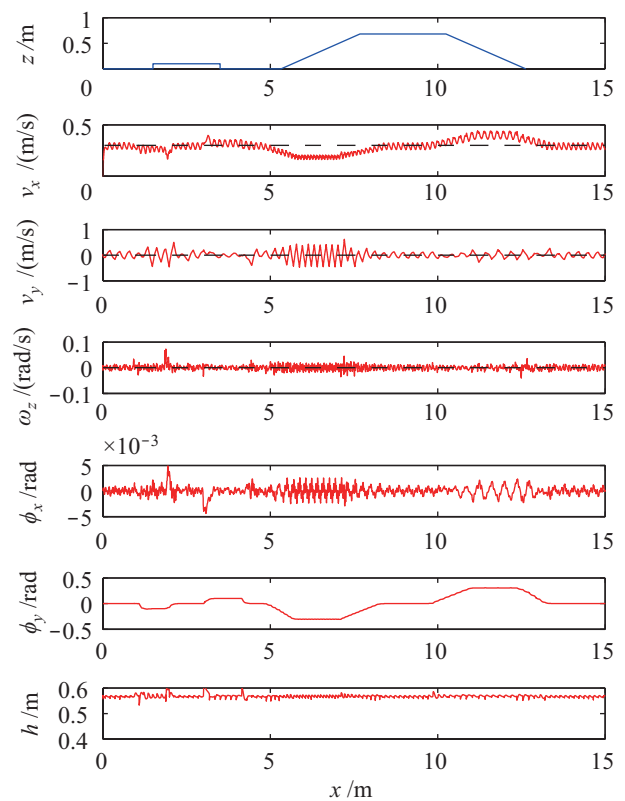


图 9 机器人对角小跑步态不平坦地面行走实验主要数据曲线  
Fig.9 Profiles of the key variables when the robot trotting on uneven ground

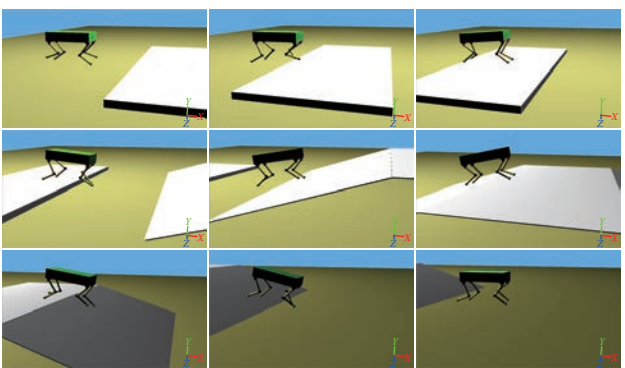


图 10 机器人对角小跑步态不平坦地面行走实验截图  
Fig.10 Snapshots showing the robot trotting on uneven ground



### 6.3 侧向冲击试验

机器人步态控制器稳定性和鲁棒性的另一个表现形式是其受到外界扰动时的自恢复能力。鉴于机器人模型前向长、侧向短的形态结构和对角腿支撑的运动特点,机器人躯干遭到侧方向的扰动时受到的影响更大,更难以保持平衡<sup>[21]</sup>。实验中当机器人以对角小跑步态原地踏步时受到来自体侧的摆锤撞击,作用在躯干上的冲量达到  $88\text{kg}\cdot\text{m/s}$  (摆锤质量为  $20\text{kg}$ , 撞击机器人前摆锤速度为  $4.4\text{m/s}$ )。为加快系统响应速度,本次实验中将机器人步态周期提高到  $2\text{Hz}$  ( $T_s = T_f = 0.25\text{s}$ )。当机器人躯干受到冲击后,机器人的运动速度与躯干姿态会偏离给定值,此时控制器通过在躯干质心处施加虚拟力来消除误差,随后将躯干质心的虚拟力转化为关节扭矩,控制支撑腿的运动。与此同时,摆动腿根据躯干侧向运动速度重新规划落足点,消除由冲击产生的侧向速度误差。图 11 展示了当机器人受冲击和恢复过程中的躯干姿态以及速度随时间变化曲线,其中垂直的黑色虚线指示出了撞击时刻,由图中可以看出,机器人能够顺利地消除冲击带来的影响并较快地恢复正常运动,整个过程躯干较为平稳。图 12 为实验过程截图。

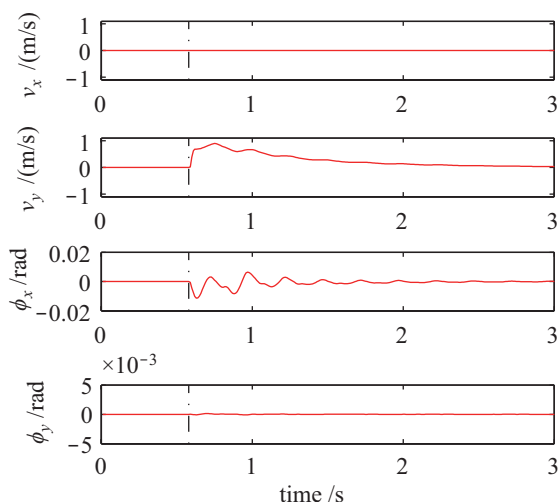


图 11 侧向冲击实验中机器人躯干速度与姿态曲线

Fig.11 Plots of the velocity and attitude of the robot collected from the lateral impact test

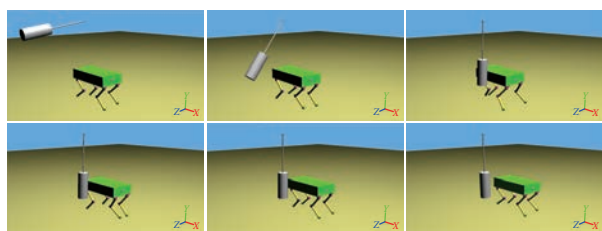


图 12 侧向冲击实验截图

Fig.12 Snapshots showing the lateral impact test

## 7 结论 (Conclusion)

本文介绍了一种基于虚拟模型的四足机器人对角小跑步态控制器的设计方法,控制器通过控制 2 条支撑腿的关节扭矩来调整作用在躯干质心上的虚拟力大小,从而控制机器人躯干高度与姿态角,以及前向速度和自转角速度;机器人侧向运动速度通过 Raibert 提出的规划摆动足落地点的方法进行调整。与此同时,本文根据机器人运动需求设计了摆动足运动轨迹,并通过在摆动足与期望运动轨迹之间虚拟“弹簧-阻尼”模型的方式控制摆动腿的运动。摆动相的终止取决于摆动腿的摆动时间和足端触地信息,并有专设的状态机控制机器人各腿部运动相位的切换。最后,本文在仿真环境中展示了机器人的全方位移动能力、应对不平坦地形能力和抗冲击能力。

本文在建模过程中仅仅对系统进行了静力分析,未考虑系统动力学,且忽略了机器人腿部的质量,这样大大简化了模型的复杂度,但也导致了控制中的误差。因此,文中的算法并非致力于实现精确的位置或力控制,而是一种基于虚拟设想的机器人运动直观控制方法,由模型不精确引起的控制误差可以看作是系统所受到的扰动。采用简化模型极大地降低了运算与控制复杂度,同时最后的仿真实验结果也证明了其控制效果依旧能够使人满意。

本文的主要贡献在于基于虚拟模型提出了一种四足机器人对角小跑步态建模方法,成功地实现了四足机器人对角小跑步态控制中 6 维运动方向 (3 个直线方向和 3 个转动方向) 的解耦,机器人具有较好的运动灵活性、地形适应性和抗扰性。控制方法直接,控制器设计较为简单,无需复杂的逆运动学和动力学运算。文中控制方法的另外一个容易被忽略的优点在于其并未完全锁定摆动腿的落足点位置,落足点在前后方向上的坐标可以自由设定,这将为机器人在崎岖地形行走时的路径选择和可落足点规划提供方便。

在将来的工作中,将对文中控制方法进行改进和延伸,使机器人能够实现具有四足腾空相的对角小跑步态。同时将在实际物理样机上测试算法的有效性。

## 参考文献 (References)

- [1] Nanua P, Waldron K J. Energy comparison between trot, bound, and gallop using a simple model[J]. Journal of Biomechanical Engineering, 1995, 117(4): 466-473.
- [2] Kimura H, Fukuoka Y, Cohen A H. Adaptive dynamic walking of a quadruped robot on natural ground based on biological

- concepts[J]. International Journal of Robotics Research, 2007, 26(5): 475-490.
- [3] Matos V, Santos C P. Omnidirectional locomotion in a quadruped robot: A CPG-based approach[C]//IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2010: 3392-3397.
- [4] Barasuol V, Buchli J, Semini C, et al. A reactive controller framework for quadrupedal locomotion on challenging terrain [C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2013: 2554-2561.
- [5] Xie H, Ahmadi M, Shang J, et al. An intuitive approach for quadruped robot trotting based on virtual model control[J]. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 2015, 229(4): 342-355.
- [6] Kurazume R, Yoneda K, Hirose S. Feedforward and feedback dynamic trot gait control for quadruped walking vehicle[J]. Autonomous Robots, 2002, 12(2): 157-172.
- [7] Zhang S, Gao J Y, Duan X G, et al. Trot pattern generation for quadruped robot based on the ZMP stability margin [C] //International Conference on Complex Medical Engineering. Piscataway, USA: IEEE, 2013: 608-613.
- [8] Raibert M H. Legged robots that balance[M]. Cambridge, USA: MIT Press, 1986.
- [9] Raibert M H. Trotting, pacing and bounding by a quadruped robot[J]. Journal of Biomechanics, 1990, 23(suppl.1): 79-98.
- [10] Raibert M, Blankespoor K, Nelson G, et al. Bigdog, the rough-terrain quadruped robot[C]//Proceedings of the 17th IFAC World Congress, vol.17. Laxenburg, Austria: IFAC, 2008: 10822-10825.
- [11] Michael K. Meet Boston dynamics' LS3-the latest robotic war machine[Z/OL]. [2015-07-29]. 2012. <http://ro.uow.edu.au/cgi/viewcontent.cgi?article=3782&context=eisapers>.
- [12] Pratt J. Virtual model control of a biped walking robot[D]. Cambridge, USA: MIT, 1995.
- [13] Pratt J, Torres A, Dilworth P, et al. Virtual actuator control[C] //IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 1996: 1219-1226.
- [14] Pratt J, Chew C M, Torres A, et al. Virtual model control: An intuitive approach for bipedal locomotion[J]. International Journal of Robotics Research, 2001, 20(2): 129-143.
- [15] Havoutis I, Semini C, Caldwell D G. Virtual model control for quadrupedal trunk stabilization[Z/OL]. [2015-07-29]. [http://ihavoutis.github.io/publications/2013/dw13\\_havoutis.pdf](http://ihavoutis.github.io/publications/2013/dw13_havoutis.pdf).
- [16] Gehring C, Coros S, Hutter M, et al. Control of dynamic gaits for a quadrupedal robot[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2013: 3287-3292.
- [17] 鄂明成, 刘虎, 张秀丽, 等. 一种粗糙地形下四足仿生机器人的柔顺步态生成方法[J]. 机器人, 2014, 36(5): 584-591.
- E M C, Liu H, Zhang X L, et al. Compliant gait generation for a quadruped bionic robot walking on rough terrains[J]. Robot, 2014, 36(5): 584-591.
- [18] Yoneda K, Iiyama H, Hirose S. Intermittent trot gait of a quadruped walking machine dynamic stability control of an omnidirectional walk[C]//IEEE International Conference on Robotics and Automation. Piscataway, USA: IEEE, 1996: 3002-3007.

- [19] Rong X, Li Y, Ruan J, et al. Design and simulation for a hydraulic actuated quadruped robot[J]. Journal of Mechanical Science and Technology, 2012, 26(4): 1171-1177.
- [20] 柴汇, 孟健, 荣学文, 等. 高性能液压驱动四足机器人 SCalf 的设计与实现[J]. 机器人, 2014, 36(4): 385-391.
- Chai H, Meng J, Rong X W, et al. Design and implementation of SCalf, an advanced hydraulic quadruped robot[J]. Robot, 2014, 36(4): 385-391.
- [21] Havoutis I, Semini C, Buchli J, et al. Quadrupedal trotting with active compliance[C]//IEEE International Conference on Mechatronics. Piscataway, USA: IEEE, 2013: 610-616.

## 附录: 矩阵 $Q$ 的逆

由式 (18) 可以看出,  $Q^{-1}$  右乘的矩阵最后 1 行元素为 0, 因此  $Q^{-1}$  最后 1 列可以忽略. 为表示方便, 记

$$P^{-1} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & \times \\ q_{21} & q_{22} & q_{23} & q_{24} & q_{25} & \times \\ q_{31} & q_{32} & q_{33} & q_{34} & q_{35} & \times \\ q_{41} & q_{42} & q_{43} & q_{44} & q_{45} & \times \\ q_{51} & q_{52} & q_{53} & q_{54} & q_{55} & \times \\ q_{61} & q_{62} & q_{63} & q_{64} & q_{65} & \times \end{bmatrix} \quad (38)$$

其中各元素的具体表达形式为

$$q_{11} = \frac{(x_F - x_H)y_H z_F - ((x_F + x_H)y_F - 2x_F y_H)z_H}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (39)$$

$$q_{12} = \frac{(x_F + x_H)(x_F y_H - x_H y_F)}{2(y_F - y_H)(x_F z_H - x_H z_F)} \quad (40)$$

$$q_{13} = \frac{(x_F - x_H)(x_F + x_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (41)$$

$$q_{14} = \frac{x_F + x_H}{2x_H z_F - 2x_F z_H} \quad (42)$$

$$q_{15} = \frac{(x_F - x_H)(z_F + z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (43)$$

$$q_{21} = \frac{y_H z_F - y_F z_H}{2x_H z_F - 2x_F z_H} \quad (44)$$

$$q_{22} = \frac{x_H y_F - x_F y_H}{2x_H z_F - 2x_F z_H} \quad (45)$$

$$q_{23} = \frac{x_F - x_H}{2x_H z_F - 2x_F z_H} \quad (46)$$

$$q_{24} = \frac{y_F - y_H}{2x_H z_F - 2x_F z_H} \quad (47)$$

$$q_{25} = \frac{z_F - z_H}{2x_H z_F - 2x_F z_H} \quad (48)$$

$$q_{31} = \frac{(z_F + z_H)(y_H z_F - y_F z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (49)$$

$$q_{32} = \frac{x_F y_H (z_H - z_F) + x_H (y_F (z_F + z_H) - 2y_H z_F)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (50)$$

$$q_{33} = \frac{(x_F + x_H)(z_F - z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (51)$$

$$q_{34} = \frac{z_F + z_H}{2x_H z_F - 2x_F z_H} \quad (52)$$

$$q_{35} = \frac{(z_F - z_H)(z_F + z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (53)$$

$$q_{41} = \frac{(2x_H y_F - (x_F + x_H)y_H)z_F + (x_H - x_F)y_F z_H}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (54)$$

$$q_{42} = \frac{(x_F + x_H)(x_F y_H - x_H y_F)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (55)$$

$$q_{43} = \frac{(x_F - x_H)(x_F + x_H)}{2(y_F - y_H)(x_F z_H - x_H z_F)} \quad (56)$$

$$q_{44} = \frac{x_F + x_H}{2x_F z_H - 2x_H z_F} \quad (57)$$

$$q_{45} = -\frac{(x_F - x_H)(z_F + z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (58)$$

$$q_{51} = \frac{y_H z_F - y_F z_H}{2x_H z_F - 2x_F z_H} \quad (59)$$

$$q_{52} = \frac{x_H y_F - x_F y_H}{2x_H z_F - 2x_F z_H} \quad (60)$$

$$q_{53} = \frac{x_F - x_H}{2x_H z_F - 2x_F z_H} \quad (61)$$

$$q_{54} = \frac{y_F - y_H}{2x_H z_F - 2x_F z_H} \quad (62)$$

$$q_{55} = \frac{z_F - z_H}{2x_H z_F - 2x_F z_H} \quad (63)$$

$$q_{61} = \frac{(z_F + z_H)(y_F z_H - y_H z_F)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (64)$$

$$q_{62} = \frac{x_H y_F(z_F - z_H) + x_F(y_H(z_F + z_H) - 2y_F z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (65)$$

$$q_{63} = -\frac{(x_F + x_H)(z_F - z_H)}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (66)$$

$$q_{64} = \frac{z_F + z_H}{2x_F z_H - 2x_H z_F} \quad (67)$$

$$q_{65} = \frac{z_H^2 - z_F^2}{2(y_F - y_H)(x_H z_F - x_F z_H)} \quad (68)$$

### 作者简介:

张国腾 (1989-), 男, 博士生. 研究领域: 机器人运动控制.

荣学文 (1973-), 男, 博士, 高级工程师. 研究领域: 机器人机构设计与分析, 液压伺服驱动系统.

李贻斌 (1960-), 男, 博士, 教授. 研究领域: 机器人技术, 机电一体化技术, 模式识别与智能系统.

(上接第 63 页)

- [20] Svanberg K. A class of globally convergent optimization methods based on conservative convex separable approximations[J]. SIAM Journal on Optimization, 2002, 12(2): 555-573.
- [21] Jones D R, Schonlau M, Welch W J. Efficient global optimization of expensive black-box functions[J]. Journal of Global Optimization, 1998, 13(4): 455-492.
- [22] Kanehiro F, Hirukawa H, Kajita S. OpenHRP: Open architecture humanoid robotics platform[J]. International Journal of Robotics Research, 2004, 23(2): 155-165.
- [23] OpenHRP Team, Humanoid Research Group. OpenHRP users manual sample humanoid robot[DB/OL]. (2012-10-15) [2015-11-17]. [http://fkanehiro.github.io/openhrp3-doc/en/sample\\_model.html](http://fkanehiro.github.io/openhrp3-doc/en/sample_model.html).
- [24] Roustant O, Ginsbourger D, Deville Y. DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by Kriging-based metamodeling and optimization[J]. Journal of Statistical Software, 2013, 51(1): 1-55.

- [25] Viana F A. SURROGATES toolboxes[DB/OL]. (2011-11-1) [2015-11-17]. <https://sites.google.com/site/felipeacviana/surrogatetoolbox>.
- [26] Lophaven S N, Nielsen H B, Sndergaard J. DACE: A Matlab Kriging toolbox[EB/OL]. (2007-11-07) [2015-11-17]. [http://www2.imm.dtu.dk/pubdb/views/publication\\_details.php?id=1460](http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=1460).

### 作者简介:

张继文 (1983-), 男, 博士, 助理研究员. 研究领域: 仿人机器人运动控制与智能决策.

刘 莉 (1965-), 女, 博士, 研究员. 研究领域: 仿人机器人理论与技术.

李昌硕 (1987-), 男, 博士生. 研究领域: 仿人机器人智能决策, 仿人机器人环境建模.