**Specifications**

Given:

Cost functions:

Constraints:

First attempt:

* Iterated until
* Used sequential quadratic programming with a variety of initial conditions
* Confirmed with generalized reduced gradient method

**Implementation in Matlab**

Design variables: x = [qf1, qf2, qf3]’

Cost function

d = 0.5;

t = 0:0.001:d;

[q, qdot, qddot] = min\_jerk(0, x, d, t);

M = I\*qddot + D\*qdot + K\*q + G

C\_MW = trapz(q(1,:), M(1,:)) + trapz(q(2,:), M(2,:)) + trapz(q(3,:), M(3,:));

f(x) = C\_MW;

Initial guess: x0 = [qi1,qi2,qi3]’, where

qi1 can be obtained from Fig. 6

qi3 = asin(sin(qi1)\*x – cos(qi1)\*y)

qi2 = asin( (sin(qi1)\*sin(qi3) – x) / (cos(qi1)\*cos(qi3)) )

x = sin(pi/12)\*cos(theta)

y = sin(pi/12)\*sin(theta)

theta = 90 deg

Bounds on x: lb = [-30, -30, -80]’\*pi/180; ub = [30, 30, 80]’\*pi/180

Nonlinear inequalities: none, i.e. c(x) is not needed

Nonlinear equalities:

xs = -cos(x(1))\*sin(x(2))\*cos(x(3)) + sin(x(1))\*sin(x(3))

ys = -sin(x(1))\*sin(x(2))\*cos(x(3)) - cos(x(1))\*sin(x(3))

ceq(x) = [abs(xs – (-cos(x(1))\*sin(x(2))\*cos(x(3)) + sin(x(1))\*sin(x(3)));

abs(ys – (-sin(x(1))\*sin(x(2))\*cos(x(3)) - cos(x(1))\*sin(x(3)))]

Iterate until difference is less than or equal to 0.001