

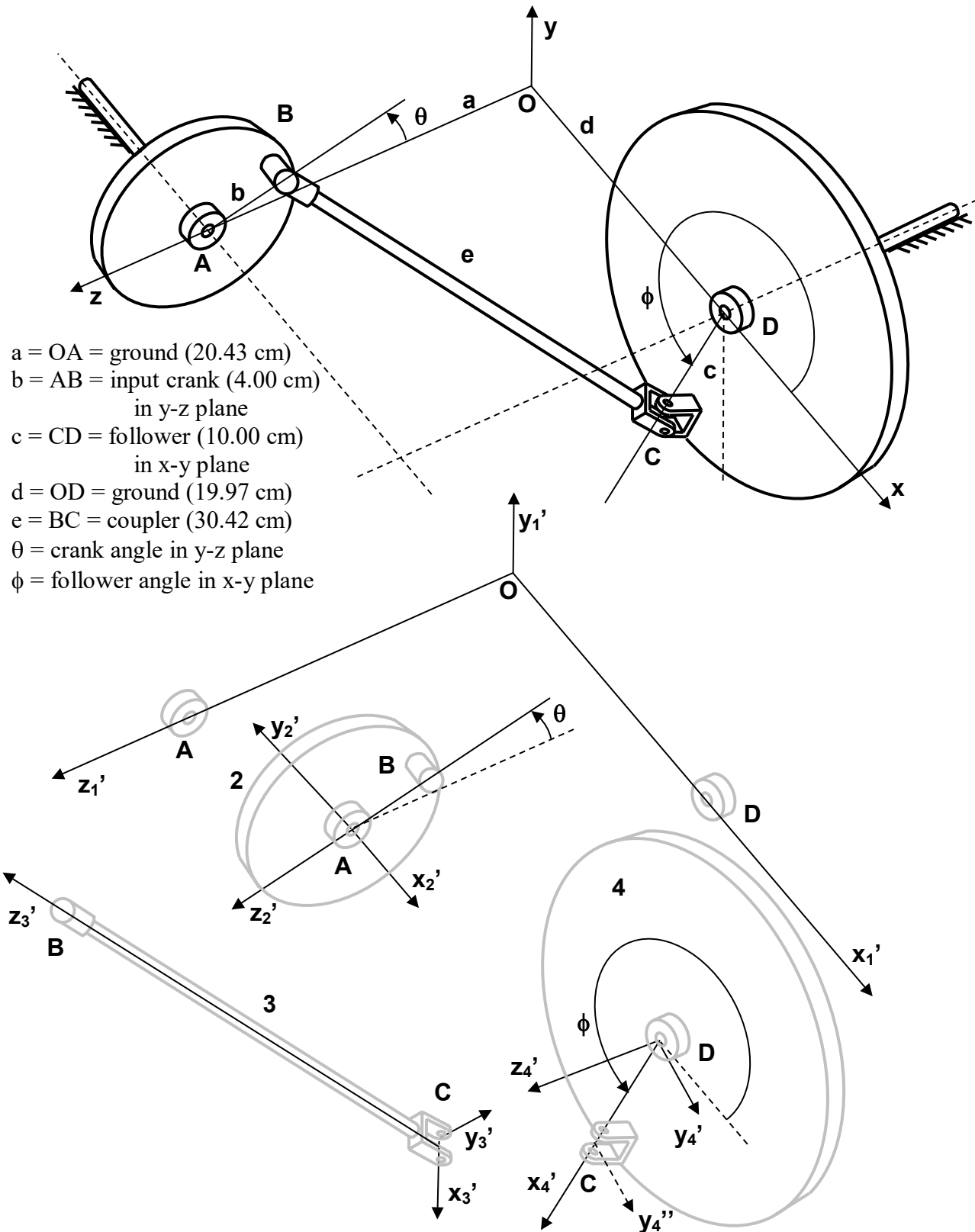
RSUR Generalized Coordinates

A = revolute R

B = spherical S

C = universal U

D = revolute R



CONSTRAINTS

$$\begin{aligned}
\{\mathbf{q}\}_{21 \times 1} &= \begin{Bmatrix} \{\mathbf{r}_2\} \\ \{\mathbf{p}_2\} \\ \{\mathbf{r}_3\} \\ \{\mathbf{p}_3\} \\ \{\mathbf{r}_4\} \\ \{\mathbf{p}_4\} \end{Bmatrix} \quad \{\Phi\}_{21 \times 1} = \begin{Bmatrix} \{\mathbf{r}_1\}^A - \{\mathbf{r}_2\}^A & \text{revA} & A, \mathbf{r}_1 - \mathbf{r}_2, i=2, j=1 \\ \{\hat{\mathbf{g}}_2\}^T \{\hat{\mathbf{f}}_1\} & \mathbf{y}_2' \perp \mathbf{x}_1' & \mathbf{g}_2 \mathbf{f}_1, i=2, j=1 \\ \{\hat{\mathbf{h}}_2\}^T \{\hat{\mathbf{f}}_1\} & \mathbf{z}_2' \perp \mathbf{x}_1' & \mathbf{h}_2 \mathbf{f}_1, i=2, j=1 \\ \{\mathbf{r}_2\}^B - \{\mathbf{r}_3\}^B & \text{sphB} & B, \mathbf{r}_2 - \mathbf{r}_3, i=3, j=2 \\ \{\mathbf{r}_3\}^C - \{\mathbf{r}_4\}^C & \text{univC} & C, \mathbf{r}_3 - \mathbf{r}_4, i=4, j=3 \\ \{\hat{\mathbf{f}}_4\}^T \{\hat{\mathbf{f}}_3\} & \mathbf{x}_4' \perp \mathbf{x}_3' & \mathbf{f}_4 \mathbf{f}_3, i=4, j=3 \\ \{\mathbf{r}_1\}^D - \{\mathbf{r}_4\}^D & \text{revD} & D, \mathbf{r}_1 - \mathbf{r}_4, i=4, j=1 \\ \{\hat{\mathbf{f}}_4\}^T \{\hat{\mathbf{h}}_1\} & \mathbf{x}_4' \perp \mathbf{z}_1' & \mathbf{f}_4 \mathbf{h}_1, i=4, j=1 \\ \{\hat{\mathbf{g}}_4\}^T \{\hat{\mathbf{h}}_1\} & \mathbf{y}_4' \perp \mathbf{z}_1' & \mathbf{g}_4 \mathbf{h}_1, i=4, j=1 \\ \{\mathbf{p}_2\}^T \{\mathbf{p}_2\} - 1 & \text{Euler parameters} & \mathbf{p}_2 \\ \{\mathbf{p}_3\}^T \{\mathbf{p}_3\} - 1 & \text{Euler parameters} & \mathbf{p}_3 \\ \{\mathbf{p}_4\}^T \{\mathbf{p}_4\} - 1 & \text{Euler parameters} & \mathbf{p}_4 \\ \theta - C - f(t) & \text{driver} & \text{driver} \end{Bmatrix}
\end{aligned}$$

CONSTANTS

$$\{\mathbf{r}_1\}^A = \begin{Bmatrix} 0 \\ 0 \\ a \end{Bmatrix} \quad \{\mathbf{r}_1\}^D = \begin{Bmatrix} d \\ 0 \\ 0 \end{Bmatrix}$$

$$\{\mathbf{s}_2\}'^A = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \quad \{\mathbf{s}_2\}'^B = \begin{Bmatrix} 0 \\ 0 \\ -b \end{Bmatrix} \quad \{\mathbf{s}_3\}'^B = \begin{Bmatrix} 0 \\ 0 \\ e \end{Bmatrix} \quad \{\mathbf{s}_3\}'^C = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \quad \{\mathbf{s}_4\}'^C = \begin{Bmatrix} c \\ 0 \\ 0 \end{Bmatrix} \quad \{\mathbf{s}_4\}'^D = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}$$

INITIAL ESTIMATES

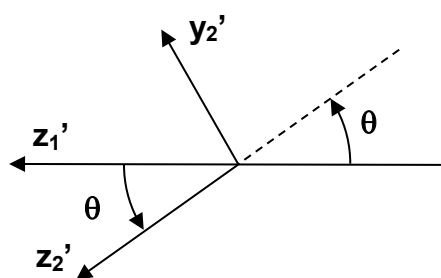
$$\{\mathbf{r}_2\} = \begin{Bmatrix} 0 \\ 0 \\ a \end{Bmatrix} \quad \chi_2 = \theta = 0 \text{ deg} \quad \{\hat{\mathbf{u}}_2\} = \begin{Bmatrix} u \\ v \\ w \end{Bmatrix} = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix} \quad \{\mathbf{p}_2\} = \begin{Bmatrix} \mathbf{e}_0 \\ \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{Bmatrix} = \begin{Bmatrix} C \frac{\chi}{2} \\ u S \frac{\chi}{2} \\ v S \frac{\chi}{2} \\ w S \frac{\chi}{2} \end{Bmatrix} = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

$$\{\mathbf{r}_3\} = \begin{Bmatrix} d \\ -c \\ 0 \end{Bmatrix} \quad \{\hat{\mathbf{h}}_3\} = \text{unit} \begin{Bmatrix} -d \\ c \\ a-b \end{Bmatrix} \quad \{\hat{\mathbf{f}}_3\} = \text{unit} \begin{Bmatrix} a-b \\ 0 \\ d \end{Bmatrix} \quad [\mathbf{A}_3] = \begin{bmatrix} \{\hat{\mathbf{f}}_3\} & \{\hat{\mathbf{h}}_3\} \times \{\hat{\mathbf{f}}_3\} & \{\hat{\mathbf{h}}_3\} \end{bmatrix}$$

$$\mathbf{e}_0^2 = (\text{tr}[\mathbf{A}] + 1)/4 \quad \begin{Bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{Bmatrix} = \frac{1}{4\mathbf{e}_0} \begin{Bmatrix} a_{32} - a_{23} \\ a_{13} - a_{31} \\ a_{21} - a_{12} \end{Bmatrix}$$

$$\{\mathbf{r}_4\} = \begin{Bmatrix} d \\ 0 \\ 0 \end{Bmatrix} \quad \chi_4 = \phi = 270 \text{ deg} \quad \{\hat{\mathbf{u}}_4\} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} = \begin{Bmatrix} \mathbf{u} \\ \mathbf{v} \\ \mathbf{w} \end{Bmatrix} \quad \{\mathbf{p}_4\} = \begin{Bmatrix} \mathbf{e}_0 \\ \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{Bmatrix} = \begin{Bmatrix} C\frac{\chi}{2} \\ \mathbf{u}S\frac{\chi}{2} \\ \mathbf{v}S\frac{\chi}{2} \\ \mathbf{w}S\frac{\chi}{2} \end{Bmatrix} = \begin{Bmatrix} -0.7071 \\ 0 \\ 0 \\ 0.7071 \end{Bmatrix}$$

FIXED REVOLUTE DRIVER

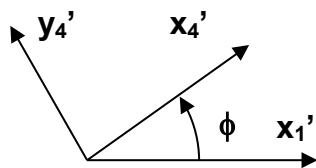


$$\{\hat{\mathbf{h}}_1\}^T \{\hat{\mathbf{h}}_2\} = C\theta$$

$$\{\hat{\mathbf{h}}_1\}^T \{\hat{\mathbf{g}}_2\} = S\theta$$

$$\theta = a \tan 2 \left(\frac{\{\hat{\mathbf{h}}_1\}^T \{\hat{\mathbf{g}}_2\}}{\{\hat{\mathbf{h}}_1\}^T \{\hat{\mathbf{h}}_2\}} \right)$$

OUTPUT LINK



$$\{\hat{\mathbf{f}}_1\}^T \{\hat{\mathbf{f}}_4\} = C\phi$$

$$\{\hat{\mathbf{f}}_1\}^T \{\hat{\mathbf{g}}_4\} = C(\phi + 90^\circ) = -S\phi$$

$$\phi = a \tan 2 \left(\frac{-\{\hat{\mathbf{f}}_1\}^T \{\hat{\mathbf{g}}_4\}}{\{\hat{\mathbf{f}}_1\}^T \{\hat{\mathbf{f}}_4\}} \right)$$

JACOBIAN

$$[\Phi_q] = \begin{bmatrix} -[I_3] & 2[A_2][\tilde{s}_2]'^A[G_2] & 0_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} & 0_{3 \times 4} \\ 0_{1 \times 3} & -2\{f_1\}'^T[A_1]^T[A_2][\tilde{g}_2]'^[G_2] & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} \\ 0_{1 \times 3} & -2\{f_1\}'^T[A_1]^T[A_2][\tilde{h}_2]'^[G_2] & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} \\ \\ [I_3] & -2[A_2][\tilde{s}_2]'^B[G_2] & -[I_3] & 2[A_3][\tilde{s}_3]'^B[G_3] & 0_{3 \times 3} & 0_{3 \times 4} \\ \\ 0_{3 \times 3} & 0_{3 \times 4} & [I_3] & -2[A_3][\tilde{s}_3]'^C[G_3] & -[I_3] & 2[A_4][\tilde{s}_4]'^C[G_4] \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & -2\{f_4\}'^T[A_4]^T[A_3][\tilde{f}_3]'^[G_3] & 0_{1 \times 3} & -2\{f_3\}'^T[A_3]^T[A_4][\tilde{f}_4]'^[G_4] \\ \\ 0_{3 \times 3} & 0_{3 \times 4} & 0_{3 \times 3} & 0_{3 \times 4} & -[I_3] & 2[A_4][\tilde{s}_4]'^D[G_4] \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & -2\{h_1\}'^T[A_1]^T[A_4][\tilde{f}_4]'^[G_4] \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & -2\{h_1\}'^T[A_1]^T[A_4][\tilde{g}_4]'^[G_4] \\ \\ 0_{1 \times 3} & 2\{p_2\}^T & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 2\{p_3\}^T & 0_{1 \times 3} & 0_{1 \times 4} \\ 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 2\{p_4\}^T \\ \\ 0_{1 \times 3} & 2\{\hat{u}_2\}^T[G_2] & 0_{1 \times 3} & 0_{1 \times 4} & 0_{1 \times 3} & 0_{1 \times 4} \end{bmatrix} \begin{matrix} A, r_1 - r_2, i = 2, j = 1 \\ g_2 f_1, i = 2, j = 1 \\ h_2 f_1, i = 2, j = 1 \\ \\ B, r_2 - r_3, i = 3, j = 2 \\ \\ C, r_3 - r_4, i = 4, j = 3 \\ f_4 f_3, i = 4, j = 3 \\ \\ D, r_1 - r_4, i = 4, j = 1 \\ f_4 h_1, i = 4, j = 1 \\ g_4 h_1, i = 4, j = 1 \\ \\ p_2 \\ p_3 \\ p_4 \\ \\ driver \end{matrix}$$

$$\frac{\partial}{\partial \{p_i\}} = 2 \left(\frac{\partial}{\partial \{\pi_i\}'} \right) [G_i]$$

VELOCITY

$$\{v\} = \left\{ \begin{array}{l} 0 \\ 0 \\ 0 \\ \\ 0 \\ \\ 0 \\ 0 \\ 0 \\ \\ 0 \\ 0 \\ 0 \\ \\ 0 \\ 0 \\ 0 \\ + f_t \end{array} \right\} \begin{array}{l} A, r_1 - r_2, i = 2, j = 1 \\ g_2 f_1, i = 2, j = 1 \\ h_2 f_1, i = 2, j = 1 \\ \\ B, r_2 - r_3, i = 3, j = 2 \\ \\ C, r_3 - r_4, i = 4, j = 3 \\ f_4 f_3, i = 4, j = 3 \\ \\ D, r_1 - r_4, i = 4, j = 1 \\ f_4 h_1, i = 4, j = 1 \\ g_4 h_1, i = 4, j = 1 \\ \\ p_2 \\ p_3 \\ p_4 \\ \\ driver \end{array}$$

ACCELERATION

$$\{\gamma\} = \left\{ \begin{array}{l} [A_2][\tilde{\omega}_2]'[\tilde{\omega}_2]'\{s_2\}'^A \\ -\{f_1\}'^T ([A_1]^T [A_2][\tilde{\omega}_2]'[\tilde{\omega}_2]')\{g_2\}' \\ -\{f_1\}'^T ([A_1]^T [A_2][\tilde{\omega}_2]'[\tilde{\omega}_2]')\{h_2\}' \\ \\ [A_3][\tilde{\omega}_3]'[\tilde{\omega}_3]'\{s_3\}'^B - [A_2][\tilde{\omega}_2]'[\tilde{\omega}_2]'\{s_2\}'^B \\ \\ [A_4][\tilde{\omega}_4]'[\tilde{\omega}_4]'\{s_4\}'^C - [A_3][\tilde{\omega}_3]'[\tilde{\omega}_3]'\{s_3\}'^C \\ -\{f_3\}'^T ([A_3]^T [A_i][\tilde{\omega}_4]'[\tilde{\omega}_4]'+2[\tilde{\omega}_3]'^T [A_3]^T [A_4][\tilde{\omega}_4]'+[\tilde{\omega}_3]'[\tilde{\omega}_3]'[A_3]^T [A_4])\{f_4\}' \\ \\ [A_4][\tilde{\omega}_4]'[\tilde{\omega}_4]'\{s_4\}'^D \\ -\{h_1\}'^T ([A_1]^T [A_4][\tilde{\omega}_4]'[\tilde{\omega}_4]')\{f_4\}' \\ -\{h_1\}'^T ([A_1]^T [A_4][\tilde{\omega}_4]'[\tilde{\omega}_4]')\{g_4\}' \\ \\ -2\{\dot{p}_2\}'^T \{\dot{p}_2\}' \\ -2\{\dot{p}_3\}'^T \{\dot{p}_3\}' \\ -2\{\dot{p}_4\}'^T \{\dot{p}_4\}' \\ \\ + f_{tt} \end{array} \right\} \begin{array}{l} A, r_1 - r_2, i = 2, j = 1 \\ g_2 f_1, i = 2, j = 1 \\ h_2 f_1, i = 2, j = 1 \\ \\ B, r_2 - r_3, i = 3, j = 2 \\ \\ C, r_3 - r_4, i = 4, j = 3 \\ f_4 f_3, i = 4, j = 3 \\ \\ D, r_1 - r_4, i = 4, j = 1 \\ f_4 h_1, i = 4, j = 1 \\ g_4 h_1, i = 4, j = 1 \\ \\ p_2 \\ p_3 \\ p_4 \\ \\ driver \end{array}$$

$$\text{dot} - 1 \quad \Phi = \{\hat{a}_i\}'^T \{\hat{a}_j\} \quad \gamma = -\{a_j\}'^T ([A_j]^T [A_i][\tilde{\omega}_i]'[\tilde{\omega}_i]'+2[\tilde{\omega}_j]'^T [A_j]^T [A_i][\tilde{\omega}_i]'+[\tilde{\omega}_j]'[\tilde{\omega}_j]'[A_j]^T [A_i])\{a_i\}'$$

$$\text{sphere} \quad \{\Phi\} = \{r_j\}'^P - \{r_j\}'^P \quad \gamma = [A_i][\tilde{\omega}_i]'[\tilde{\omega}_i]'\{s_i\}'^P - [A_j][\tilde{\omega}_j]'[\tilde{\omega}_j]'\{s_j\}'^P$$

JERK

$$\{\eta\} = \left\{ \begin{array}{l} \begin{array}{l} [A_2][H_2]\{s_2\}'^A \\ -\{f_1\}'^T [A_1]^T [A_2][H_2]\{g_2\}' \\ -\{f_1\}'^T [A_1]^T [A_2][H_2]\{h_2\}' \end{array} \\ \\ [A_3][H_3]\{s_3\}'^B - [A_2][H_2]\{s_2\}'^B \\ \\ [A_4][H_4]\{s_4\}'^C - [A_3][H_3]\{s_3\}'^C \\ \left(\begin{array}{l} -\{f_3\}'^T [A_3]^T [A_4][H_4]\{f_4\}' - \{f_4\}'^T [A_4]^T [A_3][H_3]\{f_3\}' \\ -3\{f_3\}'^T ([\tilde{\omega}_3]'[\tilde{\omega}_3]' - [\dot{\tilde{\omega}}_3]') [A_3]^T [A_4][\tilde{\omega}_4]'\{f_4\}' - 3\{f_4\}'^T ([\tilde{\omega}_4]'[\tilde{\omega}_4]' - [\dot{\tilde{\omega}}_4]') [A_4]^T [A_3][\tilde{\omega}_3]'\{f_3\}' \end{array} \right) \\ \\ \begin{array}{l} [A_4][H_4]\{s_4\}'^D \\ -\{h_1\}'^T [A_1]^T [A_4][H_4]\{f_4\}' \\ -\{h_1\}'^T [A_1]^T [A_4][H_4]\{g_4\}' \end{array} \\ \\ \begin{array}{l} -6\{\ddot{p}_2\}'^T \{\dot{p}_2\}' \\ -6\{\ddot{p}_3\}'^T \{\dot{p}_3\}' \\ -6\{\ddot{p}_4\}'^T \{\dot{p}_4\}' \end{array} \\ \\ -\dot{\theta}^3/4 + f_{\text{tot}} \quad \text{for link 2} \end{array} \right\}$$

A, $r_1 - r_2, i = 2, j = 1$ $g_2 f_1, i = 2, j = 1$ $h_2 f_1, i = 2, j = 1$ B, $r_2 - r_3, i = 3, j = 2$ C, $r_3 - r_4, i = 4, j = 3$ $f_4 f_3, i = 4, j = 3$ D, $r_1 - r_4, i = 4, j = 1$ $f_4 h_1, i = 4, j = 1$ $g_4 h_1, i = 4, j = 1$ p_2 p_3 p_4

driver

$$\text{dot} - 1 \quad \left(\begin{array}{l} \text{jerk product} \quad [H_i] = 2[\tilde{\omega}_i]'[\dot{\tilde{\omega}}_i]' + [\tilde{\omega}_i]'[\ddot{\tilde{\omega}}_i]' + [\tilde{\omega}_i]'[\tilde{\omega}_i]'[\ddot{\tilde{\omega}}_i]' \\ \left(\begin{array}{l} \eta = -\{a_j\}'^T [A_j]^T [A_i][H_i]\{a_i\}' - \{a_i\}'^T [A_i]^T [A_j][H_j]\{a_j\}' \\ -3\{a_j\}'^T ([\tilde{\omega}_j]'[\tilde{\omega}_j]' - [\dot{\tilde{\omega}}_j]') [A_j]^T [A_i][\tilde{\omega}_i]'\{a_i\}' - 3\{a_i\}'^T ([\tilde{\omega}_i]'[\tilde{\omega}_i]' - [\dot{\tilde{\omega}}_i]') [A_i]^T [A_j][\tilde{\omega}_j]'\{a_j\}' \end{array} \right) \\ \text{sphere} \quad \eta = [A_i][H_i]\{s_i\}'^P - [A_j][H_j]\{s_j\}'^P \end{array} \right)$$

RSUR - Inertial Properties

link 2

density $\rho = 1.18 \text{ g/cm}^3$ plexiglass

diameter $D = 5 \text{ inch} = 12.7 \text{ cm}$

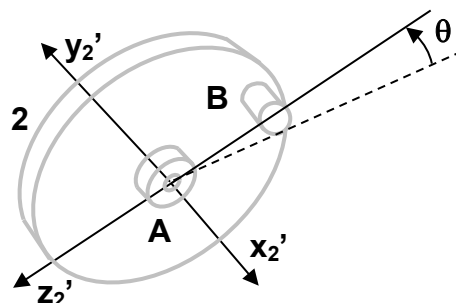
$t = 0.5 \text{ inch} = 1.27 \text{ cm}$

$$m_2 = \rho t \pi D^2 / 4 = 189.84 \text{ g} = 0.18984 \text{ kg}$$

$$J_{xx_2} = mR^2 / 2 = mD^2 / 8 = 3.8274 \text{ kg.cm}^2$$

$$J_{yy_2} = mR^2 / 4 + mt^2 / 12 = 1.9177 \text{ kg.cm}^2$$

$$J_{zz_2} = mR^2 / 4 + mt^2 / 12 = 1.9177 \text{ kg.cm}^2$$



link 3

density $\rho = 1.18 \text{ g/cm}^3$ plexiglass

diameter $D = 0.5 \text{ inch} = 1.27 \text{ cm}$

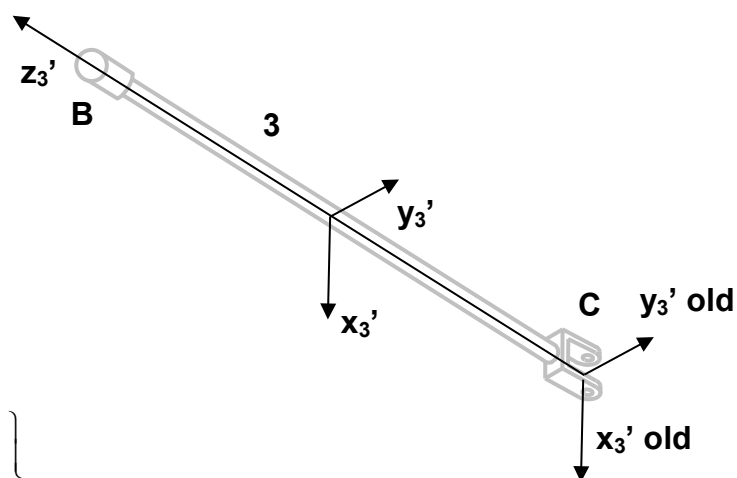
length $L = 30.42 \text{ cm}$

$$m_3 = \rho L \pi D^2 / 4 = 45.47 \text{ g} = 0.04547 \text{ kg}$$

$$J_{xx_3} = mL^2 / 12 = 3.5064 \text{ kg.cm}^2$$

$$J_{yy_3} = mL^2 / 12 = 3.5064 \text{ kg.cm}^2$$

$$J_{zz_3} = mR^2 / 2 = mD^2 / 8 = 0.009167 \text{ kg.cm}^2$$



shift centroid $\{s_3\}^B = \begin{Bmatrix} 0 \\ 0 \\ +e/2 \end{Bmatrix}$ $\{s_3\}^C = \begin{Bmatrix} 0 \\ 0 \\ -e/2 \end{Bmatrix}$

initial estimate $\{r_3\} = \begin{Bmatrix} d/2 \\ -c/2 \\ (a-b)/2 \end{Bmatrix}$

link 4

density $\rho = 1.18 \text{ g/cm}^3$ plexiglass

diameter $D = 9.5 \text{ inch} = 24.13 \text{ cm}$

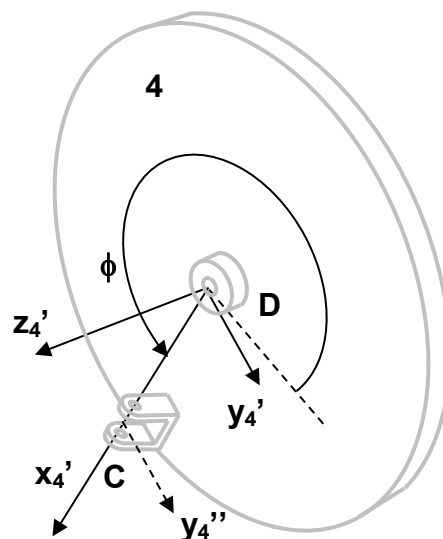
$t = 0.5 \text{ inch} = 1.27 \text{ cm}$

$$m_4 = \rho t \pi D^2 / 4 = 685.32 \text{ g} = 0.68532 \text{ kg}$$

$$J_{xx_4} = mR^2 / 4 + mt^2 / 12 = 24.9534 \text{ kg.cm}^2$$

$$J_{yy_4} = mR^2 / 4 + mt^2 / 12 = 24.9534 \text{ kg.cm}^2$$

$$J_{zz_4} = mR^2 / 2 = mD^2 / 8 = 49.879 \text{ kg.cm}^2$$

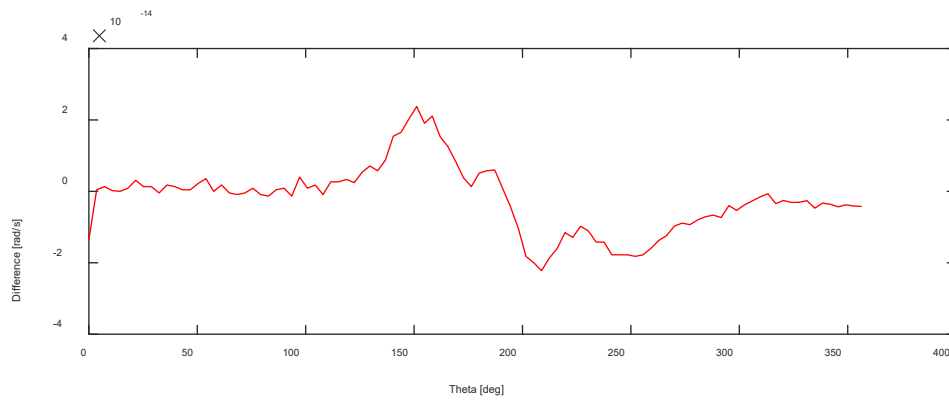
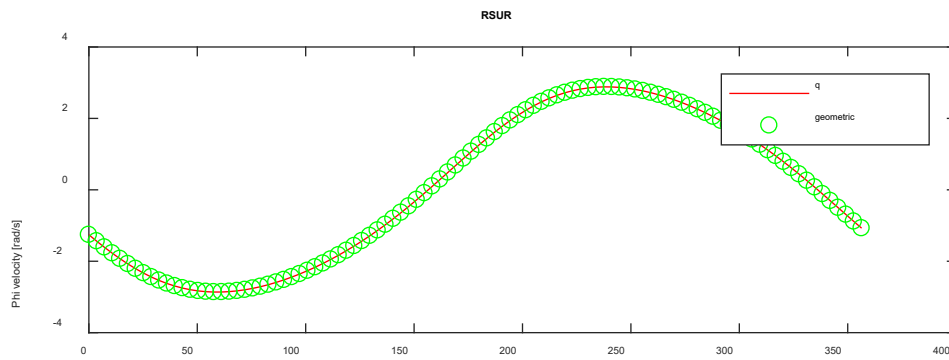
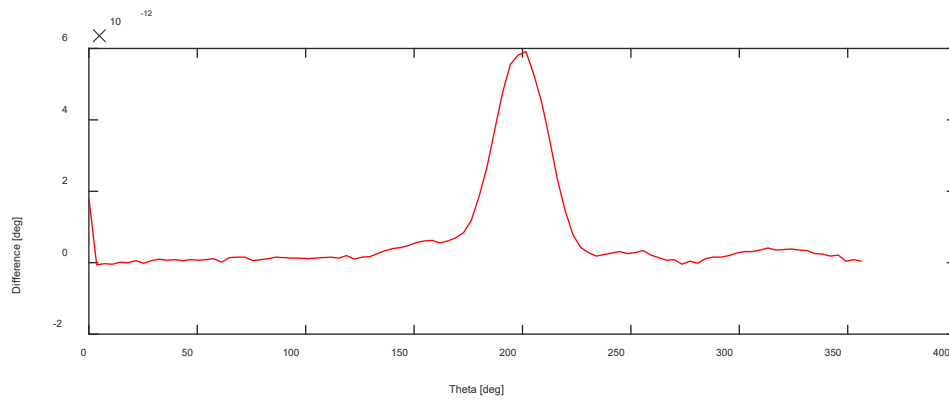
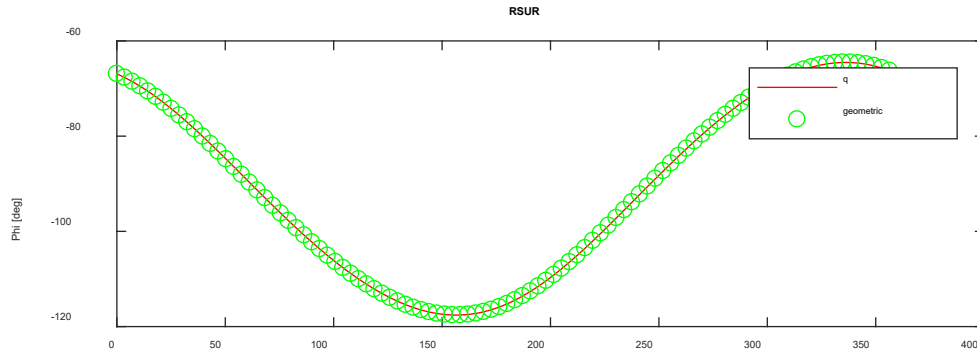


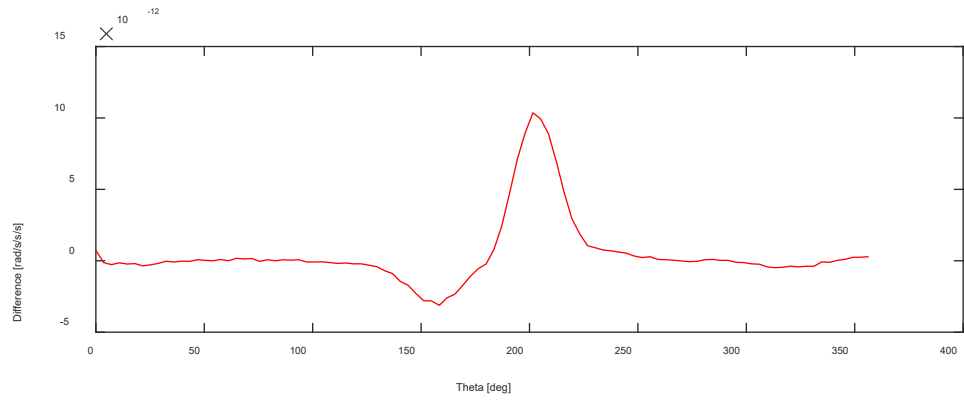
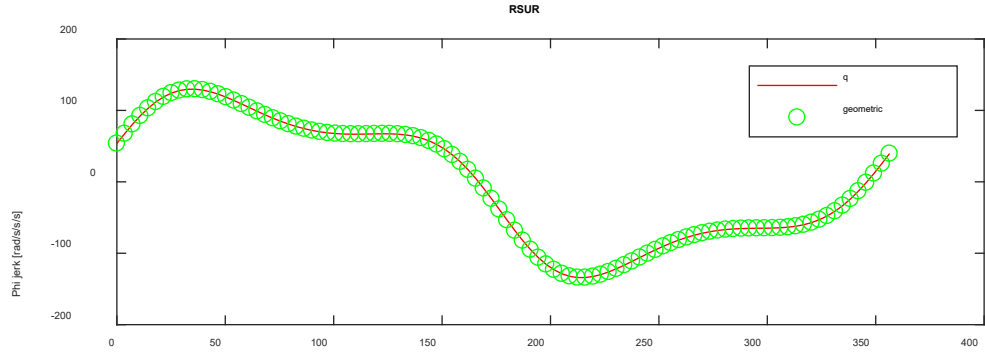
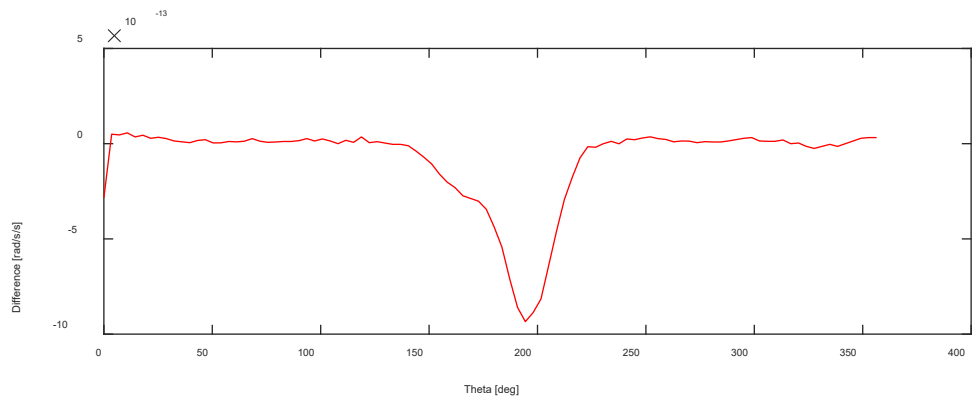
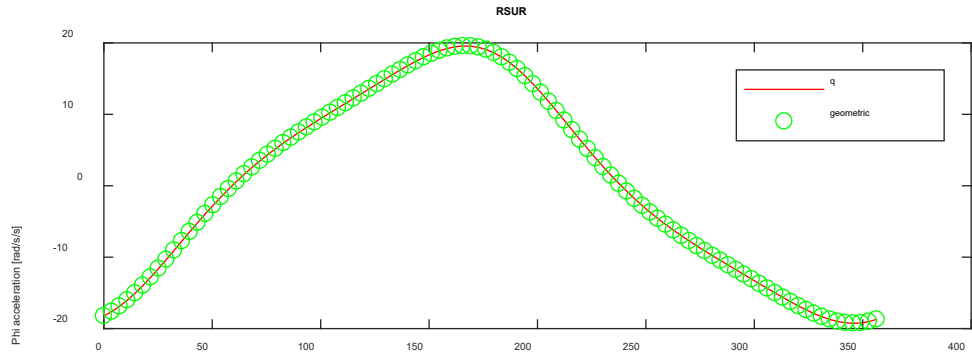
RSUR - Forward Time Integration

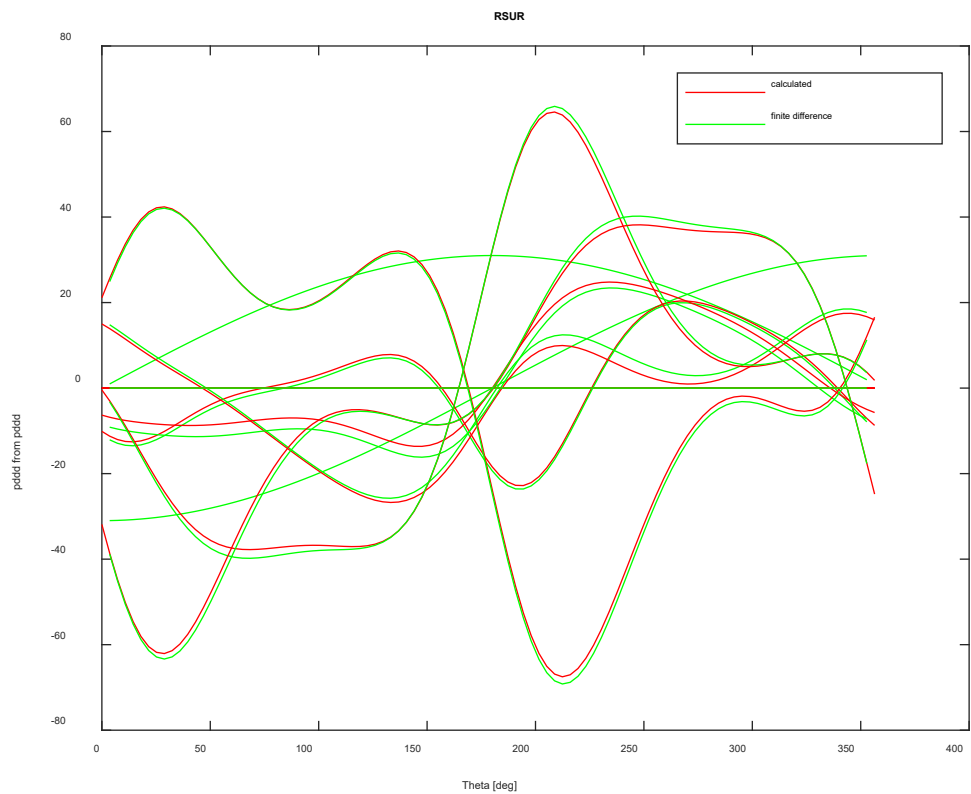
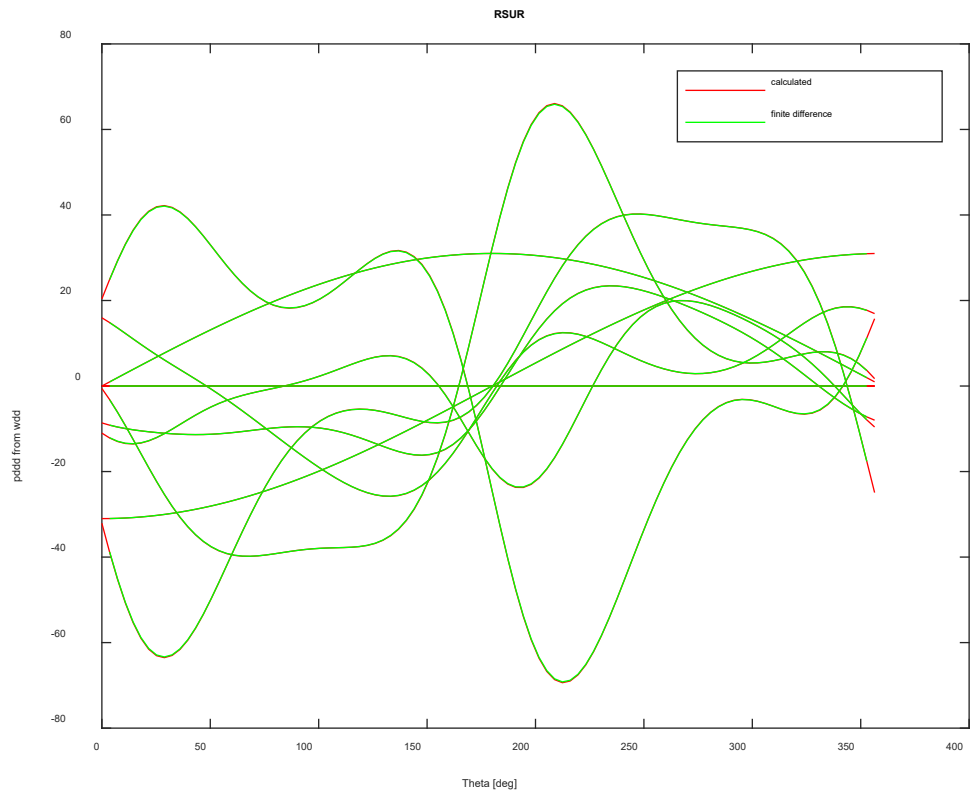
$$\{y\} = \begin{Bmatrix} \{r\}_{9 \times 1} \\ \{p\}_{12 \times 1} \\ \{\dot{r}\}_{9 \times 1} \\ \{\omega\}'_{9 \times 1} \end{Bmatrix}$$

$$\{\dot{y}\} = \begin{Bmatrix} \{\dot{r}\}_{9 \times 1} \\ \{\dot{p}\}_{12 \times 1} \\ \{\ddot{r}\}_{9 \times 1} \\ \{\dot{\omega}\}'_{9 \times 1} \end{Bmatrix} \quad \{\dot{p}\} = \frac{1}{2} [G]^T \{\omega\}' \quad [G] = \begin{bmatrix} [G_2] & \mathbf{0}_{3 \times 4} & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{3 \times 4} & [G_3] & \mathbf{0}_{3 \times 4} \\ \mathbf{0}_{3 \times 4} & \mathbf{0}_{3 \times 4} & \mathbf{O} \end{bmatrix}$$

$$\{\ddot{y}\} = \begin{Bmatrix} \{\ddot{r}\}_{9 \times 1} \\ \{\ddot{p}\}_{12 \times 1} \\ \{\ddot{r}\}_{9 \times 1} \\ \{\ddot{\omega}\}'_{9 \times 1} \end{Bmatrix} \quad \{\ddot{p}\} = \frac{1}{2} [G]^T \{\dot{\omega}\}' - \frac{1}{4} \{p\} (\{\omega\}'^T \{\omega\}')$$







```

% rsur_main.m - RSUR mechanism
% position, velocity, acceleration, jerk
% HJSIII 21.04.19

clear

% initialize
rsur_ini;

% decelerating driver for crank
% time for one revolution of constant speed rotation driver
t_one_rev = 2*pi / driver_speed;
dt = t_one_rev / 100; % time step
tend = 99 * dt; % time duration

% allocate space to save results
keep_q = [];
keep_geo = [];
keep_fd_rw = [];
keep_fd_p = [];

% time loop
for t = 0 : dt : tend;

% kinematics
rsur_kin;

th_q = theta;
phi_q = phi;
phid_q = phid;
phidd_q = phidd;
phiddd_q = phiddd;

% geometric solution

% decelerating driver for crank
th = driver_start + driver_speed*t + driver_accel*t*t/2;
thd = driver_speed + driver_accel*t;
thdd = driver_accel;
thddd = 0;

% 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_geo = (-g - sqrt( f*f + g*g - h*h )) / (h+f);
u2_geo = (-g + sqrt( f*f + g*g - h*h )) / (h+f);
phi1 = 2*atan(u1_geo);
phi2 = 2*atan(u2_geo); % 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;

% save kinematics
keep_q = [ keep_q ; th_q phi_q phid_q phidd_q phiddd_q ];
keep_geo = [ keep_geo ; th_phi phid phidd phiddd ];
keep_fd_rw = [ keep_fd_rw ; [ test_pdd_rw test_pddd_rw ] ];
keep_fd_p = [ keep_fd_p ; [ test_pdd_p test_pddd_p ] ];

% bottom of crank rotation loop
end

% values for plotting
th_q = keep_q(:,1) /d2r;
phi_q = keep_q(:,2) /d2r;
phid_q = keep_q(:,3);
phidd_q = keep_q(:,4);
phiddd_q = keep_q(:,5);

th_geo = keep_geo(:,1) /d2r;
phi_geo = keep_geo(:,2) /d2r;
phid_geo = keep_geo(:,3);
phidd_geo = keep_geo(:,4);
phiddd_geo = keep_geo(:,5);

% difference between explicit geometric and constraint solutions
err_pos = phi_q - phi_geo;
err_vel = phid_q - phid_geo;
err_acc = phidd_q - phidd_geo;
err_jrk = phiddd_q - phiddd_geo;

rms_pos = std( err_pos );
rms_vel = std( err_vel );
rms_acc = std( err_acc );
rms_jrk = std( err_jrk );

max_tol = e;
max_pos = max( abs( phi_geo ) );
max_vel = max( abs( phid_geo ) );
max_acc = max( abs( phidd_geo ) );
max_jrk = max( abs( phiddd_geo ) );

nor_tol = assy_tol / max_tol;
nor_pos = rms_pos / max_pos;
nor_vel = rms_vel / max_vel;
nor_acc = rms_acc / max_acc;
nor_jrk = rms_jrk / max_jrk;

disp( ' ' )
disp( ' assy_tol rms_pos rms_vel rms_acc rms_jrk nor_tol nor_pos nor_vel nor_acc
nor_jrk' )
disp( log10( [ assy_tol rms_pos rms_vel rms_acc rms_jrk nor_tol nor_pos nor_vel nor_acc
nor_jrk ] ) )

% plot position solution
figure( 1 )
clf
subplot( 2, 1, 1 )
plot( th_q,phi_q,'r', th_q,phi_geo,'go' )
title( 'RSUR' )
ylabel( 'Phi [deg]' )
legend( 'q', 'geometric' )

subplot( 2, 1, 2 )
plot( th_q,err_pos,'r' )
ylabel( 'Difference [deg]' )
xlabel( 'Theta [deg]' )

% plot velocity solution
figure( 2 )
clf
subplot( 2, 1, 1 )

```

```

plot( th_q,phid_q,'r', th_q,phid_geo,'go' )
title( 'RSUR' )
ylabel( 'Phi velocity [rad/s]' )
legend( 'q', 'geometric' )

subplot( 2, 1, 2 )
plot( th_q,err_vel,'r' )
ylabel( 'Difference [rad/s]' )
xlabel( 'Theta [deg]' )

% plot acceleration solution
figure( 3 )
clf
subplot( 2, 1, 1 )
plot( th_q,phidd_q,'r', th_q,phidd_geo,'go' )
title( 'RSUR' )
ylabel( 'Phi acceleration [rad/s/s]' )
legend( 'q', 'geometric' )

subplot( 2, 1, 2 )
plot( th_q,err_acc,'r' )
ylabel( 'Difference [rad/s/s]' )
xlabel( 'Theta [deg]' )

% plot jerk solution
figure( 4 )
clf
subplot( 2, 1, 1 )
plot( th_q,phidd_q,'r', th_q,phidd_geo,'go' )
title( 'RSUR' )
ylabel( 'Phi jerk [rad/s/s/s]' )
legend( 'q', 'geometric' )

subplot( 2, 1, 2 )
plot( th_q,err_jrk,'r' )
ylabel( 'Difference [rad/s/s/s]' )
xlabel( 'Theta [deg]' )

% check pddd from rw by finite difference
ncol = 12;
figure( 5 )
clf
raw_rw = keep_fd_rw(:,1:ncol);
der_rw = keep_fd_rw(:,ncol+1:end);

row_NaN = NaN * ones(1,ncol);
fd_rw = ( [ row_NaN; diff(raw_rw) ] + [ diff(raw_rw); row_NaN ] ) /2/dt;

plot( th_q,der_rw(:,1),'r', th_q,fd_rw(:,1),'g' )
hold on
for icol = 2 : ncol,
    plot( th_q,der_rw(:,icol),'r', th_q,fd_rw(:,icol),'g' )
end
title( 'RSUR' )
xlabel( 'Theta [deg]' )
ylabel( 'pddd from rw' )
legend( 'calculated', 'finite difference' )

% check pddd from p by finite difference
ncol = 12;
figure( 6 )
clf
raw_p = keep_fd_p(:,1:ncol);
der_p = keep_fd_p(:,ncol+1:end);

row_NaN = NaN * ones(1,ncol);
fd_p = ( [ row_NaN; diff(raw_p) ] + [ diff(raw_p); row_NaN ] ) /2/dt;

plot( th_q,der_p(:,1),'r', th_q,fd_p(:,1),'g' )
hold on
for icol = 2 : ncol,

```

```
    plot( th_q,der_p(:,icol),'r', th_q,fd_p(:,icol),'g' )  
end  
title( 'RSUR' )  
xlabel( 'Theta [deg]' )  
ylabel( 'pddd from p' )  
legend( 'calculated', 'finite difference' )  
  
% bottom - rsur_main
```



```

% rsur_ini.m - RSUR mechanism
% initialize constants and assembly estimates
% HJSIII, 21.04.08

% general constants
d2r = pi/180;

% local coordinate axes
fp = [ 1 0 0]';
gp = [ 0 1 0]';
hp = [ 0 0 1]';

% link lengths - units = [cm]
a = 20.43;
b = 4.00;
c = 10.00;
d = 19.97;
e = 30.42;

% link 1 is fixed at origin
r1 = [ 0 0 0 ]';
p1 = [ 1 0 0 0 ]';
[ E1,G1,A1,f1,g1,h1 ] = make_ega(p1);

% joint locations on links - units = [cm]
slpA = [ 0 0 a ]';
slpD = [ d 0 0 ]';

s2pA = [ 0 0 0 ]';
s2pB = [ 0 0 -b ]';

s3pB = [ 0 0 e ]';
s3pC = [ 0 0 0 ]';

s4pC = [ c 0 0 ]';
s4pD = [ 0 0 0 ]';

% fixed joint locations
r1A = r1 + A1*slpA;
r1D = r1 + A1*slpD;

% initial estimates
r2 = [ 0 0 a ]';
chi2 = 5 * d2r;
u2 = [ 1 0 0 ]';
p2 = [ cos(chi2/2) u2(1)*sin(chi2/2) u2(2)*sin(chi2/2) u2(3)*sin(chi2/2) ]';

r3 = [ d -c 0 ]';
chi3 = 0 * d2r;
h3 = [ -d c a-b ]';
h3 = h3 / norm(h3);
f3 = [ h3(3) 0 -h3(1) ]';
f3 = f3 / norm(f3);
g3 = cross( h3, f3 );
A3 = [ f3 g3 h3 ];
e0 = sqrt( ( trace(A3)+1 ) / 4 );
p3 = [ e0 ;
        (A3(3,2)-A3(2,3))/4/e0 ;
        (A3(1,3)-A3(3,1))/4/e0 ;
        (A3(2,1)-A3(1,2))/4/e0 ];

r4 = [ d 0 0 ]';
chi4 = 270 * d2r;
u4 = [ 0 0 1 ]';
p4 = [ cos(chi4/2) u4(1)*sin(chi4/2) u4(2)*sin(chi4/2) u4(3)*sin(chi4/2) ]';

% generalized coordinates
q = [ r2 ; p2 ; r3 ; p3 ; r4 ; p4 ];

% fixed revolute rotation driver at A for link 2 about f2
% driver = driver_start + driver_speed*t + driver_accel*t*t/2

```

```
driver_start = 5 * d2r; % [rad] problems for old rotation driver Jacobian at 0 degrees
driver_start = 0;      % [rad]
driver_speed = 2*pi;   % [rad/s] 1 rev/sec
driver_accel = -1;     % [rad/s/s]
driver_accel = 0;      % [rad/s/s]
t = 0;                 % time

% bottom - rsur_ini
```

```

% rsur_phi.m - RSUR mechanism
% evaluate constraints and Jacobian
% HJSIII, 21.04.19

% global location of local frames
r2 = q(1:3);
p2 = q(4:7);
[ E2,G2,A2,f2,g2,h2 ] = make_ega(p2);

r3 = q(8:10);
p3 = q(11:14);
[ E3,G3,A3,f3,g3,h3 ] = make_ega(p3);

r4 = q(15:17);
p4 = q(18:21);
[ E4,G4,A4,f4,g4,h4 ] = make_ega(p4);

% global locations of joint centers
r2A = r2 + A2*s2pA;
r2B = r2 + A2*s2pB;

r3B = r3 + A3*s3pB;
r3C = r3 + A3*s3pC;

r4C = r4 + A4*s4pC;
r4D = r4 + A4*s4pD;

% constraints and Jacobian
PHI = zeros(21,1);
JAC = zeros(21,21);

% revolute A - j=1, i=2
PHI(1:3) = r1A - r2A; % j=1, i=2
JAC(1:3,1:3) = -eye(3);
JAC(1:3,4:7) = 2 * A2 * skew_sym(s2pA) * G2;

PHI(4) = g2' * f1; % ai=g2, aj=f1
JAC(4,1:3) = zeros(1,3);
JAC(4,4:7) = -2 * fp' * A1' * A2 * skew_sym(gp) * G2;

PHI(5) = h2' * f1; % ai=h2, aj=f1
JAC(5,1:3) = zeros(1,3);
JAC(5,4:7) = -2 * fp' * A1' * A2 * skew_sym(hp) * G2;

% spherical B - j=2, i=3
PHI(6:8) = r2B - r3B; % j=2, i=3
JAC(6:8,1:3) = eye(3);
JAC(6:8,4:7) = -2 * A2 * skew_sym(s2pB) * G2;
JAC(6:8,8:10) = -eye(3);
JAC(6:8,11:14) = 2 * A3 * skew_sym(s3pB) * G3;

% universal C - j=3, i=4
PHI(9:11) = r3C - r4C; % j=3, i=4
JAC(9:11,8:10) = eye(3);
JAC(9:11,11:14) = -2 * A3 * skew_sym(s3pC) * G3;
JAC(9:11,15:17) = -eye(3);
JAC(9:11,18:21) = 2 * A4 * skew_sym(s4pC) * G4;

PHI(12) = f4' * f3; % ai=f4, aj=f3
JAC(12,8:10) = zeros(1,3);
JAC(12,11:14) = -2 * fp' * A4' * A3 * skew_sym(fp) * G3;
JAC(12,15:17) = zeros(1,3);
JAC(12,18:21) = -2 * fp' * A3' * A4 * skew_sym(fp) * G4;

% revolute D - j=1, i=4
PHI(13:15) = r1D - r4D; % j=1, i=4
JAC(13:15,15:17) = -eye(3);
JAC(13:15,18:21) = 2 * A4 * skew_sym(s4pD) * G4;

PHI(16) = f4' * h1; % ai=f4, aj=h1
JAC(16,15:17) = zeros(1,3);

```

```

JAC(16,18:21) = -2 * hp' * A1' * A4 * skew_sym(fp) * G4;

PHI(17) = g4' * h1; % ai=g4, aj=h1
JAC(17,15:17) = zeros(1,3);
JAC(17,18:21) = -2 * hp' * A1' * A4 * skew_sym(gp) * G4;

% Euler parameters
PHI(18) = p2'*p2 - 1;
JAC(18,4:7) = 2 * p2';

PHI(19) = p3'*p3 - 1;
JAC(19,11:14) = 2 * p3';

PHI(20) = p4'*p4 - 1;
JAC(20,18:21) = 2 * p4';

% fixed revolute rotation driver at A for link 2 about u2
e0 = p2(1);
if abs(e0) > 1;
    e0 = sign( e0 );
end
theta = mod( (atan2( h1'*g2, h1'*h2 ) + 2*pi) , 2*pi );

% fixed revolute rotation driver at A for link 2 about u2
PHI(21) = theta - driver_start - driver_speed*t - driver_accel*t*t/2;

JAC(21,4:7) = 2*u2'*G2;

% constraints and Jacobian for r and w'
PHI_rw = [ PHI(1:17) ;
            PHI(21) ];

JAC_r = [ JAC(1:17,1:3)  JAC(1:17,8:10)  JAC(1:17,15:17) ;
           JAC(21,1:3)   JAC(21,8:10)     JAC(21,15:17) ];
JAC_w = [ JAC(1:17,4:7)*G2'/2  JAC(1:17,11:14)*G3'/2  JAC(1:17,18:21)*G4'/2 ;
           JAC(21,4:7)*G2'/2   JAC(21,11:14)*G3'/2   JAC(21,18:21)*G4'/2 ];
JAC_rw = [ JAC_r  JAC_w ];

% output angle
phi = atan2( -f1'*g4, f1'*f4 );

% bottom - rsur_phi

```

```

% rsur_kin.m - RSUR mechanism
% position, velocity, acceleration, jerk
% HJSIII, 21.04.19

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Newton-Raphson position solution
assy_tol = 1.0e-12;
rsur_phi;
%test_jac; JAC=jtest; % numerical Jacobian

while max(abs(PHI)) > assy_tol,
    q = q - inv(JAC) * PHI;
    rsur_phi;
    % test_jac; JAC=jtest; % numerical Jacobian
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% velocity

% pd formulation works OK
velrhs = zeros(21,1);

% fixed revolute driver at A for link 2 about u2
velrhs(21) = driver_speed + driver_accel*t;

qd = inv(JAC) * velrhs;

% velocity from pd formulation
r2d = qd(1:3);
p2d = qd(4:7);

r3d = qd(8:10);
p3d = qd(11:14);

r4d = qd(15:17);
p4d = qd(18:21);

w2p = 2 * G2 * p2d;
w3p = 2 * G3 * p3d;
w4p = 2 * G4 * p4d;

test_rd_p = [ r2d' r3d' r4d' ];
test_pd_p = [ p2d' p3d' p4d' ];
test_wp_p = [ w2p' w3p' w4p' ];

% rw formulation works OK
velrhs_rw = zeros(18,1);
velrhs_rw(18) = velrhs(21); % same as pd formulation

qd_rw = inv(JAC_rw) * velrhs_rw;

% velocity from rw formulation
rd = qd_rw(1:9);
r2d = rd(1:3);
r3d = rd(4:6);
r4d = rd(7:9);

wp = qd_rw(10:18);
w2p = wp(1:3);
w3p = wp(4:6);
w4p = wp(7:9);

p2d = G2'*w2p/2;
p3d = G3'*w3p/2;
p4d = G4'*w4p/2;

test_rd_rw = [ r2d' r3d' r4d' ];
test_pd_rw = [ p2d' p3d' p4d' ];
test_wp_rw = [ w2p' w3p' w4p' ];

% use values

```

```

w2psk = skew_sym( w2p );
w3psk = skew_sym( w3p );
w4psk = skew_sym( w4p );

w4 = A4 * w4p;
phid = w4(3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% acceleration

% pdd formulation works OK
accrhs = zeros(21,1);

% revolute A - j=1, i=2
accrhs(1:3) = A2*w2psk*w2psk*s2pA;
accrhs(4) = -fp'*(A1'*A2*w2psk*w2psk)*gp;
accrhs(5) = -fp'*(A1'*A2*w2psk*w2psk)*hp;

% spherical B - j=2, i=3
accrhs(6:8) = A3*w3psk*w3psk*s3pB - A2*w2psk*w2psk*s2pB;

% universal C - j=3, i=4
accrhs(9:11) = A4*w4psk*w4psk*s4pC - A3*w3psk*w3psk*s3pC;

accrhs(12) = -fp' * (A3'*A4*w4psk*w4psk ...
                    +2*w3psk'*A3'*A4*w4psk ...
                    +w3psk*w3psk*A3'*A4) * fp;

% revolute D - j=1, i=4
accrhs(13:15) = A4*w4psk*w4psk*s4pD;
accrhs(16) = -hp'*(A1'*A4*w4psk*w4psk)*fp;
accrhs(17) = -hp'*(A1'*A4*w4psk*w4psk)*gp;

% Euler parameters
accrhs(18) = -2*p2d'*p2d;
accrhs(19) = -2*p3d'*p3d;
accrhs(20) = -2*p4d'*p4d;

% fixed revolute driver at A for link 2 about u2
accrhs(21) = driver_accel;

qdd = inv(JAC) * accrhs;

% acceleration from pdd formulation
r2dd = qdd(1:3);
p2dd = qdd(4:7);

r3dd = qdd(8:10);
p3dd = qdd(11:14);

r4dd = qdd(15:17);
p4dd = qdd(18:21);

w2pd = 2 * G2 * p2dd;
w3pd = 2 * G3 * p3dd;
w4pd = 2 * G4 * p4dd;

test_rdd_p = [ r2dd' r3dd' r4dd' ];
test_pdd_p = [ p2dd' p3dd' p4dd' ];
test_wpd_p = [ w2pd' w3pd' w4pd' ];

% rw formulation works OK
accrhs_rw = zeros(18,1);
accrhs_rw(1:17) = accrhs(1:17);
accrhs_rw(18) = accrhs(21); % same as pdd formulation

qdd_rw = inv(JAC_rw) * accrhs_rw;

% acceleration from rw formulation
rdd = qdd_rw(1:9);
r2dd = rdd(1:3);

```

```

r3dd = rdd(4:6);
r4dd = rdd(7:9);

wpd = qdd_rw(10:18);
w2pd = wpd(1:3);
w3pd = wpd(4:6);
w4pd = wpd(7:9);

p2dd = G2'*w2pd/2 - p2*w2p'*w2p/4;
p3dd = G3'*w3pd/2 - p3*w3p'*w3p/4;
p4dd = G4'*w4pd/2 - p4*w4p'*w4p/4;

test_rdd_rw = [ r2dd' r3dd' r4dd' ];
test_pdd_rw = [ p2dd' p3dd' p4dd' ];
test_wpd_rw = [ w2pd' w3pd' w4pd' ];

% use values
w2pdk = skew_sym( w2pd );
w3pdk = skew_sym( w3pd );
w4pdk = skew_sym( w4pd );

w4d = A4 * w4pd;
phidd = w4d(3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% jerk

% pddd formulation does NOT work
jrkrhs = zeros(21,1);
H2 = 2*w2psk*w2pdk + w2pdk*w2psk + w2psk*w2psk*w2psk;
H3 = 2*w3psk*w3pdk + w3pdk*w3psk + w3psk*w3psk*w3psk;
H4 = 2*w4psk*w4pdk + w4pdk*w4psk + w4psk*w4psk*w4psk;

% revolute A - j=1, i=2
jrkrhs(1:3) = A2*H2*s2pA;
jrkrhs(4) = -fp'*A1'*A2*H2*gp;
jrkrhs(5) = -fp'*A1'*A2*H2*hp;

% spherical B - j=2, i=3
jrkrhs(6:8) = A3*H3*s3pB - A2*H2*s2pB;

% universal C - j=3, i=4
jrkrhs(9:11) = A4*H4*s4pC - A3*H3*s3pC;

jrkrhs(12) = -fp'*A3'*A4*H4*fp ...
             -fp'*A4'*A3*H3*fp ...
             -3*fp'*(w3psk*w3psk-w3pdk)*A3'*A4*w4psk*fp ...
             -3*fp'*(w4psk*w4psk-w4pdk)*A4'*A3*w3psk*fp;

% revolute D - j=1, i=4
jrkrhs(13:15) = A4*H4*s4pD;
jrkrhs(16) = -hp'*A1'*A4*H4*fp;
jrkrhs(17) = -hp'*A1'*A4*H4*gp;

jrkrhs(18) = -6*p2dd'*p2d;
jrkrhs(19) = -6*p3dd'*p3d;
jrkrhs(20) = -6*p4dd'*p4d;

% fixed revolute driver at A for link 2 about u2
jrkrhs(21) = -driver_speed*driver_speed*driver_speed/4; % should work

qddd = inv(JAC) * jrkrhs;

% jerk from pddd formulation - OK for rddd, does NOT work for pddd
r2ddd = qddd(1:3);
p2ddd = qddd(4:7);

r3ddd = qddd(8:10);
p3ddd = qddd(11:14);

r4ddd = qddd(15:17);

```

```

p4ddd = qddd(18:21);

w2pdd = 2*G2*p2ddd - w2psk*w2pd/2 + w2p*w2p'*w2p/4;
w3pdd = 2*G3*p3ddd - w3psk*w3pd/2 + w3p*w3p'*w3p/4;
w4pdd = 2*G4*p4ddd - w4psk*w4pd/2 + w4p*w4p'*w4p/4;

test_pddd_p = [ p2ddd' p3ddd' p4ddd' ];
test_wpdd_p = [ w2pdd' w3pdd' w4pdd' ];
test_rddd_p = [ r2ddd' r3ddd' r4ddd' ];

% rw formulation works OK
jrkrhs_rw = zeros(18,1);
jrkrhs_rw(1:17) = jrkrhs(1:17);
jrkrhs_rw(18) = 0; % different from pddd formulation

% fixed revolute rotation driver at A for link 2 about u2
jrkrhs_rw(18) = 0;

qddd_rw = inv(JAC_rw) * jrkrhs_rw;

% jerk from rw formulation
rddd = qdd_rw(1:9);
r2ddd = rddd(1:3);
r3ddd = rddd(4:6);
r4ddd = rddd(7:9);

wpdd = qddd_rw(10:18);
w2pdd = wpdd(1:3);
w3pdd = wpdd(4:6);
w4pdd = wpdd(7:9);

p2ddd = G2'*( w2pdd +w2psk*w2pd/2 -w2p*w2p'*w2p/4 )/2 -3*p2*w2p'*w2pd/4;
p3ddd = G3'*( w3pdd +w3psk*w3pd/2 -w3p*w3p'*w3p/4 )/2 -3*p3*w3p'*w3pd/4;
p4ddd = G4'*( w4pdd +w4psk*w4pd/2 -w4p*w4p'*w4p/4 )/2 -3*p4*w4p'*w4pd/4;

test_pddd_rw = [ p2ddd' p3ddd' p4ddd' ];
test_wpdd_rw = [ w2pdd' w3pdd' w4pdd' ];
test_rddd_rw = [ r2ddd' r3ddd' r4ddd' ];

%del_rddd = test_rddd_p - test_rddd_rw % works OK
%del_pddd = test_pddd_p - test_pddd_rw
%del_wpdd = test_wpdd_p - test_wpdd_rw

% use pddd from rw to test jerkrhs
% R=OK, U=no, S=no, R=OK, p=OK, driver=OK
qddd_test = [ r2ddd ; p2ddd ; r3ddd ; p3ddd ; r4ddd ; p4ddd ];
jrkrhs_test = JAC * qddd_test;

%[ jrkrhs jrkrhs_test ]
%pause

% use values
w4dd = A4 * w4pdd;
phidd = w4dd(3);

% bottom - rsur_kin

```