《空间机器人动力学模型》说明

弓铎

2018 年 3 月 7 日

空间机械臂动力学模型，根据 Matlab 语言 SpaceDyn 编写

- Python3.6
- 只完成前向动力学
- 某些变量（构型变量）的存储下标并未从零开始，待标准化
- 关节类型只考虑转动关节（rotational），未处理平动关节
- 使用多维数组时，多次强制进行形状转换（reshape），可能导致不易察觉的错误，待改进

0. 变量说明 (self)

1. 构型相关：（connect 相关变量存储下标存在问题，并未从 0 开始）
   - link_num: 连杆数目 n，编号从 0 到 n-1，其中，0 为基座
   - connect_lower: 低连接，$1 \times n-1$ 数组，connect_lower[i] 表示编号 i+1 连杆的低连接编号
   - connect_upper: 高连接，$n-1 \times n-1$ 数组
   - connect_end,connect_base: 是否与末端、基座相连（1 是，0 否）
2. inertia: 惯量矩阵，$n \times 3 \times 3$ 数组，第一下标表示连杆编号
3. mass: 质量，分别为连杆质量
4. q_from_Bi_to_i: 连杆姿态，为坐标系 $\sum_{Bi}$ 到坐标系 $\sum_i$ 的欧拉角变换
5. link_vector: 连杆矢量，$n+1 \times n+1 \times 3$ 数组，意义如图

1. 方向余弦和坐标变换

1. 坐标系定义：原点固连在关节上，z 轴为关节旋转轴（关节角 q 相关），其余定义可由 self.q_from_Bi_to_i 和 self.link_vector 确定
2. 方向余弦，采用如下定义:

$$\begin{aligned}\left\{\sum\nolimits_i\right\} &= {}^iC_{i-1}\left\{\sum\nolimits_{i-1}\right\} \\ &= [C_3(q_i)C_3(\gamma_i)C_2(\beta_i)C_1(\alpha_i)]^T\left\{\sum\nolimits_{i-1}\right\}\end{aligned}$$

3. 坐标变换：${}^{I}A_i = C_i^T$

2. 运动学

1. 计算坐标变换 $A = clac\_coordinate\_transform(A_0, q)$

$$
\begin{aligned}
A_i &= A_{B_i} \cdot {}^{i}C_{B_i}^T \\
&= A_{B_i} \cdot (C_z(\gamma + q_i) C_y(\beta) C_x(\alpha))^T
\end{aligned}
$$

2. 计算连杆位置 $R = calc\_position(R_0, A, q)$

$$
R_i = R_{B_i} + A_{B_i} \cdot l_{B_i,i} - A_i \cdot l_{i,i}
$$

3. 计算连杆速度 $v, \omega = calc\_velocity(A, v_0, \omega_0, q, \dot{q})$

$$
\begin{aligned}
\omega_i &= \omega_{B_i} + (A_i \cdot e_z)\dot{q}_i \\
v_i &= v_{B_i} + \omega_{B_i} \times (A_{B_i} \cdot l_{B_i,i}) - \omega_i \times (A_i \cdot l_{i,i})
\end{aligned}
$$

4. 计算连杆加速度

3. 前向动力学

系统动力学方程写作：

$$
\begin{bmatrix} \boldsymbol{H}_b & \boldsymbol{H}_{bm} \\ \boldsymbol{H}_{bm}^T & \boldsymbol{H}_m \end{bmatrix} \begin{bmatrix} \ddot{x}_b \\ \ddot{\boldsymbol{\Theta}} \end{bmatrix} + \begin{bmatrix} c_b \\ c_m \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_b \\ \boldsymbol{\tau}_m \end{bmatrix}
$$

1. 计算平动雅克比矩阵 $J_t = calc\_translational\_jacobian(R, A)$

$$
\begin{aligned}
J_{ti} &= [k_1 \times (r_i - p_1), \cdots, k_i \times (r_i - p_i), 0, \cdots, 0] \in \mathbb{R}^{3\times n} \\
&= (A_j \cdot e_z) \times (R_i - R_j - (A_j \cdot l_{j,j}))
\end{aligned}
$$

(i,j 详细计算见程序)

2. 计算转动雅克比矩阵 $J_r = calc\_rotational\_jacobian(R, A)$

$$
\begin{aligned}
J_{ri} &= [k_1, \cdots, k_i, 0, \cdots, 0] \in \mathbb{R}^{3\times n} \\
&= (A_j \cdot e_z)
\end{aligned}
$$

(i,j 详细计算见程序)

3. 计算惯量矩阵 $H = clalc\_inertia\_matrices(R, A)$

$$
\begin{aligned}
J_t &= calc\_translational\_jacobian(R, A) \\
J_r &= calc\_rotational\_jacobian(R, A) \\
\omega E &= ME_{3\times 3} \\
\omega r_{0g} &= (R_g - R_0)M \\
J_{tg} &= \sum m_i J_{ti} \\
H_\omega &= \sum A_i \cdot inertia_i \cdot A_i^T + m_i \tilde{r}_{0i}^T \cdot \tilde{r}_{0i} \\
H_{\omega q} &= \sum (A_i \cdot inertia_i \cdot A_i^T) \cdot J_{ri} + m_i \tilde{r}_{0i} \cdot J_{ti} \\
H_q &= J_{ri}^T \cdot A_i \cdot inertia_i \cdot A_i^T \cdot J_{ri} + m_i J_{ti}^T J_{ti}
\end{aligned}
$$

$$
\boldsymbol{H} = \begin{bmatrix} \omega E & \tilde{\omega r}_{0g}^T & J_{tg} \\ \tilde{\omega r}_{0g} & H_\omega & H_{\omega q} \\ J_{tg}^T & H_{\omega q}^T & H_q \end{bmatrix}
$$

a b