RSUR Geometric Method

A = revolute R B = spherical S C = universal U D = revolute R

**z**

**y**

**x**

**a**

**b**

**c**

**d**

**C**

**D**

**A**

**B**

**O**

**e**

****

****

a = OA = ground (20.43 cm)

b = AB = input crank (4.00 cm)

 in y-z plane

c = CD = follower (10.00 cm)

 in x-y plane

d = OD = ground (19.97 cm)

e = BC = coupler (30.42 cm)

 = crank angle in y-z plane

= follower angle in x-y plane

**Position solution** - 























**Velocity solution** - 













**Acceleration solution** - 













**Jerk solution** - 































**Snap solution** - 













































% rsur\_geometric.m - geometric solution for RSUR mechanism

% HJSIII, 19.04.13

clear

% constants

d2r = pi / 180;

% mechanism dimensions - units = [cm]

a = 20.43;

b = 4.00;

c = 10.00;

d = 19.97;

e = 30.42;

% constant 60 rpm CCW = 1 rev/sec = 2 pi rad/sec

w2 = 60 \* 2\*pi /60; % rad/sec

th\_start = 0;

% time for one revolution

tend = 1 / (w2/2/pi);

% 180 steps

dt = tend / 90;

% allocate space to save results

keep = [];

% time loop

for t = 0 : dt : tend;

% constant speed rotation

 th = th\_start + w2\*t;

 thd = w2;

 thdd = 0;

 thddd = 0;

 thdddd = 0;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% geometric solution

% position

 f = 2\*c\*d;

 g = 2\*b\*c\*sin(th);

 h = e\*e - a\*a - b\*b - c\*c - d\*d + 2\*a\*b\*cos(th);

 u1 = (-g - sqrt( f\*f + g\*g - h\*h )) / (h+f);

 u2 = (-g + sqrt( f\*f + g\*g - h\*h )) / (h+f);

 phi = 2\*atan(u1);

 phi2 = 2\*atan(u2); % alternate assembly configuration

% general terms

 A = c\*d\*sin(phi) +b\*c\*sin(th)\*cos(phi);

 B = a\*b\*sin(th) -b\*c\*cos(th)\*sin(phi);

 C = a\*b\*cos(th) +b\*c\*sin(th)\*sin(phi);

 D = -c\*d\*cos(phi) +b\*c\*sin(th)\*sin(phi);

 E = b\*c\*cos(th)\*cos(phi);

 F = b\*c\*sin(th)\*cos(phi);

 G = b\*c\*cos(th)\*sin(phi);

 H = b\*c\*sin(th)\*sin(phi);

% velocity

 phid = B \* thd /A;

% acceleration

 phidd = ( B\*thdd +C\*thd\*thd +D\*phid\*phid -2\*E\*thd\*phid ) /A;

% jerk

 phiddd = ( B\*(thddd-thd\*thd\*thd) +3\*C\*thd\*thdd +3\*D\*phid\*phidd ...

 +A\*phid\*phid\*phid -3\*E\*(thdd\*phid+thd\*phidd) ...

 +3\*thd\*phid\*(F\*thd+G\*phid) ) /A;

% snap

 phidddd = ( B\*(thdddd-6\*thd\*thd\*thdd) +C\*(4\*thddd\*thd+3\*thdd\*thdd-thd\*thd\*thd\*thd) ...

 +D\*(4\*phiddd\*phid +3\*phidd\*phidd -phid\*phid\*phid\*phid) +6\*A\*phid\*phid\*phidd ...

 -2\*E\*(2\*thddd\*phid +3\*thdd\*phidd -2\*thd\*thd\*thd\*phid ...

 +thd\*phid\*phid\*phid +2\*thd\*phiddd) ...

 +6\*F\*(2\*thd\*thdd\*phid +thd\*thd\*phidd) ...

 +6\*G\*(thdd\*phid\*phid +2\*thd\*phid\*phidd) -6\*H\*thd\*thd\*phid\*phid ) /A;

% save results

 keep = [ keep ; th thd thdd thddd thdddd phi phid phidd phiddd phidddd ];

end

% values for plotting

th = keep(:,1);

th\_deg = th /d2r;

thd = keep(:,2);

thdd = keep(:,3);

thddd = keep(:,4);

thdddd = keep(:,5);

phi = keep(:,6);

phi\_deg = phi /d2r;

phid = keep(:,7);

phidd = keep(:,8);

phiddd = keep(:,9);

phidddd = keep(:,10);

% simple finite difference approximations

phid\_fd = ( [diff(phi);NaN] + [NaN;diff(phi)] ) /2 /dt;

phidd\_fd = ( [diff(phid);NaN] + [NaN;diff(phid)] ) /2 /dt;

phiddd\_fd = ( [diff(phidd);NaN] + [NaN;diff(phidd)] ) /2 /dt;

phidddd\_fd = ( [diff(phiddd);NaN] + [NaN;diff(phiddd)] ) /2 /dt;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% plot position solution

figure( 1 )

 clf

 plot( th\_deg,phi\_deg,'r' )

 title( 'RSUR - crank speed 60 rpm CCW' )

 xlabel( 'Theta [deg]' )

 ylabel( 'Phi [deg]' )

 axis( [ 0 360 -120 -60 ] )

% plot velocity solution

figure( 2 )

 clf

 plot( th\_deg,phid,'r', th\_deg,phid\_fd,'go' )

 title( 'RSUR - crank speed 60 rpm CCW' )

 xlabel( 'Theta [deg]' )

 ylabel( 'Phi velocity [rad/s]' )

% axis( [ 0 360 -3 3 ] )

 legend( 'Geometric', 'Finite difference' )

% plot acceleration solution

figure( 3 )

 clf

 plot( th\_deg,phidd,'r', th\_deg,phidd\_fd,'go' )

 title( 'RSUR - crank speed 60 rpm CCW' )

 xlabel( 'Theta [deg]' )

 ylabel( 'Phi acceleration [rad/s/s]' )

% axis( [ 0 360 -20 20 ] )

 legend( 'Geometric', 'Finite difference' )

% plot jerk solution

figure( 4 )

 clf

 plot( th\_deg,phiddd,'r', th\_deg,phiddd\_fd,'go' )

 title( 'RSUR - crank speed 60 rpm CCW' )

 xlabel( 'Theta [deg]' )

 ylabel( 'Phi jerk [rad/s/s/s]' )

% axis( [ 0 360 -150 150 ] )

 legend( 'Geometric', 'Finite difference' )

% plot snap solution

figure( 5 )

 clf

 plot( th\_deg,phidddd,'r', th\_deg,phidddd\_fd,'go' )

 title( 'RSUR - crank speed 60 rpm CCW' )

 xlabel( 'Theta [deg]' )

 ylabel( 'Phi snap [rad/s/s/s/s]' )

% axis( [ 0 360 -150 150 ] )

 legend( 'Geometric', 'Finite difference' )

% 3D sketch

n = length( th );

rA = [ 0 0 a ] ;

rB = [ zeros(n,1) b\*sin(th) a-b\*cos(th) ] ;

rC = [ d+c\*cos(phi) c\*sin(phi) zeros(n,1) ] ;

rD = [ d 0 0 ] ;

figure( 6 )

 clf

 plot3( -rB(:,3), -rB(:,1), rB(:,2), 'r' )

 hold on

 plot3( -rC(:,3), -rC(:,1), rC(:,2), 'g' )

 xlabel( 'Z' )

 ylabel( 'X' )

 zlabel( 'Y' )

 ax = [ 0 0 0 ;

 rD ; % X axis

 NaN NaN NaN ;

 0 0 0 ;

 0 10 0 ; % Y axis

 NaN NaN NaN ;

 0 0 0 ;

 rA ]; % Z axis

 plot3( -ax(:,3), -ax(:,1), ax(:,2), 'k' )

 text( -ax(2,3), -ax(2,1), ax(2,2), 'D' )

 text( -ax(5,3), -ax(5,1), ax(5,2), 'y', 'VerticalAlignment', 'bottom' )

 text( -ax(8,3), -ax(8,1), ax(8,2), 'A', 'HorizontalAlignment', 'right' )

 axis equal

 nskip = 10;

 for i = 1 : nskip : n,

 plot3( [ -rB(i,3) -rC(i,3) ], [ -rB(i,1) -rC(i,1) ], [ rB(i,2) rC(i,2) ], 'b' )

 end

% swivel angle for S joint

sCwrtB = rC - rB;

sBwrtA = rB - ones(n,1)\*rA;

swivel\_ang = 90 - acos( diag( sBwrtA\*sCwrtB' )/b/e )/d2r;

% bottom of rsur\_geometric

**Spherical Four Bar Geometric**

**A**

**3**

**2**

**4**

**1**

****

****

**B**

**C**

**D**

**2**

**3**

**4**

****

****

revolutes A B C D all intersect at same point

i = angle subtended by link i

 = angle from plane for revolutes A D to center line for link 2

 = angle from plane for revolutes A D to center line for link 4

 = angle from center line for link 2 to center line for link 3

 = angle from center line for link 3 to center line for link 4

tangent of half-angle solution



test values produce phi = 90 deg when theta = 90 deg



% t\_sfb.m - test spherical four bar

% HJSIII, 20.04.28

clear

% constatnts

d2r = pi / 180;

% mechanism constants

r = 4; % radius [inch]

a1 = 90 \* d2r;

a2 = 15 \* d2r;

a3 = 78.5644 \* d2r; % calculated for a1=90, theta=90, phi=90

a4 = 50 \* d2r;

% test for a1 = 90 deg

theta = 90 \* d2r;

phi = 90 \* d2r;

rA = [ 0 0 r ]';

rB = [ 0 r\*sin(a2) r\*cos(a2) ]';

rC = [ r\*cos(a4) r\*sin(a4) 0 ]';

rD = [ r 0 0 ]';

dBC = norm( rC - rB );

a3\_calc1 = acos( 1 - dBC^2/(2\*r\*r) );

a3\_calc2 = 2 \* asin( dBC /2 /r );

% rotate theta and calculate phi

theta = ( 0 : 360 )' \* d2r;

U = sin(a2)\*sin(a4)\*sin(theta);

V = cos(a2)\*sin(a4)\*sin(a1) - sin(a2)\*sin(a4)\*cos(a1)\*cos(theta);

W = sin(a2)\*cos(a4)\*sin(a1)\*cos(theta) + cos(a2)\*cos(a4)\*cos(a1) - cos(a3);

phi = 2 \* atan( (-U - sqrt( U.\*U + V.\*V - W.\*W )) ./ (W-V) );

phi\_alt = 2 \* atan( (-U + sqrt( U.\*U + V.\*V - W.\*W )) ./ (W-V) );

% 2002 Cervantes-Sanchez - beta ????

beta = acos( (sin(a4)\*sin(a1)\*cos(phi) + cos(a4)\*cos(a1) - cos(a2)\*cos(a3)) / (sin(a2)\*sin(a3)) );

gamma = acos( (sin(a2)\*sin(a1)\*cos(theta) + cos(a2)\*cos(a1) - cos(a3)\*cos(a4)) / (sin(a3)\*sin(a4)) );

% 2018 Sun - theta2 = beta, theta4 = phi

X1 = sin(a2)\*sin(theta);

Y1 = -sin(a1)\*cos(a2) + cos(a1)\*sin(a2)\*cos(theta);

Z1 = cos(a1)\*cos(a2) + sin(a1)\*sin(a2)\*cos(theta);

theta4 = 2\*atan2( Z1\*cos(a4)-Y1\*sin(a4)-cos(a3) , ...

 -X1\*sin(a4)+sqrt( (sin(a4)\*sin(a3))^2 - (Z1-cos(a4)\*cos(a3)).^2 ) );

X4 = sin(a4)\*sin(theta4);

Y4 = -sin(a1)\*cos(a4) - cos(a1)\*sin(a4)\*cos(theta4);

Z4 = cos(a1)\*cos(a4) - sin(a1)\*sin(a4)\*cos(theta4);

theta2 = pi - 2\*atan2( cos(a2)\*(X4.\*sin(theta)-Y4.\*cos(theta)) - Z4\*sin(a2) + sin(a3) , ...

 -X4.\*cos(theta) - Y4.\*sin(theta) );

beta = theta2;

% show results

figure( 1 )

 clf

 plot( theta/d2r,phi/d2r,'r', theta/d2r,phi\_alt/d2r,'g', ...

 theta/d2r,beta/d2r,'b', theta/d2r,gamma/d2r,'k' )

 xlabel( 'theta [deg]' )

 ylabel( 'phi, alt phi, beta, gamma [deg]' )

 title( 'alpha1=90, alpha2=15, alpha3=78.5644, alpha4=50 [deg]' )

 axis( [ 0 360 -150 150 ] )

 legend( 'phi', 'alt phi', 'beta', 'gamma' )

% bottom - t\_sfb