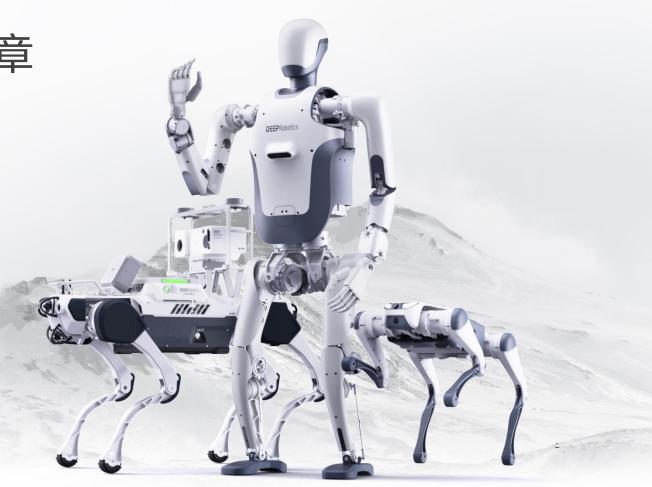


《足式机器人运动控制》第七章

虚拟模型控制

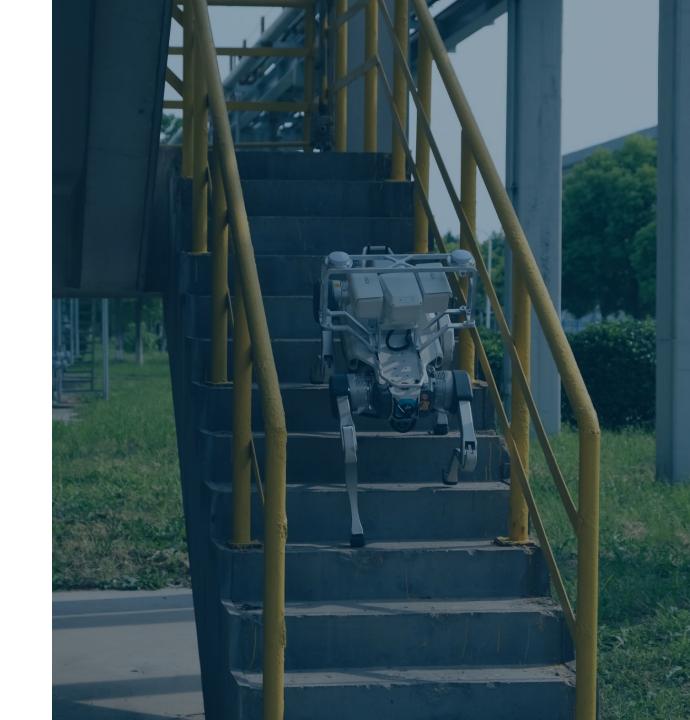








- 一、关节控制模型
- 二、单刚体模型
- 三、虚拟模型控制





□ 机器人关节



J60 -6

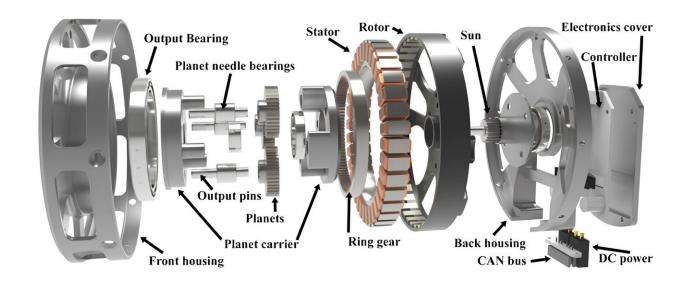
J60 -10

| 480g | 重量 | 540g | |
|-------------|-----------------|---------------|--|
| 76.5mm*63mm | 尺寸(直径×高) ——— | 76.5mm*72.5mm | |
| DC 12V~36V | 工作电压范围 | DC 12V~36V | |
| DC 24V | 标准工作电压 | DC 24V | |
| 19.94N·m | 峰值扭矩 | 30.50N · m | |
| 24.18rad/s | 峰值转速 | 15.49rad/s | |
| 0.8982N·m/A | 关节扭矩常数 | 1.3473N⋅m/A | |
| 41.54N⋅m/kg | 最大扭矩重量比 | 56.48N⋅m/kg | |
| 30A | 电机最大相电流 | 30A | |
| CAN | 通信方式 | CAN | |
| 1Mbps | 通信波特率 | 1Mbps | |
| 1kHz | 通信控制频率 | 1kHz | |
| 绝对值编码器 | 编码器类型 | 绝对值编码器 | |
| 14bit | 编码器分辨率 | 14bit | |
| | | | |

⇒ 关节模型

□ 关节构成

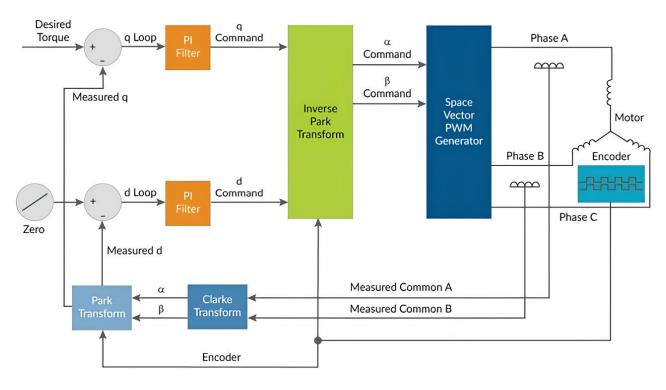
驱动器+电机+减速器+(输出转动轴)



| 组成 | 描述 |
|-----|-----------------------|
| 驱动器 | FOC算法驱动,单编或双编 |
| 电机 | 无刷直流电机,内转子或外转 子 |
| 减速器 | 行星齿轮组,谐波减速器,滚 柱丝杠等 |



□ 关节控制



关节控制方式:

$$T = kp * (pos_{goal} - pos_{real}) + kd * (vel_{goal} - vel_{real}) + t_{ff}$$

Pybullet

setJointMotorControl2/Array

| method | implementation | component | constraint error to be minimized |
|------------------|----------------|-------------------------------------|--|
| POSITION_CONTROL | constraint | velocity and position constraint | error = position_gain * (desired_p osition - actual_position) + velocit y_gain * (desired_velocity - actura l_velocity) |
| VELOCITY_CONTROL | constraint | pure velocity constraint | error = desired_velocity - actural_velocity |
| TORQUE_CONTROL | external force | | |

RaiSim

inline voidsetControlMode(ControlMode::Typemode)

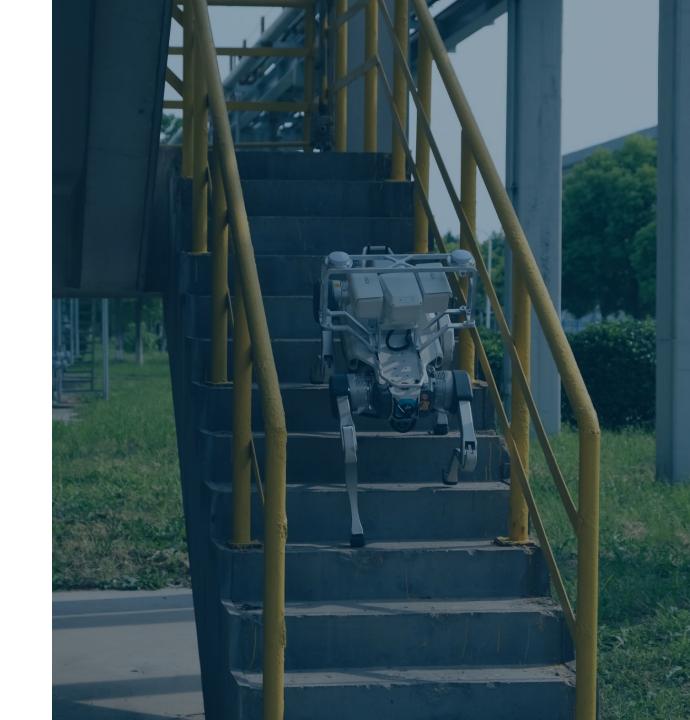
Parameters: mode - [in] control mode. Can be either

ControlMode::FORCE_AND_TORQUE or

ControlMode::PD_PLUS_FEEDFORWARD_TORQUE



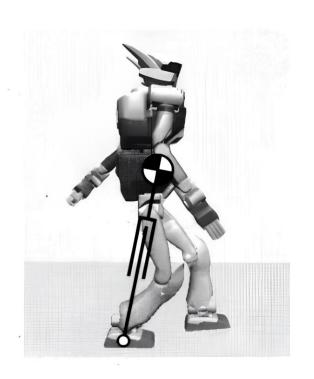
- 一、关节控制模型
- 二、单刚体模型
- 三、虚拟模型控制

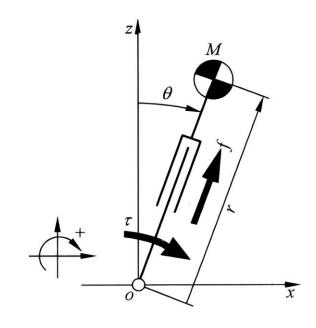




□ 线性倒立摆模型(Linear Inverted Pendulum)

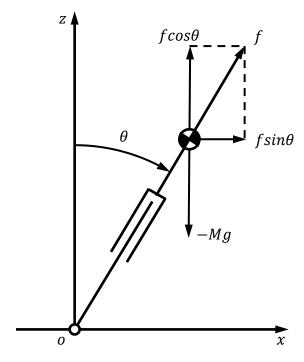
- 一、假定机器人的所有质量集中于 其质心位置。
- 二、假设机器人的腿无质量,它与 地面的接触是通过一个可以转动的 支点实现的
- 三、当前先忽略机器人侧向的移动, 机器人的移动只限于xOz平面





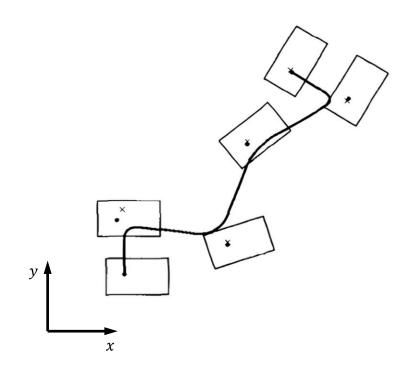


□ 线性倒立摆模型(Linear Inverted Pendulum)



倒立摆在摔倒的过程中通过伸展其腿长而保持 恒定的质心高度

$$M\ddot{x} = \frac{Mg}{\cos\theta}\sin\theta = Mg\tan\theta = Mg\frac{x}{z}$$



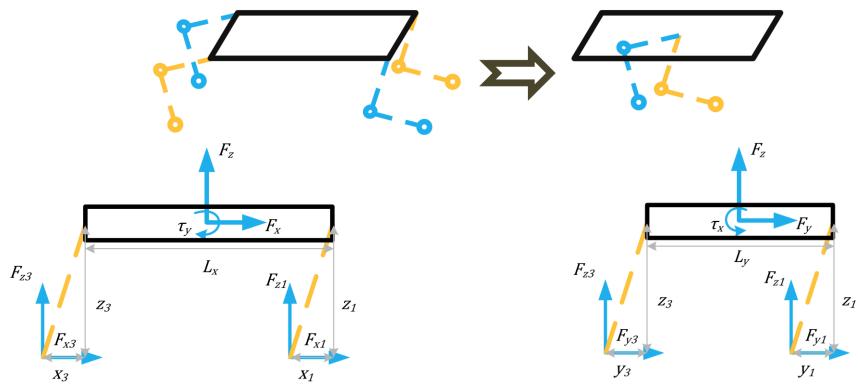
倒立摆模型和ZMP稳定性已经生成的步行轨迹



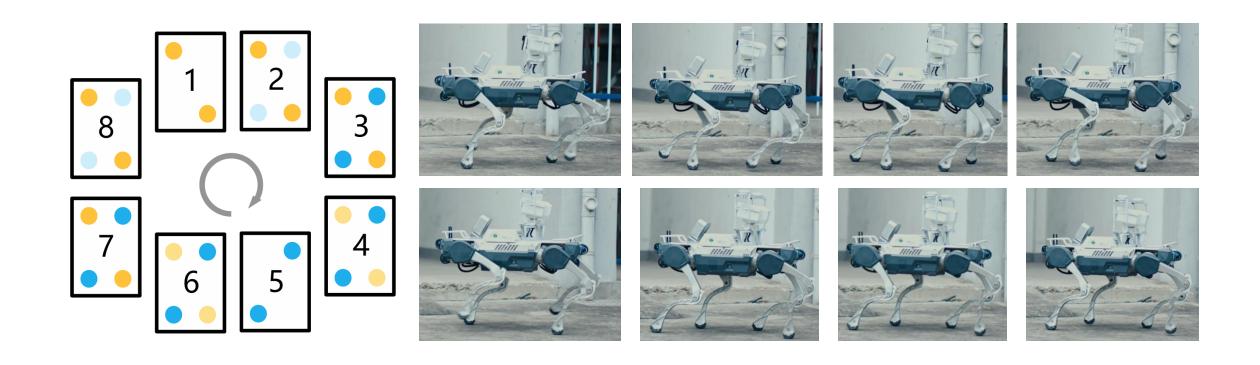
□ 线性倒立摆模型(Linear Inverted Pendulum)——在四足机器人上的应用

将同一条对角线上的两条腿等效为对角虚拟腿,

等效需满足同步性与力等效性。



□ 线性倒立摆模型(Linear Inverted Pendulum)——在四足机器人上的应用

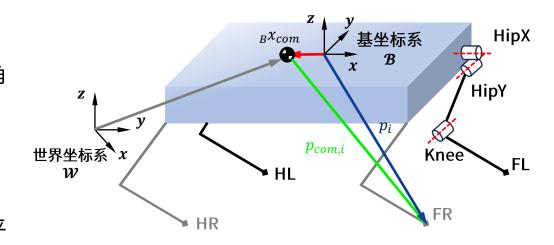




⇒ 单刚体动力学方程

机器狗的单刚体动力学模型

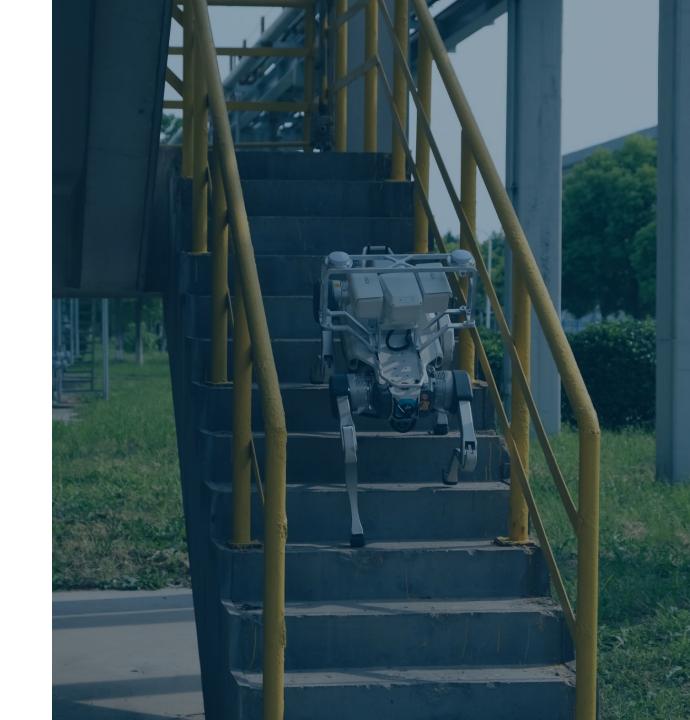
- 认为机器狗的重量全部集中在身体上。身体中心的速度和角 速度就是质心的角速度和速度。
- 忽略科氏力和离心力的影响。
- 因为四足机器人平台是点状足,假设足底不会产生力矩。
- 假设只有足底与地面产生的作用力作用在四足机器人系统平 台上。



$$m\ddot{p} = \sum_{i=1}^{n_c} f_i - G$$
 $\frac{d}{dt}(I\omega) = \sum_{i=1}^{n_c} r_i \times f_i$



- 一、关节控制模型
- 二、单刚体模型
- 三、虚拟模型控制





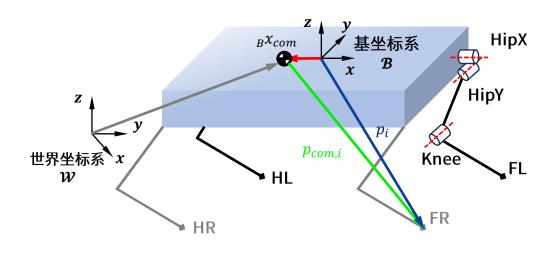
虚拟模型控制

□ 机器狗的单刚体动力学模型

$$\begin{bmatrix} \tau \\ f \end{bmatrix} = \begin{bmatrix} I\dot{\omega} + \omega \times I\omega \\ m\ddot{p} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{4} p_{com,i} \times f_{i} \\ \sum_{i=1}^{4} f_{i} - mg \end{bmatrix}$$

$$\begin{bmatrix} I & \cdots & I \\ [p_{com,1} \times] & \cdots & [p_{com,c} \times] \end{bmatrix} \begin{bmatrix} f_{1} \\ \cdots \\ f_{c} \end{bmatrix} = \begin{bmatrix} m(\ddot{x}_{com}^{d} + g) \\ I_{g}\dot{\omega}_{b}^{d} \end{bmatrix}$$

质心基座的加速度可以由PD控制器获得



$$\ddot{x}_{com}^d = K_{pcom}(x_{com}^d - x_{com}) + K_{dcom}(\dot{x}_{com}^d - \dot{x}_{com})$$
$$\dot{\omega}_b^d = K_{pbase}log(R_b^d R_b^\intercal) + K_{dbase}(\omega_b^d - \omega_b)$$

其中 $R_b \in \mathbb{R}^{3\times3}$ 和 $R_b^d \in \mathbb{R}^{3\times3}$ 是旋转矩阵,分别表示基座相对于世界参照系的实际方向和期望方向, $log(.): \mathbb{R}^{3\times3} \to \mathbb{R}^3$ 是旋转矩阵到相关旋转矢量的映射

□ 计算所需的地面反作用力

在给定了质心加速度和机器人底座角加速度的期望值后,我们希望计算出期望GRFs (Ground Reaction Forces) f:

$$\underbrace{\begin{bmatrix} I & \cdots & I \\ [p_{com,1} \times] & \cdots & [p_{com,1} \times] \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} f_1 \\ \cdots \\ f_c \end{bmatrix}}_{f} = \underbrace{\begin{bmatrix} m(\ddot{x}_{com}^d + g) \\ I_g \dot{\omega}_b^d \end{bmatrix}}_{b}$$

在每个控制周期中,我们都要求解以下QP问题:

$$f^{d} = \underset{f \in \mathbb{R}^{k}}{\operatorname{argmin}} (Af - b)^{\mathsf{T}} S (Af - b) + \alpha f^{\mathsf{T}} W f$$

s.t. $\underline{d} < Cf < \bar{d}$

其中, $S \in \mathbb{R}^{6 \times 6}$ 和 $W \in \mathbb{R}^{6 \times 6}$ 为正有限权矩阵, $\alpha \in \mathbb{R}$ 是正则化系数, $C \in \mathbb{R}^{p \times k}$ 为不等式约束矩阵, $\underline{d}, \overline{d} \in \mathbb{R}^p$ 分别为下限/上限:

- GRFs位于摩擦锥内;
- 支撑腿需要保持在触地状态。

$$\min_{f} ||Af - b|| \qquad \qquad f^* = A^T (AA^T)^{-1} b$$

虚拟模型控制

□ 计算所需的地面反作用力

用方形金字塔来近似摩擦锥,用线性约束来表示它们。然后,我们将 $C, \underline{d}, \overline{d}$ 定义为:

$$C = \begin{bmatrix} C_0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_c \end{bmatrix}, \underline{d} = \begin{bmatrix} d_0 \\ \vdots \\ d_c \end{bmatrix}, \bar{d} = \begin{bmatrix} \bar{d}_0 \\ \vdots \\ \bar{d}_c \end{bmatrix}$$

$$C_{i} = \begin{bmatrix} (-\mu_{i}n_{i} + t_{1_{i}})^{\mathsf{T}} \\ (-\mu_{i}n_{i} + t_{2_{i}})^{\mathsf{T}} \\ (\mu_{i}n_{i} + t_{2_{i}})^{\mathsf{T}} \\ (\mu_{i}n_{i} + t_{1_{i}})^{\mathsf{T}} \end{bmatrix}, \underline{d_{i}} = \begin{bmatrix} -\infty \\ -\infty \\ 0 \\ 0 \\ f_{min_{i}} \end{bmatrix}, \bar{d_{i}} = \begin{bmatrix} 0 \\ 0 \\ \infty \\ f_{max_{i}} \end{bmatrix}$$

其中, $n_i \in \mathbb{R}^3$ 是地面法线方向, $t_{1_i}, t_{2_i} \in \mathbb{R}^3$ 是前进和横向方向, $\mu_i \in \mathbb{R}$ 是摩擦系数, $f_{min_i}, f_{max_i} \in \mathbb{R}$ 分别是第i个法向力的最小允许值最大允许值

□ 地形估计

在不加感知传感器的情况下机器人对类似斜坡的地面需要做姿态适应,需要对行走地面做地形估计假设地面的平面方程在世界坐标系下为:

$$Z = Ax + By + C$$

假设机器人的四条腿在最近的接触点坐标分别为:

$$p_c^i = [p_x^i \quad p_y^i \quad p_z^i]^T, i \in [1, 2, 3, 4]$$

根据最小二乘法可以计算:

$$W = \begin{bmatrix} p_x^1 & p_y^1 & 1 \\ p_x^2 & p_y^2 & 1 \\ p_x^3 & p_y^3 & 1 \\ p_x^4 & p_y^4 & 1 \end{bmatrix} \qquad p_z = [p_z^1 \quad p_z^2 \quad p_z^3 \quad p_z^4]^T \qquad \Longrightarrow \qquad \begin{bmatrix} A \\ B \\ C \end{bmatrix} = (W^T W)^+ W^T p_z$$

□ 地面反作用力映射到关节扭矩

通过叠加两个控制动作来计算所需的关节扭矩 $t_d \in \mathbb{R}^n$ (n 为关节数):

● 将所需的 GRF f^d 映射到关节空间,我们就得到了前馈扭矩 au_{ff} :

$$\tau_{ff} = -SJ_c^{\mathsf{T}} f^d$$

 $J_c \in \mathbb{R}^{k \times n + 6}$ 是接触点的叠加雅各布系数, $S = [I_{n \times n} \quad 0_{n \times 6}]$ 是选择矩阵,用于选择驱动的 DoFs。

● 增益较低的比例-派生(PD)关节位置控制器, $\tau^d \approx 18\%$:

$$\tau^d = \tau_{ff} + PD(q^d, \dot{q}^d, c_{st})$$

 $q^d \in \mathbb{R}^n$; $\dot{q}^d \in \mathbb{R}^n$ 分别是期望的关节位置和速度, $c_{st} \in \mathbb{R}^4$ 是布尔变量向量,代表双腿的姿态条件。



□ 摆动腿控制

当前腿处于摆动周期时,可以利用PD控制实现摆动腿根据设定摆动轨迹的跟随:

$$\tau_i = K_p(p_i^d - p_i) + K_d(v_i^d - v_i) + \tau_{ff,i}$$

其中:

 K_p, K_d 关节位置和速度增益

 p_i^d , v_i^d 期望关节位置和速度,根据设定摆动腿轨迹求得

 $au_{ff,i}$ 前馈力矩,通过动力学等方式求得

摆动腿落脚点计算:

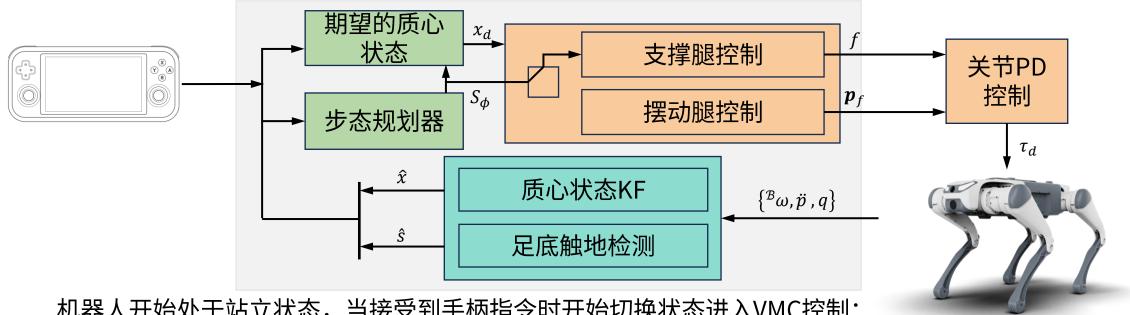
$$p_i^{step} = p_i^{hipx} + \frac{T}{2}\dot{p}_{com} + \sqrt{\frac{z_{com}}{\|g\|}}(\dot{p}_{com} - \dot{p}_{com}^d)$$
Raibert Heuristic Capture Point

实际应用中



$$p_i^{step} = p_i^{hipx} + k_1 \dot{p}_{com} + k_2 (\dot{p}_{com} - \dot{p}_{com}^d)$$

□ 控制流程



机器人开始处于站立状态,当接受到手柄指令时开始切换状态进入VMC控制:

- 1. 初始化状态估计参数 p_0, v_0
- 2. 进入到虚拟模型控制后:
 - ① 根据控制指令输入获得质心期望位置,速度等 $p_{com}^d, v_{com}^d, R_{com}^d$
 - ② 根据控制指令输入获得期望的步态输入,可以获得步态周期,支撑腿和摆动腿规划
 - ③ 状态估计实时更新
 - ④ 摆动相末期触地检测
 - ⑤ 控制指令输入
- 3. 获得结束指令,在适当状态下结束

➡ 虚拟模型控制

Project

- 验算每条腿的雅克比矩阵是否计算正确
- 在仿真环境中用伪逆和QP求解的方法实现VMC站立
- 实现姿态扭动变化



感谢聆听 ThanksforListening