五连杆并联机构大轮子

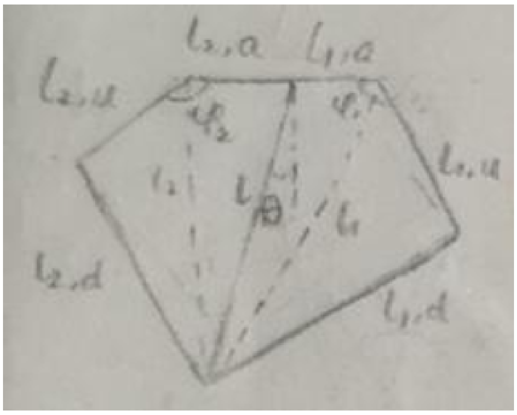
正运动学解算

```
%闭链五杆的正解，根据电机的角度求足端末点的坐标
%u1为l1从x轴顺时针转的角度，为负值；u4为l4从x轴逆时针转的角度
function [x, y, u2, u3] = Zjie(u1, u4, pitch)%选择以l5/2处为零点进行建模
global l1 l2 l3 l4 l5;
xb = l1 * cos(u1) - l5/2;
yb = l1 * sin(u1);
xd = l5/2 + l4 * cos(u4);
yd = l4 * sin(u4);
lbd = sqrt((xd - xb).^2 + (yd - yb).^2);
A0 = 2 * l2 * (xd - xb);
B0 = 2 * l2 * (yd - yb);
C0 = l2.^2 + lbd.^2 - l3.^2;
D0 = l3.^2 + lbd.^2 - l2.^2;
u2 = 2 * atan((B0 + sqrt(A0.^2 + B0.^2 - C0.^2))/(A0 + C0));%rad\
u3 = pi - 2 * atan((-B0 + sqrt(A0.^2 + B0.^2 - D0.^2))/(A0 + D0));
xc = xb + l2 * cos(u2);
yc = yb + l2 * sin(u2);
R = [cos(pitch), -sin(pitch);
    sin(pitch), cos(pitch)];
v = R*[xc;yc];
x = v(1);
y = v(2);
end
```

逆运动学解算

这部分比较简单用三角形特性求，直接在用c++写

1 参数定义



2 逆运动学

逆运动学问题：$(\varphi_1, \varphi_2) = f(l, \theta)$

$$l_1 = \sqrt{l_{1,a}^2 + l^2 + 2l_{1,a}lsin\theta} \quad (2.1)$$

$$l_2 = \sqrt{l_{2,a}^2 + l^2 - 2l_{2,a}lsin\theta} \quad (2.2)$$

$$\varphi_1 = acos\frac{l_1^2 + l_{1,a}^2 - l^2}{2l_1l_{1,a}} + acos\frac{l_1^2 + l_{1,u}^2 - l_{1,d}^2}{2l_1l_{1,u}} \quad (2.3)$$

$$\varphi_2 = acos\frac{l_2^2 + l_{2,a}^2 - l^2}{2l_2l_{2,a}} + acos\frac{l_2^2 + l_{2,u}^2 - l_{2,d}^2}{2l_2l_{2,u}} \quad (2.4)$$

```
float calculate_l1(float l1_a, float l, float theta) {
    return sqrt(l1_a * l1_a + l * l + 2 * l1_a * l * sin(theta));
}

float calculate_l2(float l2_a, float l, float theta) {
    return sqrt(l2_a * l2_a + l * l - 2 * l2_a * l * sin(theta));
}

float calculate_phi1(float l1, float l1_a, float l, float l1_u, float l1_d) {
      float phi1 = acos((l1 * l1 + l1_a * l1_a - l * l) / (2 * l1 * l1_a)) +
                   acos((l1 * l1 + l1_u * l1_u - l1_d * l1_d) / (2 * l1 * l1_u));
      return M_PI - phi1;
}

float calculate_phi2(float l2, float l2_a, float l, float l2_u, float l2_d) {
    return  acos((l2 * l2 + l2_a * l2_a - l * l) / (2 * l2 * l2_a)) +
```

```
                    acos((l2 * l2 + l2_u * l2_u - l2_d * l2_d) / (2 * l2 *
l2_u));
}
```

动力学解算

```
function [K, L_sum] = model_LQR(xc, yc, xp, yp, Ipin)
syms theta(t) x(t) phi(t) T(t) Tp(t) N(t) P(t) Nm(t) Pm(t)
syms theta_dot x_dot phi_dot x_dot_dot theta_dot_dot phi_dot_dot
%进行参数的赋值——都转化成国际单位制
L_sum = sqrt(xc ^ 2 + yc ^ 2);
L_now = sqrt((xc - xp)^2 + (yc - yp)^2);
Lm_now = L_sum - L_now;
R = 0.05;
L = L_now * 0.001;
Lm = Lm_now * 0.001;
l = 0;
mw = 3 * 2;
mp = 0.808 * 2;
M = 10;
Iw = 0.001 * 2;
Ip = Ipin;
Im = 0.78714333 * 0.062665311;
g = 9.81;
% subs(A_final, ...
%     [R, L, Lm, l, mw, mp, M, Iw, Ip, Im], ...
%     [0.05, L_now, Lm_now, 0,  0.406, 0.808, 15.245, 0.000508589, Ipin,
0.062665311]);%R=50mm;
% subs(B_final, ...
%     [R, L, Lm, l, mw, mp, M, Iw, Ip, Im], ...
%     [0.05, L_now, Lm_now, 0,  0.406, 0.808, 15.245, 0.000508589, Ipin,
0.062665311]);%R=50mm;
%列出替代方程
Nm(t) = M*diff(diff(x+(L+Lm)*sin(theta)-l*sin(phi),t) , t);
Pm(t)= M*g + M*diff(diff((L+Lm)*cos(theta)+l*cos(phi),t), t);
N(t) = Nm(t) + mp*diff(diff(x + L*sin(theta), t), t);
P(t) = Pm(t) + mp*g + mp*diff(diff(L*cos(theta), t), t);
%简化，用字母替代
Nm(t) = (subs(Nm(t), ...
    [diff(x, t, t), diff(theta, t, t), diff(phi,t,t)], ...
    [x_dot_dot, theta_dot_dot, phi_dot_dot]));
Nm(t) = vpa(subs(Nm(t), ...
    [diff(x, t), diff(theta, t), diff(phi, t)], ...
    [x_dot theta_dot, phi_dot]));
Pm(t) = subs(Pm(t), ...
```

```
    [diff(x, t, t), diff(theta, t, t), diff(phi,t,t)], ...
    [x_dot_dot, theta_dot_dot, phi_dot_dot]);
Pm(t) = vpa(subs(Pm(t), ...
    [diff(x, t), diff(theta, t), diff(phi, t)], ...
    [x_dot theta_dot, phi_dot]));
N(t) = subs(N(t), ...
    [diff(x, t, t), diff(theta, t, t), diff(phi,t,t)], ...
    [x_dot_dot, theta_dot_dot, phi_dot_dot]);
N(t) = vpa(subs(N(t), ...
    [diff(x, t), diff(theta, t), diff(phi, t)], ...
    [x_dot theta_dot, phi_dot]));
P(t) = subs(P(t), ...
    [diff(x, t, t), diff(theta, t, t), diff(phi,t,t)], ...
    [x_dot_dot, theta_dot_dot, phi_dot_dot]);
P(t) = vpa(subs(P(t), ...
    [diff(x, t), diff(theta, t), diff(phi, t)], ...
    [x_dot theta_dot, phi_dot]));
eqns = [x_dot_dot == (T-N(t)*R)/(Iw/R + mw*R);
    Ip*theta_dot_dot == (P(t)*L+Pm(t)*Lm) * sin(theta)-(N(t)*L+ Nm(t)*Lm) * cos(theta)- T(t) + Tp(t);
    Im*phi_dot_dot == Tp(t)+Nm*l*cos(phi)+Pm*l*sin(phi)];
vars = [x_dot_dot theta_dot_dot phi_dot_dot];
[x_dot_dot, theta_dot_dot, phi_dot_dot] = solve(eqns, vars);
%对平衡点进行线性化
X = [theta(t); theta_dot; x(t); x_dot; phi(t); phi_dot];
u = [T(t);Tp(t)];
X_dot = [theta_dot; theta_dot_dot; x_dot; x_dot_dot; phi_dot; phi_dot_dot];
A = jacobian(X_dot, X);
B = jacobian(X_dot, u);
A = subs(A, [theta(t), theta_dot, x_dot, phi(t), phi_dot, T(t), Tp(t)], zeros(1,7));%代入平衡点处的值, X = [0 0 x 0 0 0] u = [0 0]
B = subs(B, [theta(t), theta_dot, x_dot, phi(t), phi_dot, T(t), Tp(t)], zeros(1,7));
A = double(A);
B = double(B);
size(A,1);
%算可控矩阵的秩，如果等于A的阶数，则系统可控
if(rank(ctrb(A, B)) == size(A, 1))
    disp('系统可控');
else
    disp('系统不可控');
end
C = eye(6);
D = zeros(6,2);
v_Q = [500 200 500 200 1000 500];
Q = diag(v_Q);
v_R = [1 0.25];
```

```
R = diag(v_R);
sys = ss(A, B, C, D);
K = lqr(sys, Q, R)%得到反馈增益矩阵
end
```

VMC解算

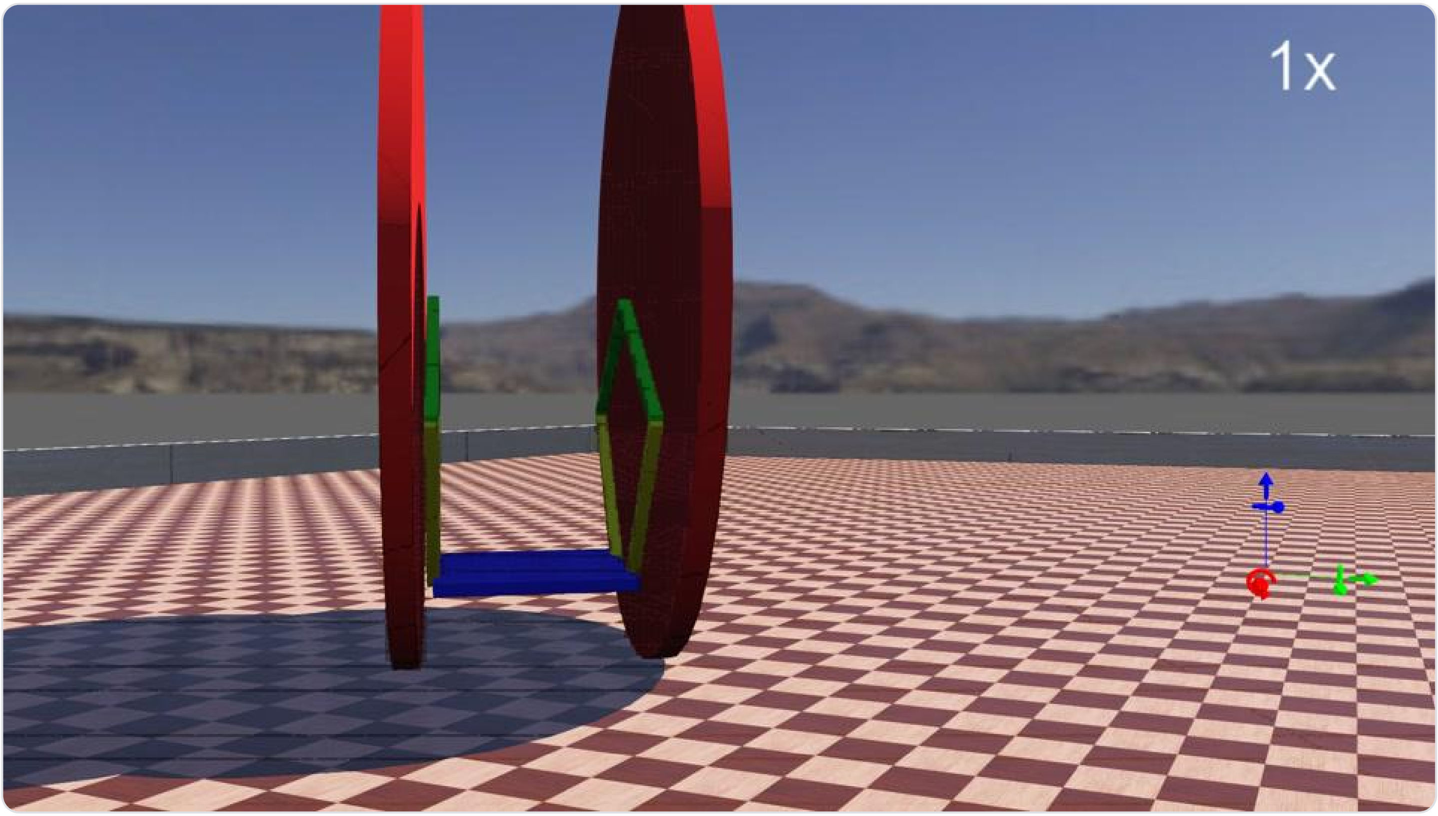
```
function [T1, T2] = VMC(F, Tp)
syms phi0(t) phi1(t) phi2(t) phi3(t) phi4(t) phi_dot_1 phi_dot_4
syms l0 l1 l2 l3 l4 l5 theta0 theta1 theta2 theta3 theta4;
x_B = l1*cos(phi1);
y_B = l1*sin(phi1);
x_C = x_B+l2*cos(phi2);
y_C = y_B+l2*sin(phi2);
x_D = l5+l4*cos(phi4);
y_D = l4*sin(phi4);
x_dot_B = diff(x_B,t);
y_dot_B = diff(y_B,t);
x_dot_C = diff(x_C,t);
y_dot_C = diff(y_C,t);
x_dot_D = diff(x_D,t);
y_dot_D = diff(y_D,t);
phi_dot_2 = ((x_dot_D-x_dot_B)*cos(phi3)+(y_dot_D-y_dot_B)*sin(phi3))/l2/sin(phi3-phi2);
x_dot_C = subs(x_dot_C,diff(phi2,t),phi_dot_2);
x_dot_C = subs(x_dot_C,...
    [diff(phi1,t),diff(phi4,t)],...
    [phi_dot_1,phi_dot_4])
y_dot_C = subs(y_dot_C,diff(phi2,t),phi_dot_2);
y_dot_C = subs(y_dot_C,...
    [diff(phi1,t),diff(phi4,t)],...
    [phi_dot_1,phi_dot_4])
x_dot = [x_dot_C; y_dot_C];
q_dot = [phi_dot_1; phi_dot_4];
x_dot = simplify(collect(x_dot,q_dot));%如果 x_dot 是一个多项式，其中包含多个项，每个项都是由 q_dot 中的变量乘以一个系数得到的。那么，使用 collect(x_dot,q_dot) 将把这些项按照它们所乘以的变量进行分组。然后，使用 simplify 函数可以进一步化简结果。
J = simplify(jacobian(x_dot,q_dot))%simplify是用来化简符号表达式的（很好用，可以把公式简化成人能看懂的排列）
R = [cos(-phi0) -sin(-phi0);
      sin(-phi0)  cos(-phi0)];
M = [0     1/l0;
      -1 0];
T = simplify(J.'*R*M)
% symdisp(T)
```

```
T = subs(T, ...
    [phi0(t), phi1(t), phi2(t), phi3(t), phi4(t)], ...
    [theta0, theta1, theta2, theta3, theta4]);
torque = (T)*[F;Tp];
T1 = torque(1);
T2 = torque(2);
end
```

gitlab仿真代码

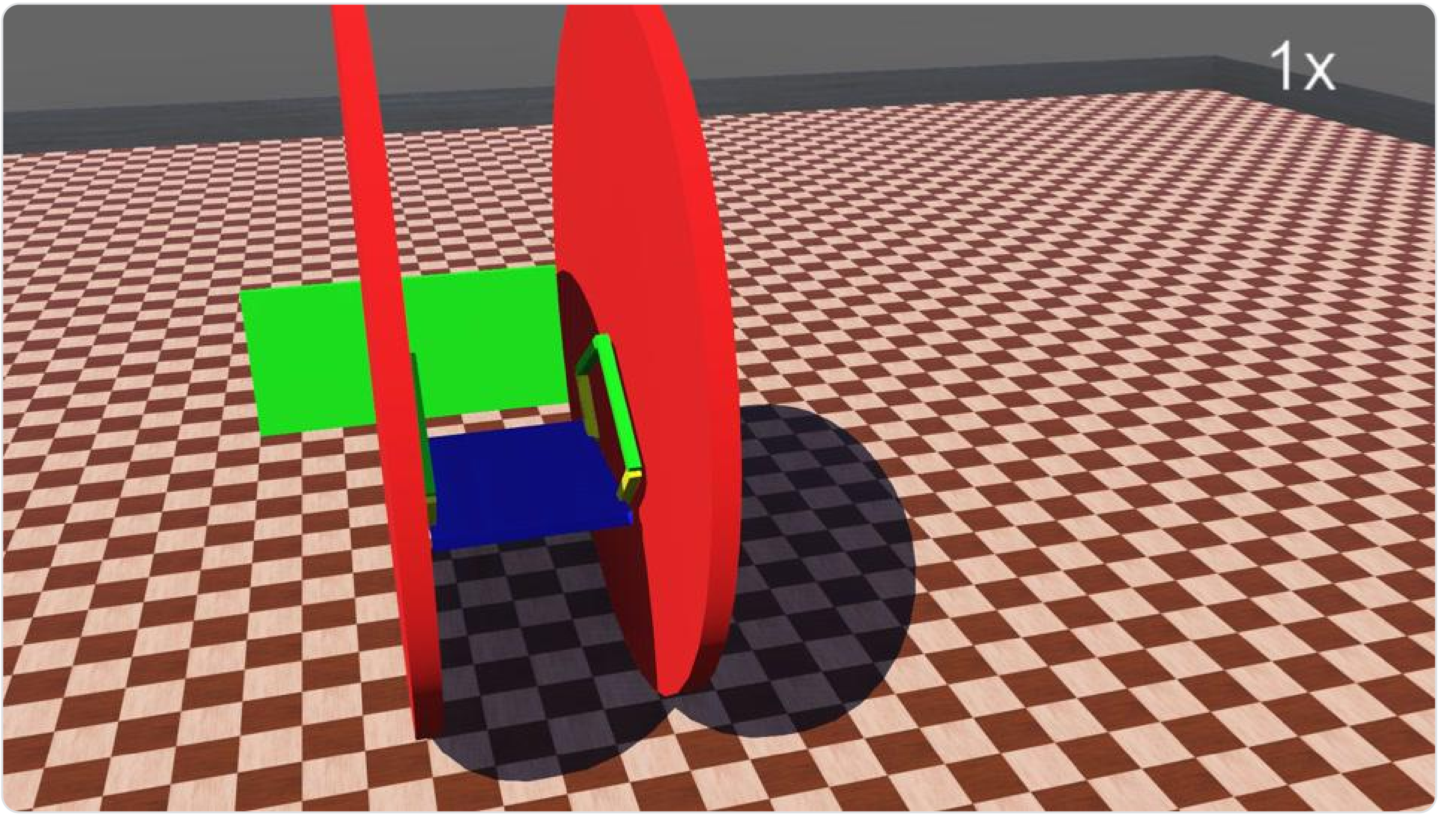
正立摆长连杆代码：https://git.ddt.dev:9281/rbt/alg/five_bar_link/-/tree/big_wheel_45?ref_type=heads

对应的wbt文件：inverted_balance.wbt



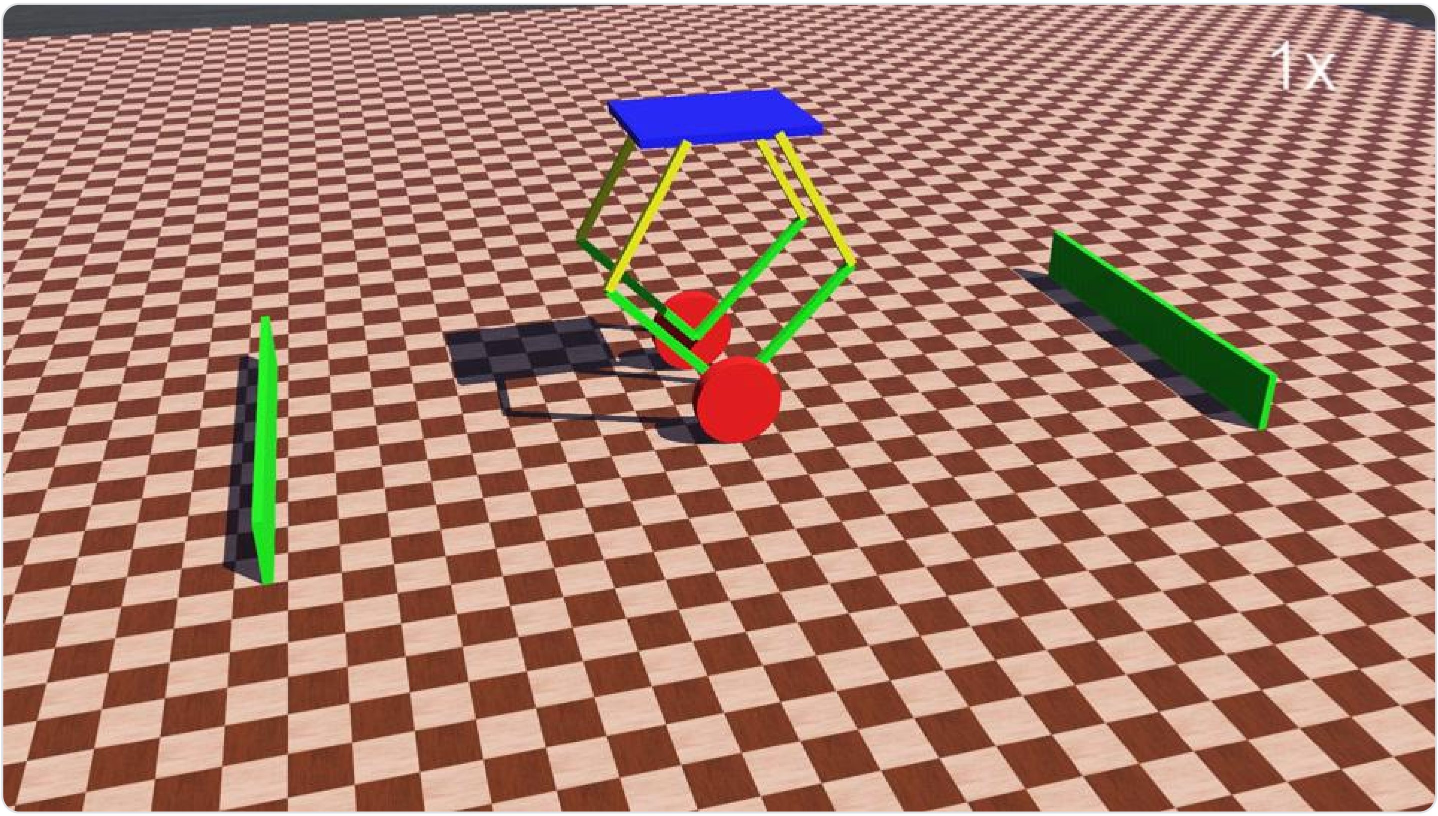
正立摆短连杆代码：https://git.ddt.dev:9281/rbt/alg/five_bar_link/-/tree/short_link?ref_type=heads

对应的wbt文件：short_link.wbt



倒立摆代码：https://git.ddt.dev:9281/rbt/alg/five_bar_link/-/tree/handstand?ref_type=heads

对应的wbt文件：five_bar_link copy.wbt



总结：

1. 长连杆下无论是倒立摆还是正立摆效果都比较好控制且比较稳定，连杆变短难控制且不稳定，实机效果和仿真效果大致一样。

2. 个人分析原因：连杆变短，腿长变化是对应的电机角度变大，容易出现超调的情况，且腿长精度难以控制。

参考文献

五连杆运动学解算与VMC

RoboMaster平衡步兵机器人控制系统设计