1. 推导“多刚体”模型的ZMP计算公式

$$p_x = \frac{Mgx + p_z\dot{\mathcal{P}}_x - \dot{\mathcal{L}}_y}{Mg + \dot{\mathcal{P}}_z} \qquad p_y = \frac{Mgy + p_z\dot{\mathcal{P}}_y + \dot{\mathcal{L}}_x}{Mg + \dot{\mathcal{P}}_z}$$

I'll do some review first.
recall the 10 parameters to discribe a robot

Mass : M
Center of Mass (CoM) : $c \equiv [c_x\ c_y\ c_z]^T$ } about world reference frame
Momentum : $\mathcal{P} \equiv [\mathcal{P}_x\ \mathcal{P}_y\ \mathcal{P}_z]^T$
Angular momentum : $\mathcal{L} \equiv [\mathcal{L}_x\ \mathcal{L}_y\ \mathcal{L}_z]^T$ } about original

① $\dot{c} = v = \mathcal{P}/M \rightarrow \dot{c} = \mathcal{P}/M$

② $\mathcal{P} = \dot{c}M$, $\dot{\mathcal{P}} = \ddot{c}M = f_{all} \rightarrow \dot{\mathcal{P}} = f_{all}$

③ $\mathcal{L} = c \times \mathcal{P}$,

$$\begin{aligned}\dot{\mathcal{L}} &= \dot{c} \times \mathcal{P} + c \times \dot{\mathcal{P}} \\ &= \dot{c} \times \dot{c}M + c \times f_{all} \\ &= \tau\end{aligned} \qquad \rightarrow \dot{\mathcal{L}} = \tau_{all}$$

denote f as all forces but gravity.
then $\dot{\mathcal{P}} = Mg + f$
denote τ as the momentum without the gravity effect
then $\dot{\mathcal{L}} = c \times Mg + \tau$

Now we derivate ZMP

$$\underbrace{\tau}_{\text{world 处 torque}} = \underbrace{p}_{\text{ZMP点}} \times \underbrace{f}_{\text{ZMP点处力}} + \underbrace{\tau_p}_{\text{ZMP处 torque, where } \tau_{px} = \tau_{py} = 0}$$

we will need to transfer the τ to origin of {w} frame
as $\begin{cases} \dot{\mathcal{P}} = Mg + f \\ \dot{\mathcal{L}} = c \times Mg + \tau \end{cases}$ are all expressed in {w} frame.

apply $\tau_p = \tau - p \times f$ to above.

$$\tau_p = \dot{\mathcal{L}} - c \times Mg - p \times (\dot{\mathcal{P}} - Mg)$$

$$\tau_p = \dot{\mathcal{L}} - c \times Mg + \dot{\mathcal{P}} \times p - Mg \times p$$

$$\begin{cases} a \equiv [a_x\ a_y\ a_z]^T \\ b \equiv [b_x\ b_y\ b_z]^T \\ a \times b = \begin{pmatrix} a_y b_z - a_z b_y \\ a_z b_x - a_x b_z \\ a_x b_y - a_y b_x \end{pmatrix} \end{cases}$$

scalar $\downarrow\uparrow$ vector

$$\tau_{px} = \dot{\mathcal{L}}_x - M(c_y g_z - c_z g_y) + \dot{\mathcal{P}}_z p_y - \dot{\mathcal{P}}_y p_z - M(g_y p_z - g_z p_y) = 0$$

$$\tau_{py} = \dot{\mathcal{L}}_y - M(c_z g_x - c_x g_z) + \dot{\mathcal{P}}_z p_x - \dot{\mathcal{P}}_x p_z - M(g_z p_x - g_x p_z) = 0$$

denote $[c_x\ c_y\ c_z]^T \equiv [x\ y\ z]^T$

$[g_x\ g_y\ g_z]^T \equiv [0\ 0\ -g]^T$

then $\begin{cases} \dot{\mathcal{L}}_x - Myg + \dot{\mathcal{P}}_z p_y - \dot{\mathcal{P}}_y p_z + Mg p_y = 0 \\ \dot{\mathcal{L}}_y + Mxg + \dot{\mathcal{P}}_z p_x - \dot{\mathcal{P}}_x p_z - Mg p_x = 0 \end{cases}$

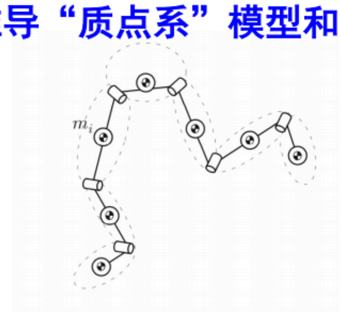
solve for p_x and p_y :

$$p_x = \frac{Mgx + p_z \dot{\mathcal{P}}_z - \dot{\mathcal{L}}_y}{Mg + \dot{\mathcal{P}}_z}$$

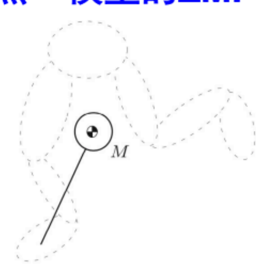
$$p_y = \frac{Mgy + p_z \dot{\mathcal{P}}_y + \dot{\mathcal{L}}_x}{Mg + \dot{\mathcal{P}}_z}$$

$\square$

2. 推导“质点系”模型和“单质点”模型的ZMP计算公式



(a) Approximation by point mass system



(b) Approximation by a sigle point mass

(a)

$$\mathcal{P} = \sum_{i=1}^{N} \mathcal{P}_i = \sum_{i=1}^{N} \dot{c}_i m_i$$

$$\mathcal{L} = \sum_{i=1}^{N} c_i \times \mathcal{P}_i = \sum_{i=1}^{N} m_i (c_i \times \dot{c}_i)$$

we denote $c \equiv [c_x\ c_y\ c_z]^T \equiv [x\ y\ z]^T$

then $\begin{cases} P_x = \sum_{i=1}^{N} m_i \dot{x}_i \\ P_y = \sum_{i=1}^{N} m_i \dot{y}_i \\ P_z = \sum_{i=1}^{N} m_i \dot{z}_i \end{cases}$ $\begin{cases} L_x = \sum_{i=1}^{N} m_i (y_i \dot{z}_i - z_i \dot{y}_i) \\ L_y = \sum_{i=1}^{N} m_i (z_i \dot{x}_i - x_i \dot{z}_i) \\ L_z = \sum_{i=1}^{N} m_i (x_i \dot{y}_i - y_i \dot{x}_i) \end{cases}$

By last problem: $p_x = \dfrac{Mgx + P_z \dot{P}_x - \dot{L}_y}{Mg + \dot{P}_z} = \dfrac{P_x g + P_z \dot{P}_x - \dot{L}_y}{Mg + \dot{P}_z}$

$$p_y = \frac{Mgy + P_z \dot{P}_y + \dot{L}_x}{Mg + \dot{P}_z} = \frac{P_y g + P_z \dot{P}_y + \dot{L}_x}{Mg + \dot{P}_z}$$

$\begin{cases} \dot{P}_x = \sum_{i=1}^{N} m_i \ddot{x}_i \\ \dot{P}_y = \sum_{i=1}^{N} m_i \ddot{y}_i \\ \dot{P}_z = \sum_{i=1}^{N} m_i \ddot{z}_i \end{cases}$ $\begin{cases} \dot{L}_x = \sum_{i=1}^{N} m_i (\dot{y}_i \dot{z}_i + y_i \ddot{z}_i - \dot{z}_i \dot{y}_i - z_i \ddot{y}_i) = \sum_{i=1}^{N} m_i (y_i \ddot{z}_i - z_i \ddot{y}_i) \\ \dot{L}_y = \sum_{i=1}^{N} m_i (\dot{z}_i \dot{x}_i + x_i \ddot{z}_i - \dot{x}_i \dot{z}_i - z_i \ddot{x}_i) = \sum_{i=1}^{N} m_i (x_i \ddot{z}_i - z_i \ddot{x}_i) \end{cases}$

$$p_x = \frac{\sum_{i=1}^{N} m_i x_i g + P_z m_i \ddot{x}_i - m_i (x_i \ddot{z}_i - z_i \ddot{x}_i)}{\sum_{i=1}^{N} m_i g + m_i \ddot{z}_i}$$

$$p_y = \frac{\sum_{i=1}^{N} m_i y_i g + P_z m_i \ddot{y}_i + m_i (y_i \ddot{z}_i - z_i \ddot{y}_i)}{\sum_{i=1}^{N} m_i g + m_i \ddot{z}_i}$$

simplify will we have :

$$p_x = \frac{\sum_{i=1}^{N} m_i [x_i (g + \ddot{z}_i) - (z_i - P_z) \ddot{x}_i]}{\sum_{i=1}^{N} m_i (g + \ddot{z}_i)}$$

$$p_y = \frac{\sum_{i=1}^{N} m_i [y_i (g + \ddot{z}_i) - (z_i - P_z) \ddot{y}_i]}{\sum_{i=1}^{N} m_i (g + \ddot{z}_i)}$$

(b) $P = M\dot{c}$ $\quad L = c \times M\dot{c}$

$\begin{cases} \dot{P}_x = M\ddot{x} \\ \dot{P}_y = M\ddot{y} \\ \dot{P}_z = M\ddot{z} \end{cases}$ $\begin{cases} \dot{L}_x = M(y\ddot{z} - z\ddot{y}) \\ \dot{L}_y = M(z\ddot{x} - x\ddot{z}) \\ \dot{L}_z = M(x\ddot{y} - y\ddot{x}) \end{cases}$

$$p_x = \frac{\dot{\mathcal{P}}_x g + p_z \dot{\mathcal{P}}_x - \dot{\mathcal{L}}_y}{Mg + \dot{\mathcal{P}}_z} = \frac{M\ddot{x}g + p_z M\ddot{x} - Mz\ddot{x} + Mx\ddot{z}}{Mg + M\ddot{z}} = x - \frac{(z-p_z)\ddot{x}}{\ddot{z}+g}$$

$$p_x = \frac{\dot{\mathcal{P}}_y g + p_z \dot{\mathcal{P}}_y + \dot{\mathcal{L}}_x}{Mg + \dot{\mathcal{P}}_z} = \frac{M\ddot{y}g + p_z M\ddot{y} + Mx\ddot{y} - My\ddot{x}}{Mg + M\ddot{z}} = y - \frac{(z-p_z)\ddot{y}}{\ddot{z}+g}$$

3. 阅读论文，总结满足ZMP稳定条件的双足步规划方法

Q. Huang *et al.*, "Planning walking patterns for a biped robot," ***IEEE Transactions on Robotics and Automation***, vol. 17, no. 3, pp. 280-289, 2001.

https://ieeexplore.ieee.org/document/938385

（可在Sakai“课程资料”/“作业附件”中下载） 34

Overview: 还是用中文说比较清楚，该方法在给定环境和行走机器人的运动的情况下，首先将行走机器人的hip ankle关节的平面轨迹x z theta约束到仅包含2个变量参数（x_ed x_sd）的曲线族，然后遍历这两个变量参数的取值范围，选择ZMP宽裕度最高、速度最快的一组解，然后利用hip ankle曲线进行逆运动学求解关节角度，控制机器人关节从而实现行走。

机器人的模型固定参数和变量参数如下

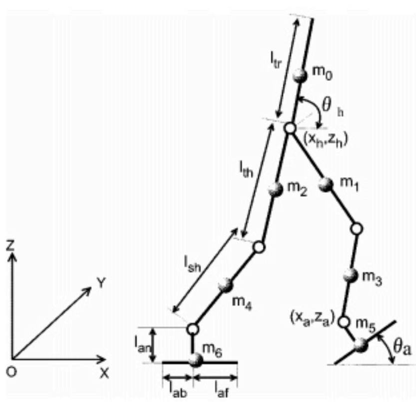


Fig. 1. Model of the biped robot.

fig1所有参数都是机器人结构决定的，fig2的qf qb参数是事先确定不变的，fig2的xed xsd是针对每一步需要搜索确定的。

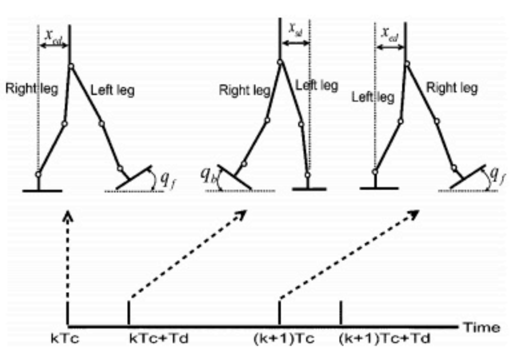


Fig. 2. Walking cycle.

整个算法的流程如下，我们先 claim如下的hip和ankle轨迹：

$$x_{\mathrm{h}}(t)=\begin{cases} kD_{\mathrm{s}}+x_{\mathrm{ed}}, & t=kT_{\mathrm{c}} \\ (k+1)D_{\mathrm{s}}-x_{\mathrm{sd}}, & t=kT_{\mathrm{c}}+T_{\mathrm{d}} \\ (k+1)D_{\mathrm{s}}+x_{\mathrm{ed}}, & t=(k+1)T_{\mathrm{c}}. \end{cases} \qquad z_{\mathrm{h}}(t)=\begin{cases} H_{\mathrm{hmin}}, & t=kT_{\mathrm{c}}+0.5T_{\mathrm{d}} \\ H_{\mathrm{hmax}}, & t=kT_{\mathrm{c}}+0.5(T_{\mathrm{c}}-T_{\mathrm{d}}) \\ H_{\mathrm{hmin}}, & t=(k+1)T_{\mathrm{c}}+0.5T_{\mathrm{d}}. \end{cases}$$

theta hip即躯干的倾斜角度，一直为0即躯干保持竖直。

$$\theta_{\mathrm{a}}(t)=\begin{cases} q_{\mathrm{gs}}(k), & t=kT_{\mathrm{c}} \\ q_{\mathrm{b}}, & t=kT_{\mathrm{c}}+T_{\mathrm{d}} \\ -q_{\mathrm{f}}, & t=(k+1)T_{\mathrm{c}} \\ -q_{\mathrm{ge}}(k), & t=(k+1)T_{\mathrm{c}}+T_{\mathrm{d}} \end{cases}$$

$$x_{\mathrm{a}}(t)=\begin{cases} kD_{\mathrm{s}}, & t=kT_{\mathrm{c}} \\ kD_{\mathrm{s}}+l_{\mathrm{an}}\sin q_{\mathrm{b}}+l_{\mathrm{af}}(1-\cos q_{\mathrm{b}}), & t=kT_{\mathrm{c}}+T_{\mathrm{d}} \\ kD_{\mathrm{s}}+L_{\mathrm{ao}}, & t=kT_{\mathrm{c}}+T_{\mathrm{m}} \\ (k+2)D_{\mathrm{s}}-l_{\mathrm{an}}\sin q_{\mathrm{f}}-l_{\mathrm{ab}}(1-\cos q_{\mathrm{f}}), & t=(k+1)T_{\mathrm{c}} \\ (k+2)D_{\mathrm{s}}, & t=(k+1)T_{\mathrm{c}}+T_{\mathrm{d}} \end{cases}$$

$$z_{\mathrm{a}}(t)=\begin{cases} h_{\mathrm{gs}}(k)+l_{\mathrm{an}}, & t=kT_{\mathrm{c}} \\ h_{\mathrm{gs}}(k)+l_{\mathrm{af}}\sin q_{\mathrm{b}}+l_{\mathrm{an}}\cos q_{\mathrm{b}}, & t=kT_{\mathrm{c}}+T_{\mathrm{d}} \\ H_{\mathrm{ao}}, & t=kT_{\mathrm{c}}+T_{\mathrm{m}} \\ h_{\mathrm{ge}}(k)+l_{\mathrm{ab}}\sin q_{\mathrm{f}}+l_{\mathrm{an}}\cos q_{\mathrm{f}}, & t=(k+1)T_{\mathrm{c}} \\ h_{\mathrm{ge}}(k)+l_{\mathrm{an}}, & t=(k+1)T_{\mathrm{c}}+T_{\mathrm{d}} \end{cases}$$

可以看到，上述的轨迹中，除了qgs（k）qge（k）hgs（k）hge（k）xsd xed外，都是机器人结构的固有参数或实现设定的常数参数。而这些与k相关的参数是由环境确定的，机器人在行走的过程中利用传感器确定。

利用这些关键点，结合速度与位置的关系，可以使用多项式插值，获得运动轨迹。具体的边界条件如下：

$$\begin{cases} \dot{\theta}_{\mathrm{a}}(kT_{\mathrm{c}})=0 \\ \dot{\theta}_{\mathrm{a}}((k+1)T_{\mathrm{c}}+T_{\mathrm{d}})=0 \end{cases} \qquad \begin{cases} \dot{x}_{\mathrm{h}}(kT_{\mathrm{c}})=\dot{x}_{\mathrm{h}}(kT_{\mathrm{c}}+T_{\mathrm{c}}) \\ \ddot{x}_{\mathrm{h}}(kT_{\mathrm{c}})=\ddot{x}_{\mathrm{h}}(kT_{\mathrm{c}}+T_{\mathrm{c}}). \end{cases}$$

$$\begin{cases} \dot{x}_{\mathrm{a}}(kT_{\mathrm{c}})=0 \\ \dot{x}_{\mathrm{a}}((k+1)T_{\mathrm{c}}+T_{\mathrm{d}})=0 \end{cases}$$

$$\begin{cases} \dot{z}_{\mathrm{a}}(kT_{\mathrm{c}})=0 \\ \dot{z}_{\mathrm{a}}((k+1)T_{\mathrm{c}}+T_{\mathrm{d}})=0. \end{cases}$$

利用上述cliam的轨迹我们可以算出给定xsd xed的ZMP，并与支撑多边形比较，算出的宽裕度。xsd和xed从0开始依次增加直到宽裕度满足要求。整体算法的流程如右图所示：