The Development and Implementation of the Kennedy Space Center Umbilical Clearance Tool

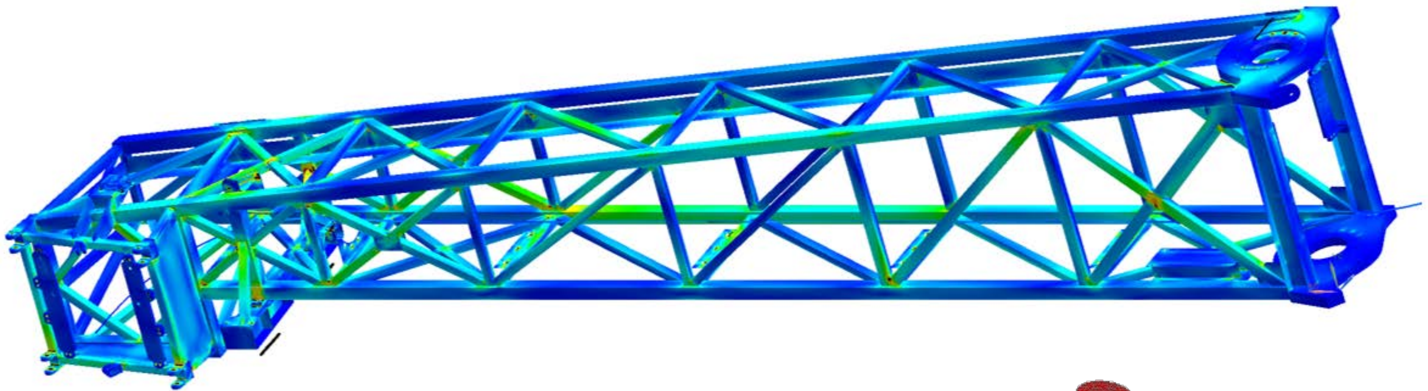
By David Chesnutt
Structures Analyst
Kennedy Space Center
Mail Code: NE-M6

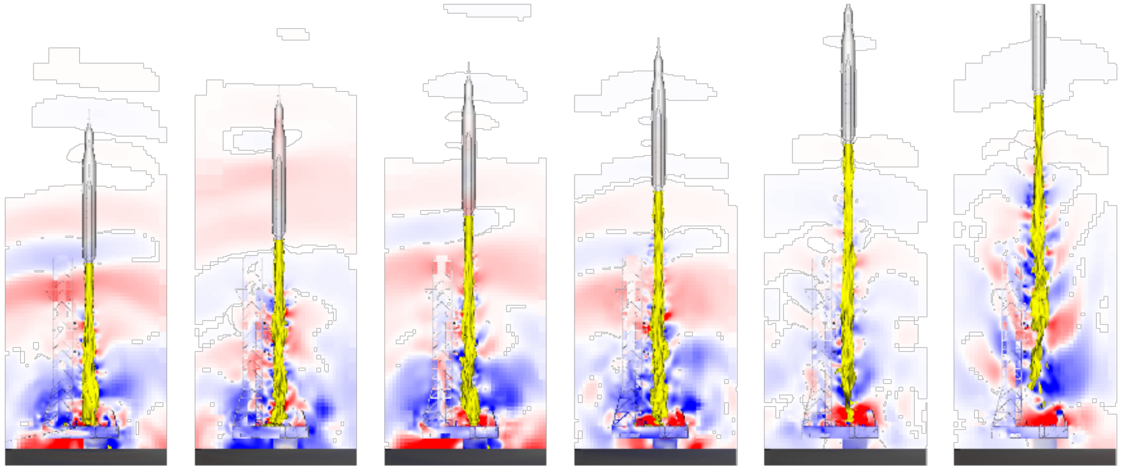
Presentation Overview

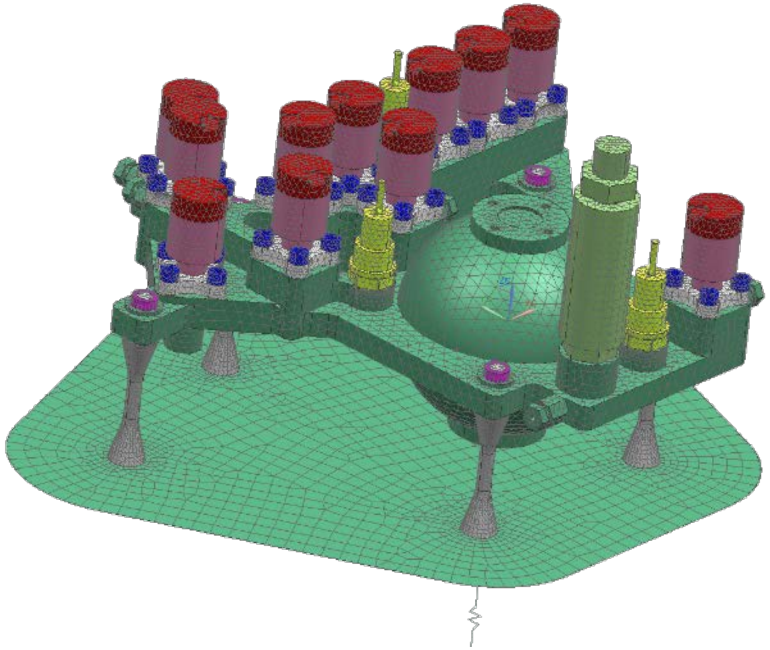
- Engineering Analysis Branch Introduction
- Project Background
- Umbilical Clearance Tool Development
- Underlying Analysis Theory
- Analysis Tool Plots

Analysis Group Background

- KSC NE Engineering Analysis Branch
  - Disciplines
    - Structures
    - Dynamics
    - Fluids
    - Thermal







Background

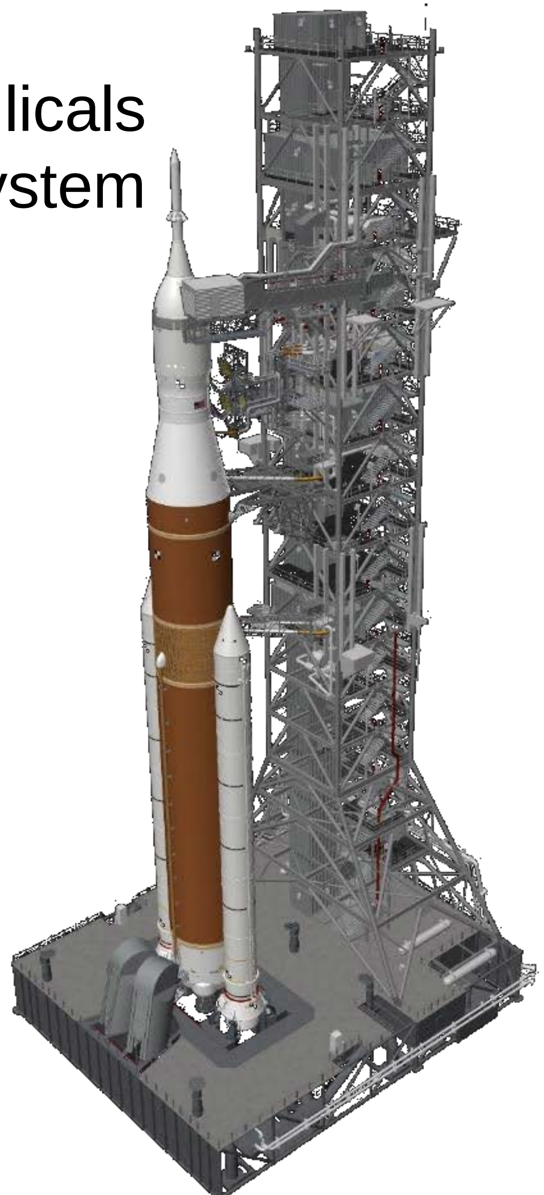
- NASA developing heavy lift launch vehicle dedicated to increasing the space launch capabilities of the nation
  - Flight vehicle named Space Launch System (SLS)

Source: NASA

Source: NASA

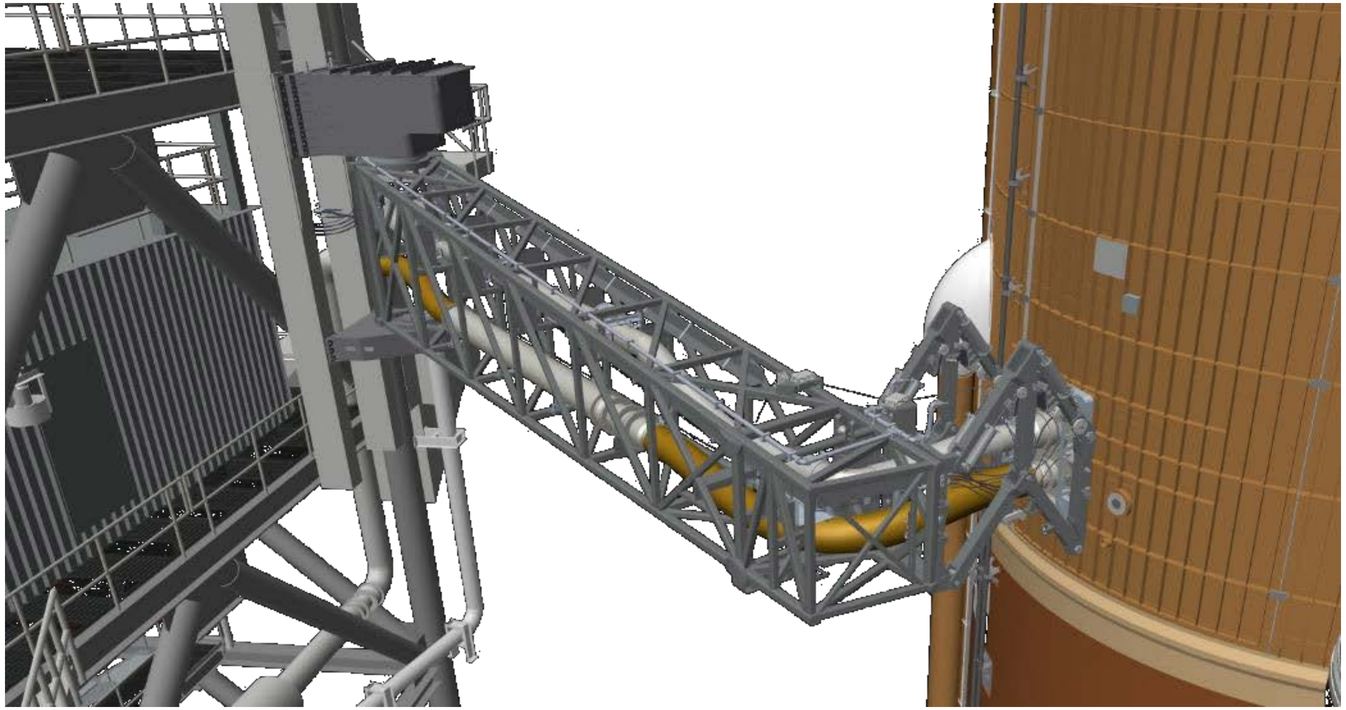
Background

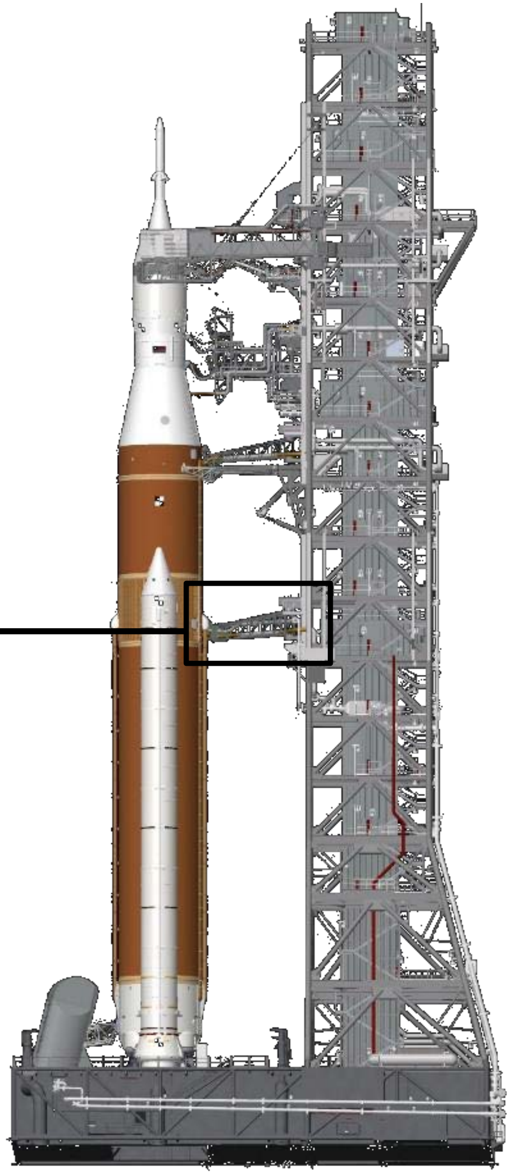
- Kennedy Space Center developing subsystems, know as umbilicals to provide interfaces and commodities to the Space Launch System (SLS) flight vehicle
- Umbilicals can provide:
  - Mechanical Support
  - Communications
  - Conditioned Air
  - Purges
  - Power
  - Fuel



Background

- The Core Stage Inner Tank Umbilical (CSITU) is a swing arm located at the top of the SLS first stage LH2 tank and provides…
  - Gaseous Hydrogen Vent Line
  - Conditioned Air
  - Gaseous Helium and Nitrogen
  - Electric Power/Control





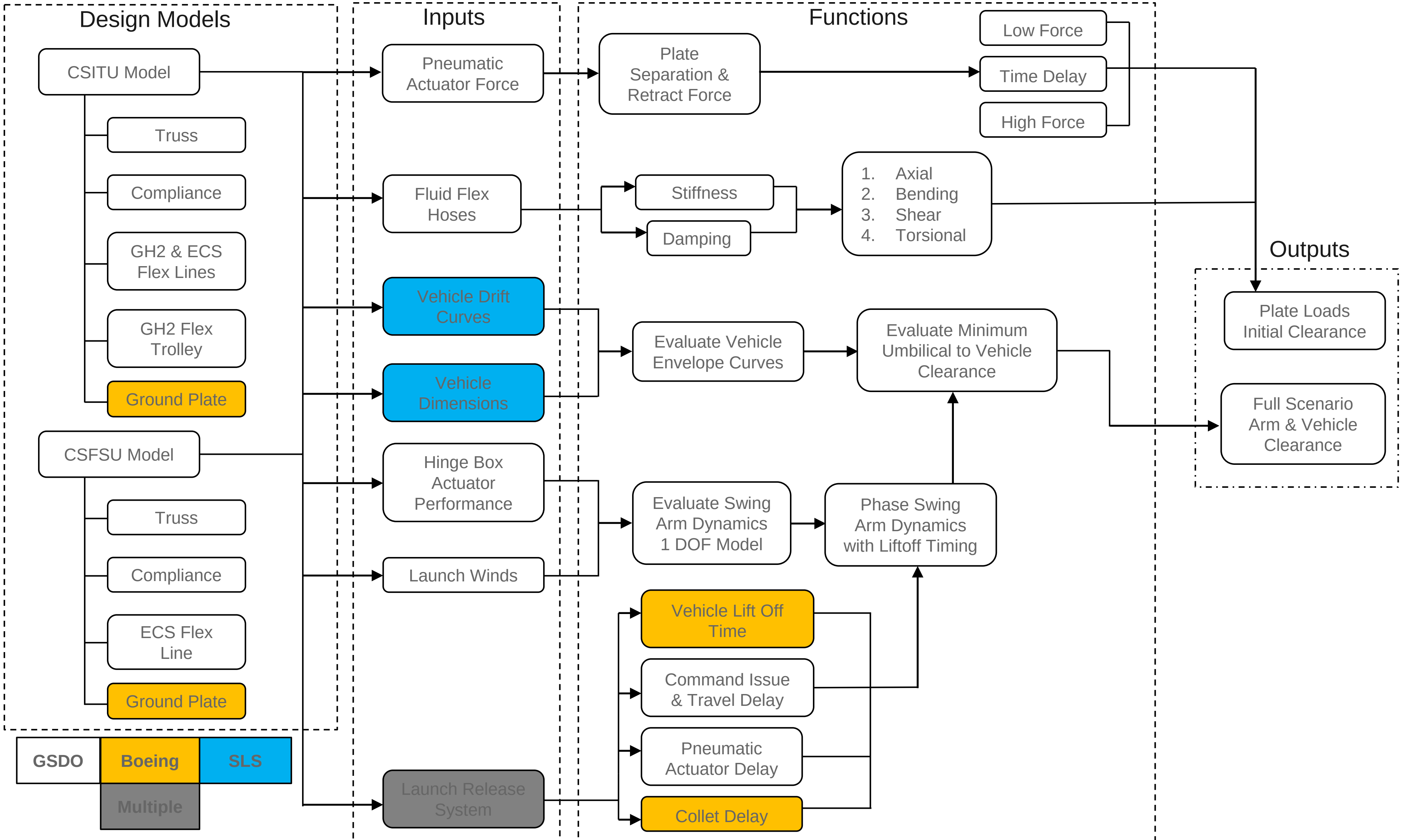
Background

- During launch umbilical arms, upon separation, required to never re-contact vehicle
- KSC required to verify re-contact requirement

Background

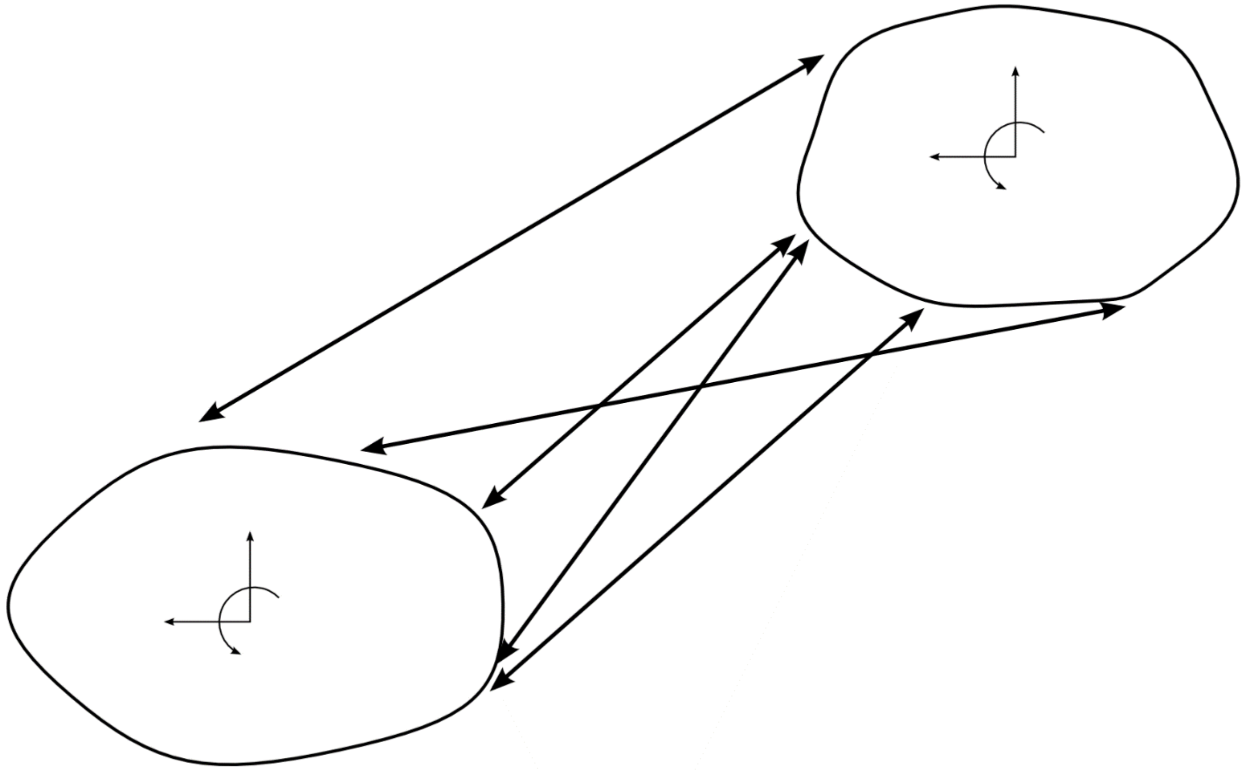
- With supplied DAC3R flight data from Marshall Space Flight Center (MSFC), and in house umbilical specifications, KSC has data to perform analysis
- Result: KSC Umbilical Clearance Tool (UCT) undertaken

Analysis Flow



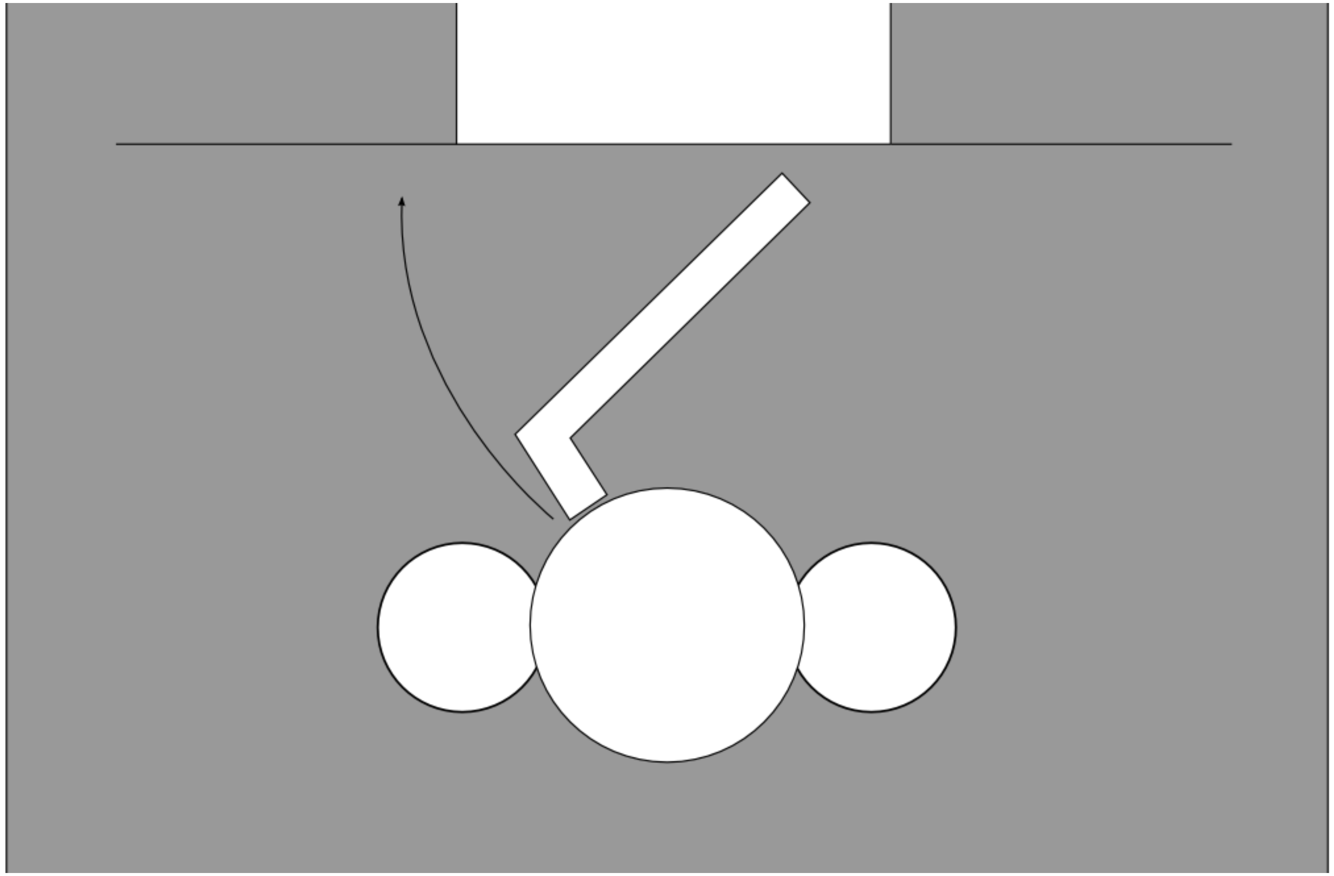
Tool Development

- Computationally intensive to compare 3D bodies moving as points of interest constantly change



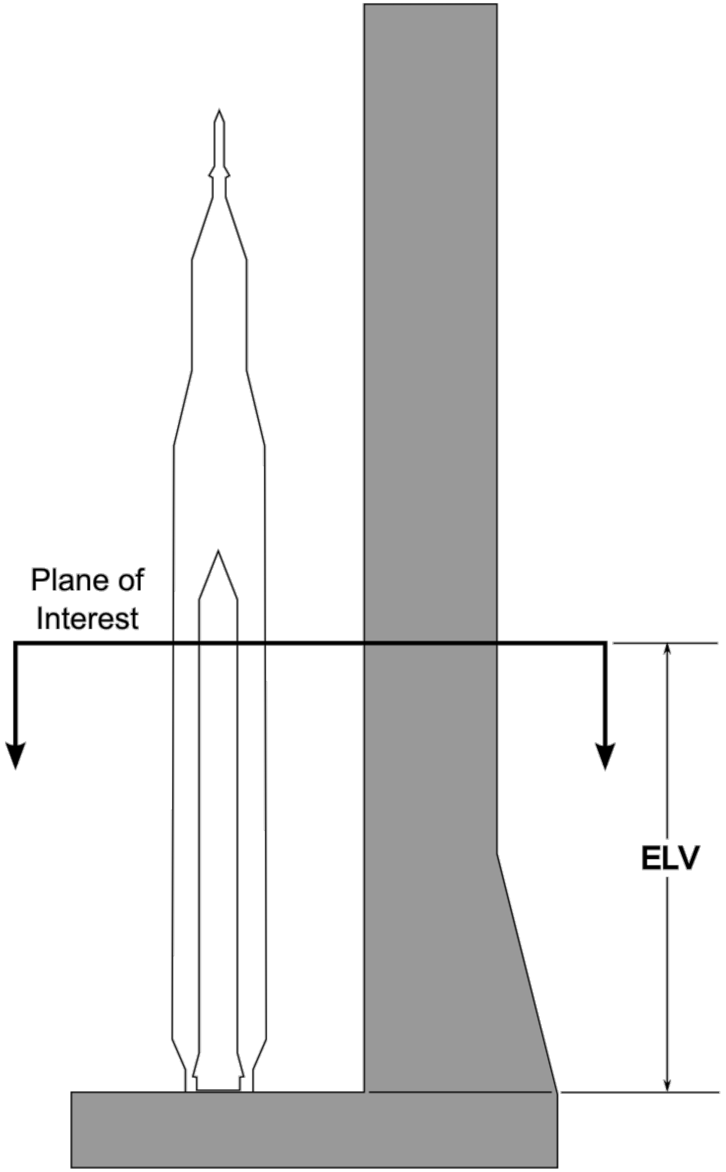
Tool Development

- A number of umbilical arms pivot about a vertical axis
  - Umbilicals of this nature called swing arms



Tool Development

- Restricted motion of swings arms can be used to simplify clearance analysis to a plane of interest



Tool Development

- 6 Step Approach to Evaluate a Data Set:
  - Extract relevant input from Marshal DAC3R raw data
  - Generate the flight vehicle projected points with respect to time for each simulation of each data set
  - Determine outlier simulations where flight vehicle is furthest from launch origin with respect to time

Tool Development

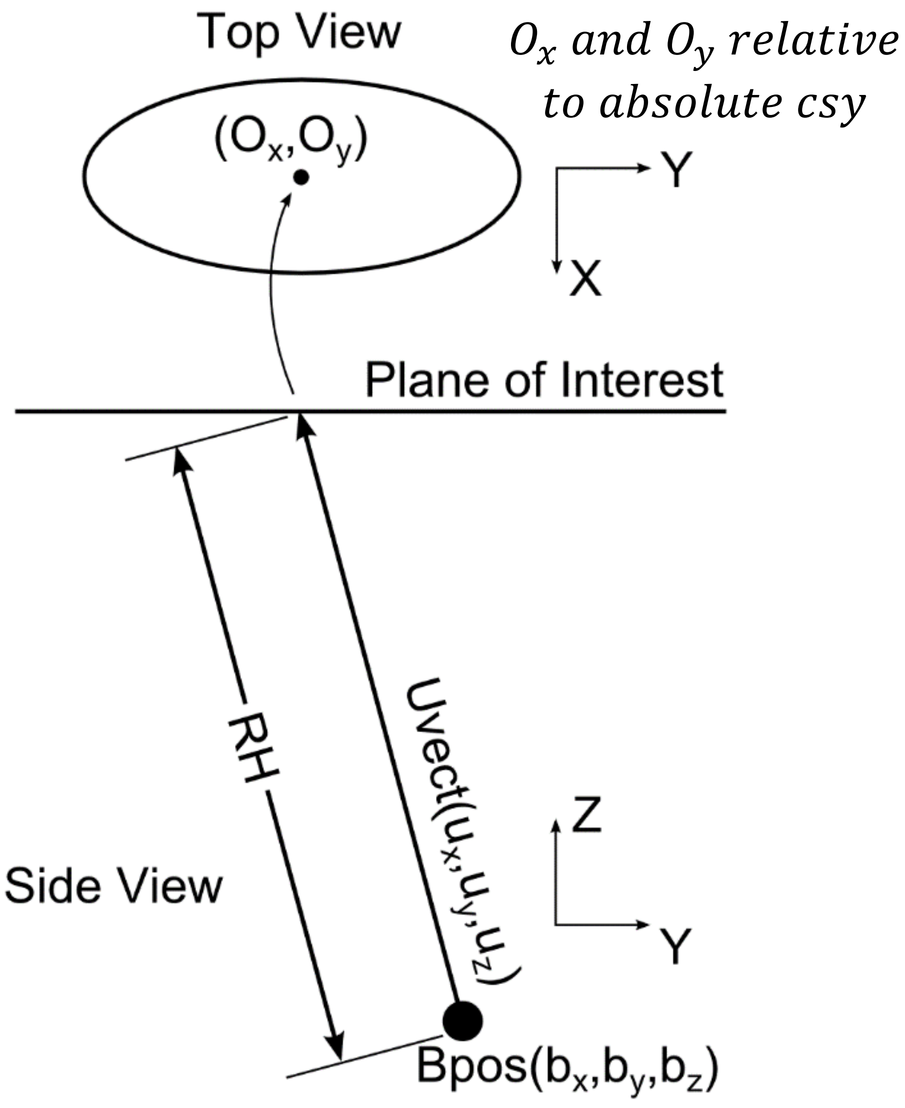
- 6 Step Approach to Evaluate a Data Set:
  - From outlier simulations generate points to define flight vehicle skin with respect to time
  - Envelope flight vehicle skin points with respect to time
  - Compare umbilical retract relative to envelope curves

Load Raw Data

- For each simulation file the function extracts:
  - Time of launch simulation
  - Core stage base point [relative to time]
  - Unit vector pointing along the core stage longitudinal axis [relative to time]
  - Solid Rocket Booster base points [relative to time]
- Extracted data is saved to a .mat file

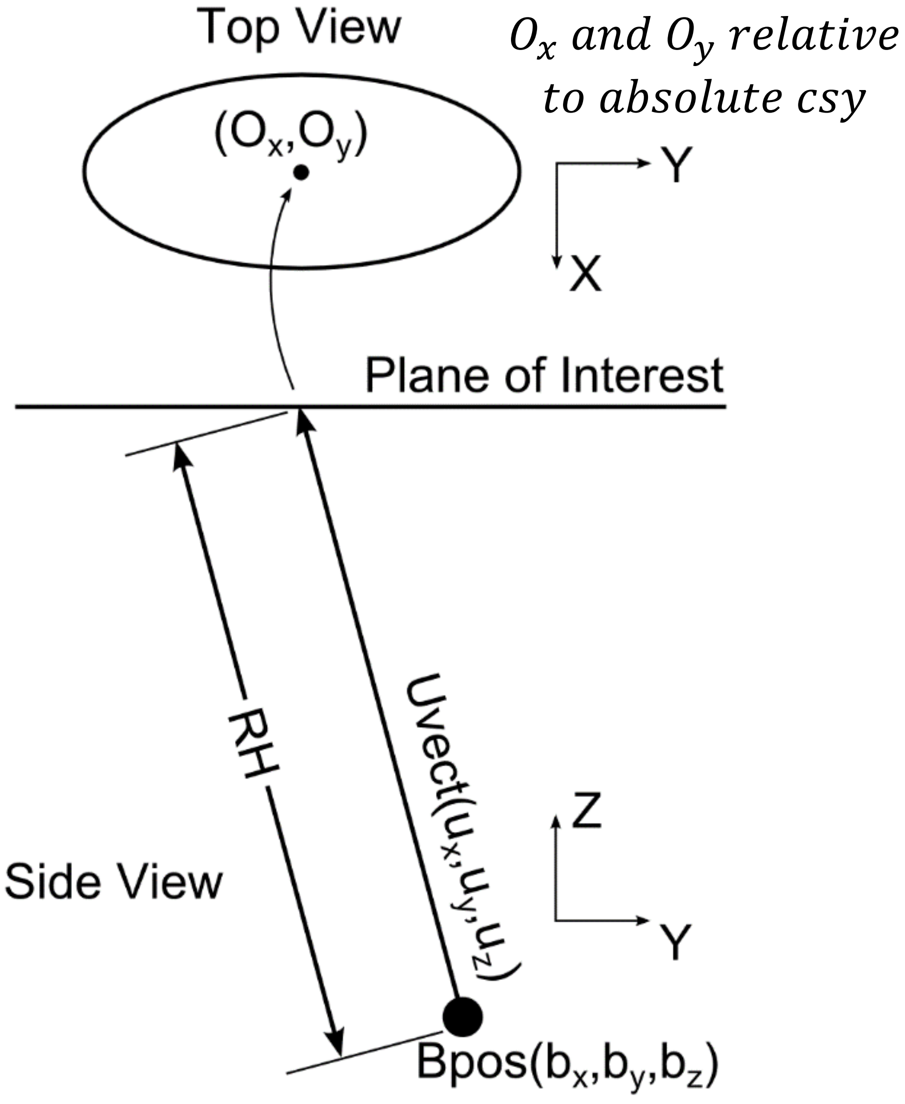
Determine Projected Points

- Calculate the coordinates of the intersection between the plane of interest and the flight vehicle unit vector.



Determine Projected Points

- RH is calculated from:
  - Plane of interest elevation (ELV)
  - Base position (Bpos)
  - Flight vehicle unit vector (Uvect)
- RH used for:
  - Lookup function for vehicle skin diameter at plane of interest
  - Calculating location of O_x & O_y

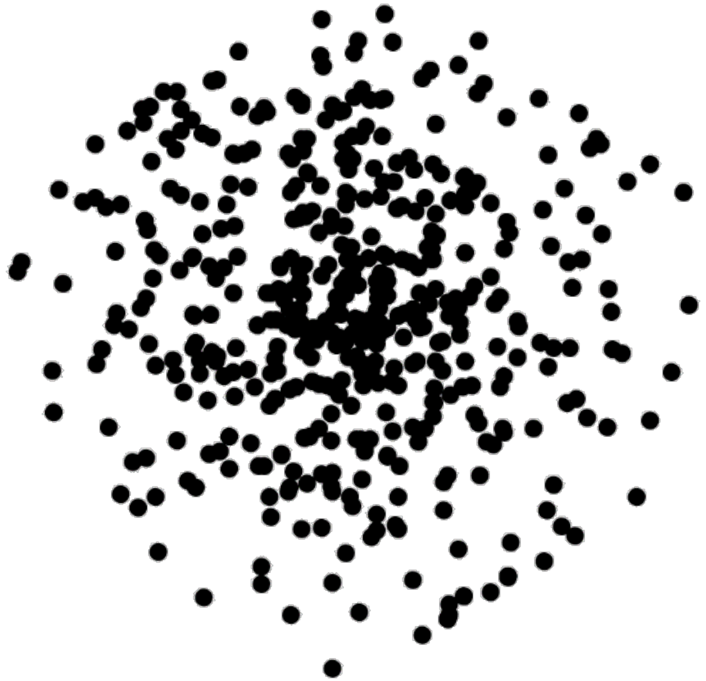


$$RH = \frac{ELV - b_z}{u_z} \qquad O_x = RH \cdot u_x + b_x \qquad O_y = RH \cdot u_y + b_y$$

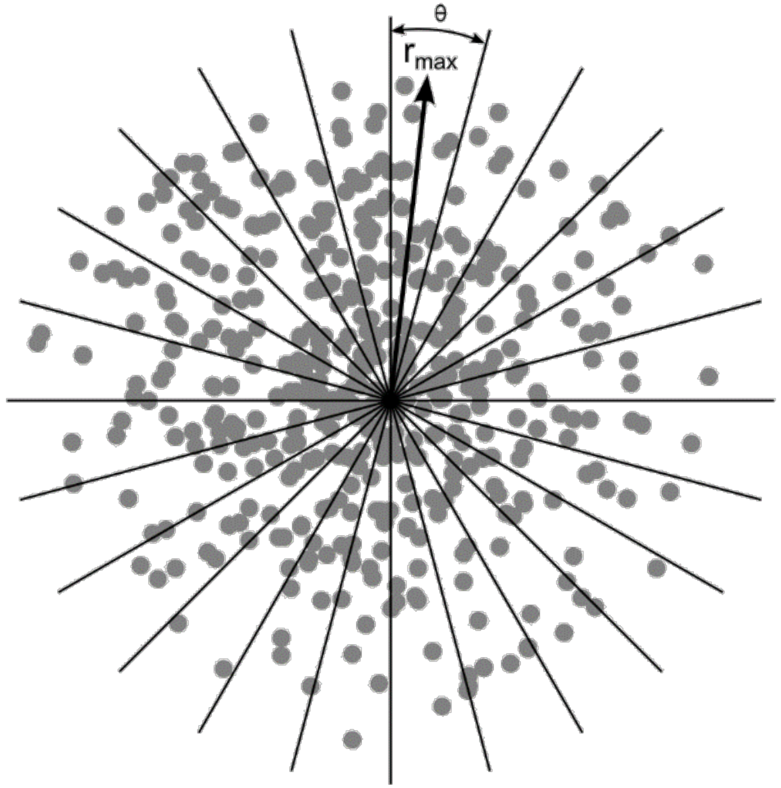
Envelope Creation

- For each time step combine projected base point results

Projected Base Points



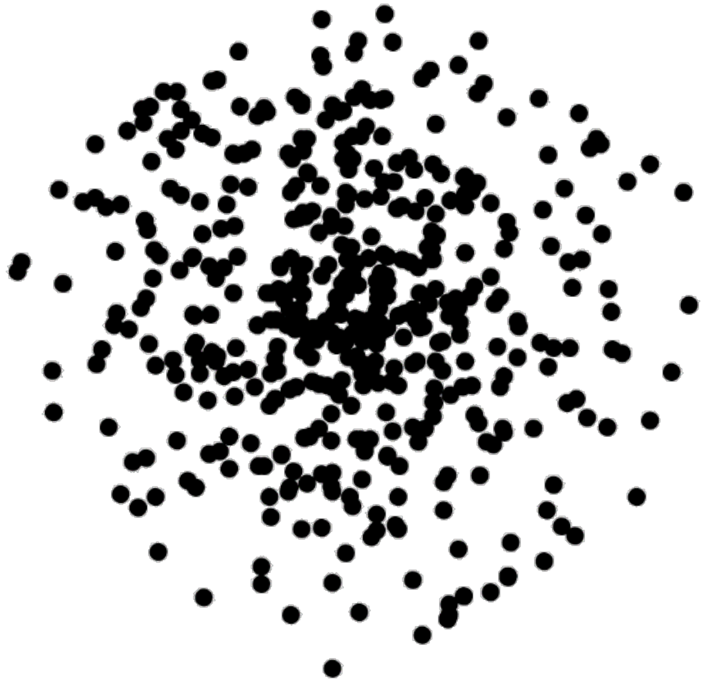
Envelope Regions



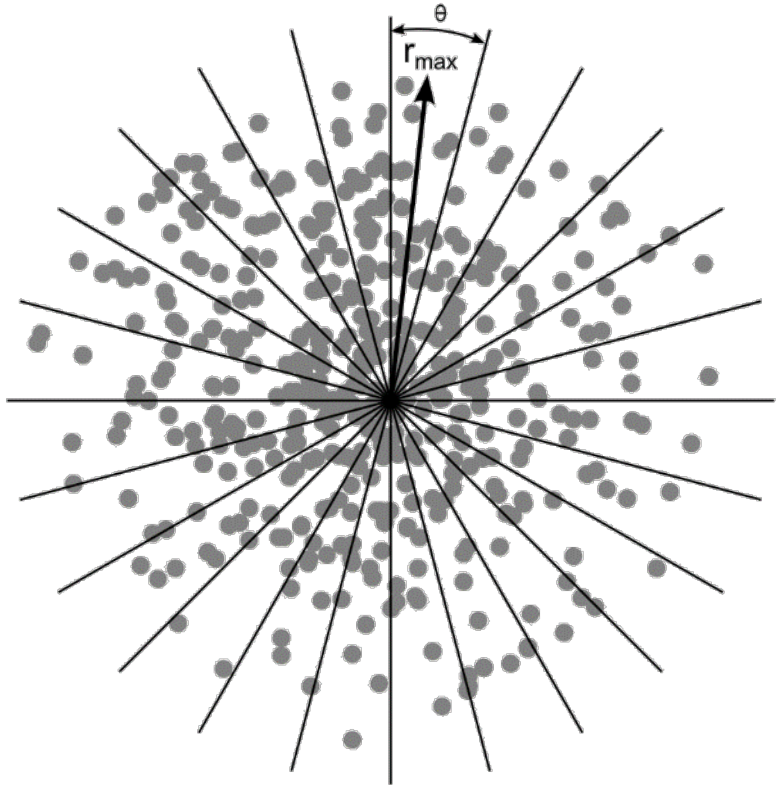
Envelope Creation

- Segment point data into angular regions
- Find n # of r_{max} points in descending order per region

Projected Base Points



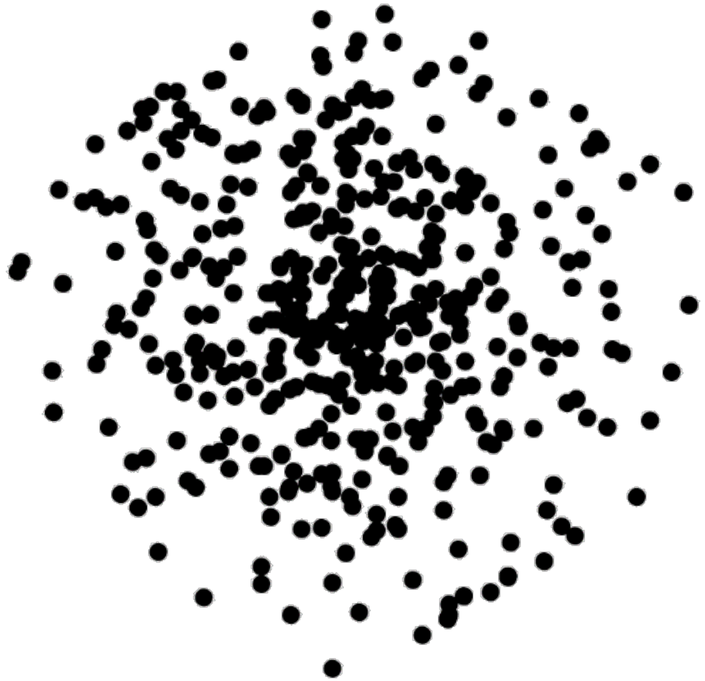
Envelope Regions



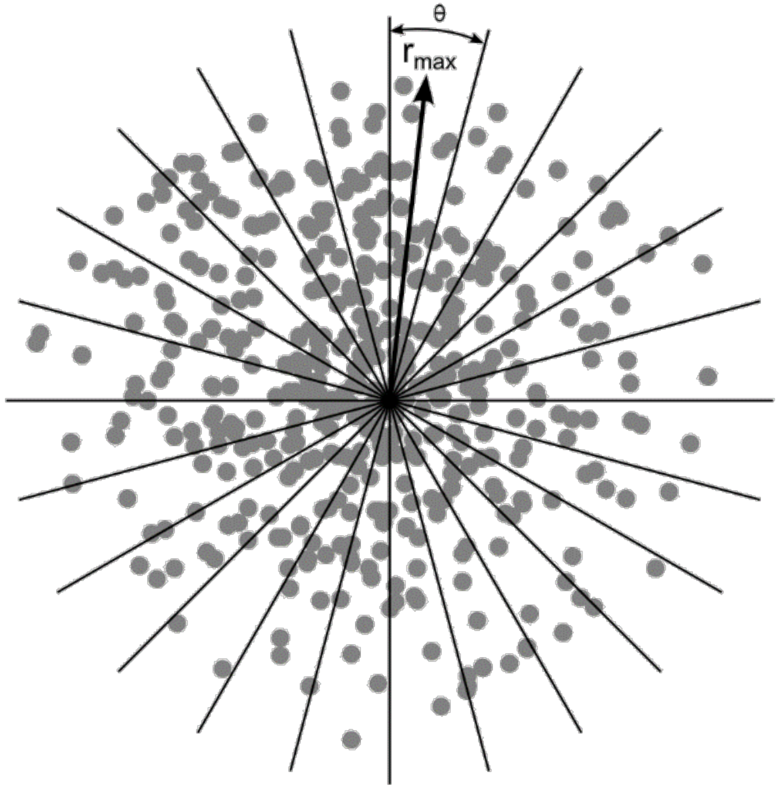
Envelope Creation

- Capture sets that correspond to outer most points
  - These sets represent simulations with the greatest drift

Projected Base Points

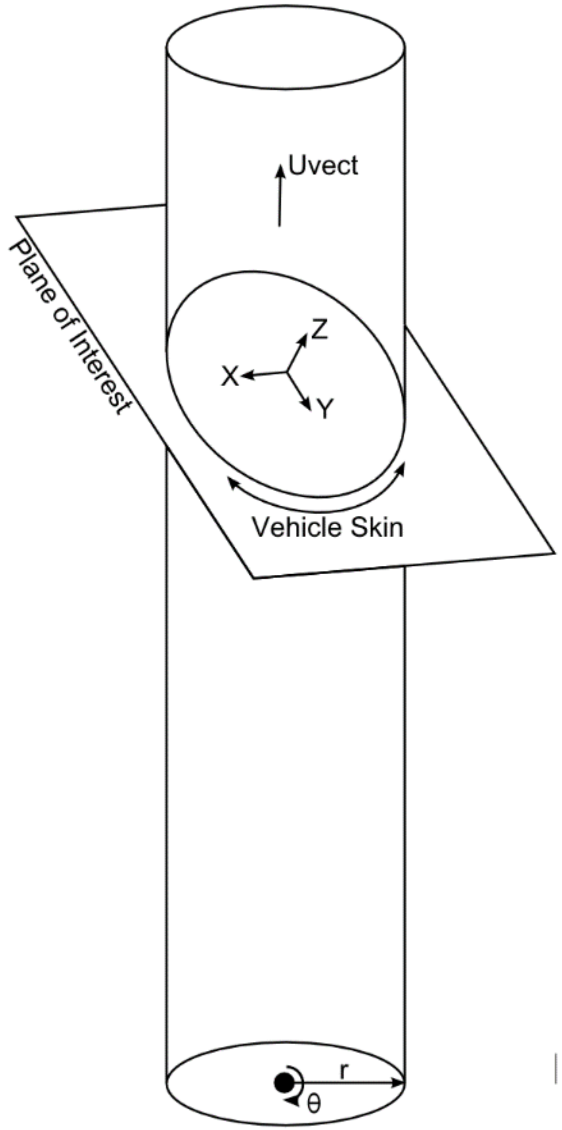


Envelope Regions



Flight Vehicle Skin Analysis

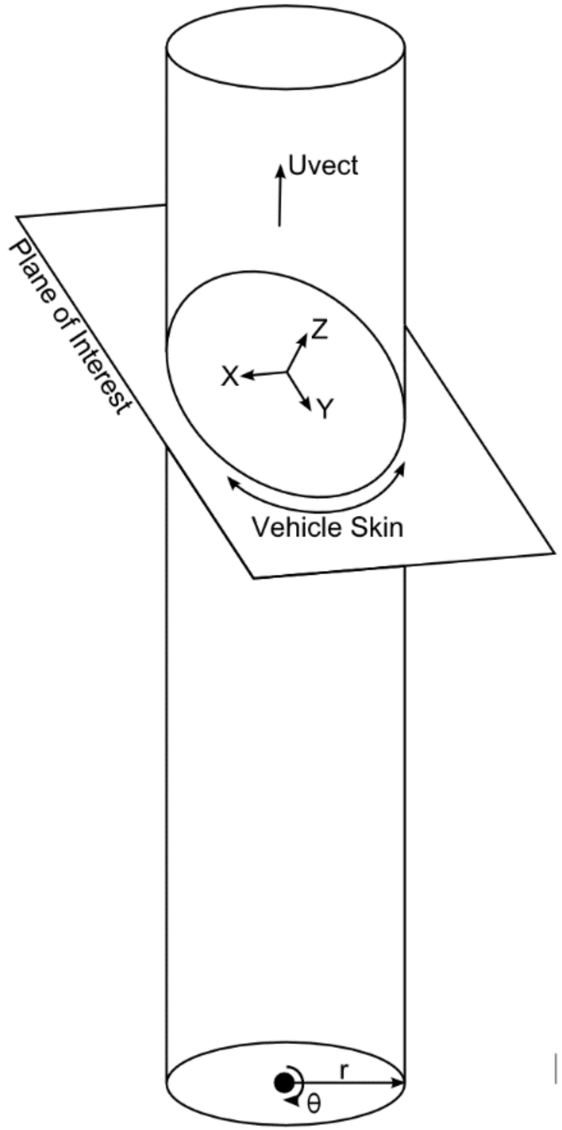
- Steps to calculate ellipse curve:
  1. Determine space fixed rotations to orient Uvect in up (Z) direction
  2. Within the flight vehicle linear space create the plane of interest equation
  3. Create a cylinder equation oriented up using flight vehicle radius at plane of interest



Flight Vehicle Skin Analysis

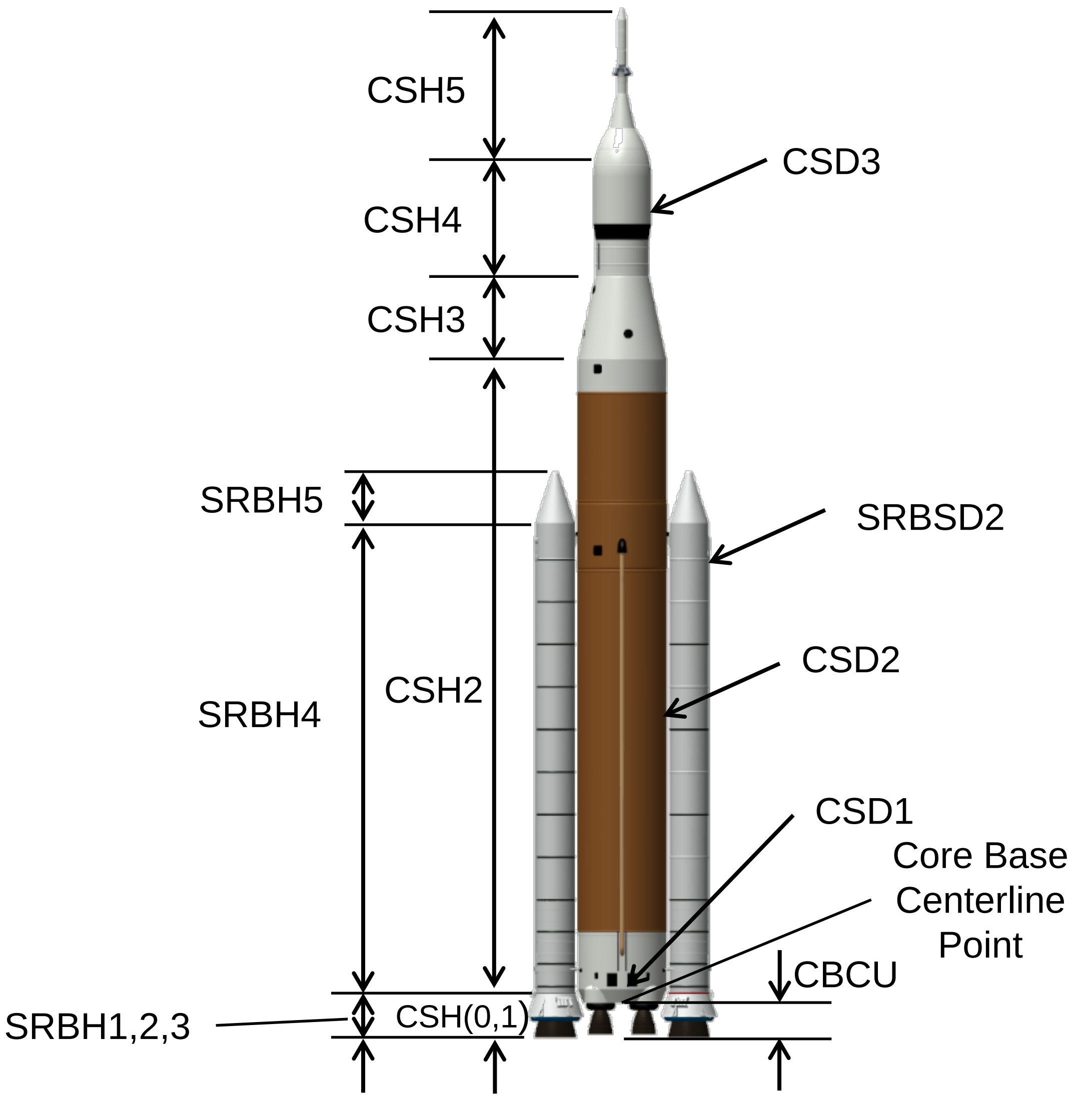
- Steps to calculate ellipse curve:
  4. Intersect the two equations to define vehicle skin linear space
  5. Transform vehicle skin linear space to global coordinate system using space fixed rotations calculated in step 1

$$\begin{bmatrix} s_x(r,\theta) \\ s_y(r,\theta) \\ s_z(r,\theta) \end{bmatrix}^{abs} = [R_x][R_y]\begin{bmatrix} s_x(r,\theta) \\ s_y(r,\theta) \\ s_z(r,\theta) \end{bmatrix}^{veh}$$



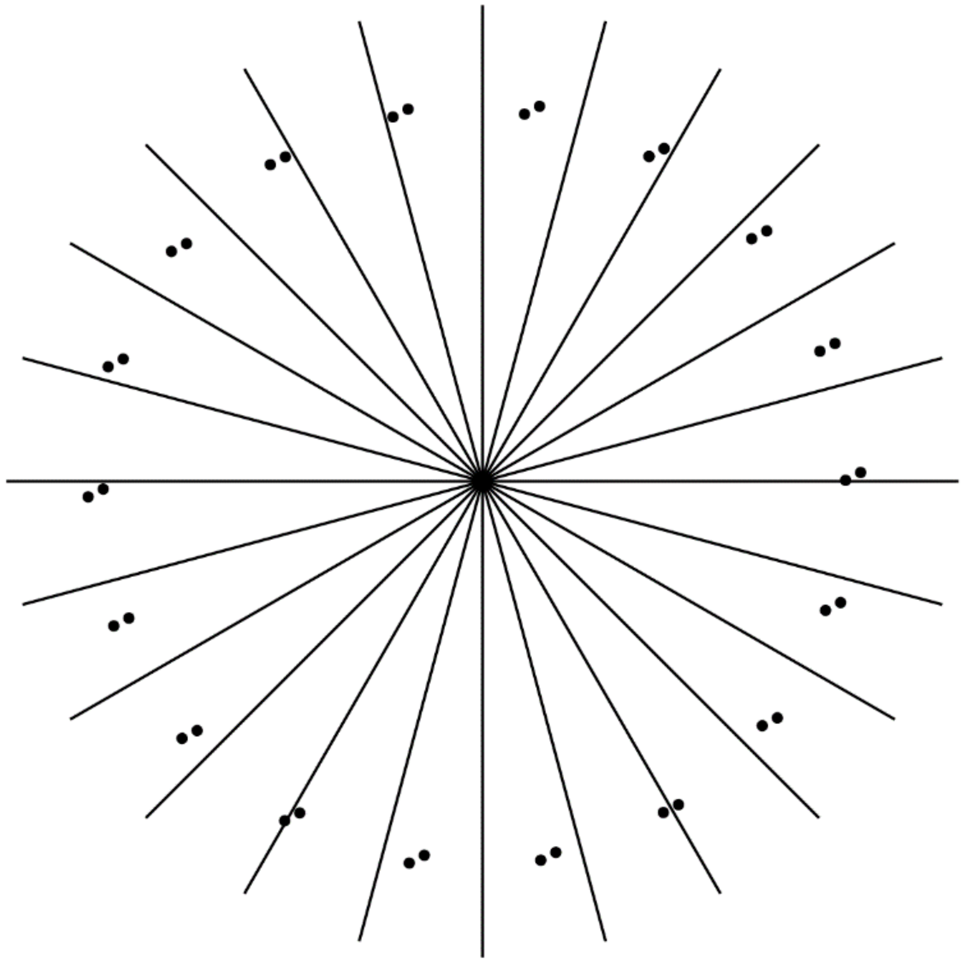
Lookup Parameters

- Radius value r in skin function determined from a look up function
  - Input: RH (Relative Height)
  - Output: r (vehicle skin radius)



Envelope Creation

- θ is left to the analyst to decide the number of points and their relative angle between them
- Dilemma arises if same θ values used for each data set
- Resolution:
  - Vary θ values using normalized simulation id as perturbation

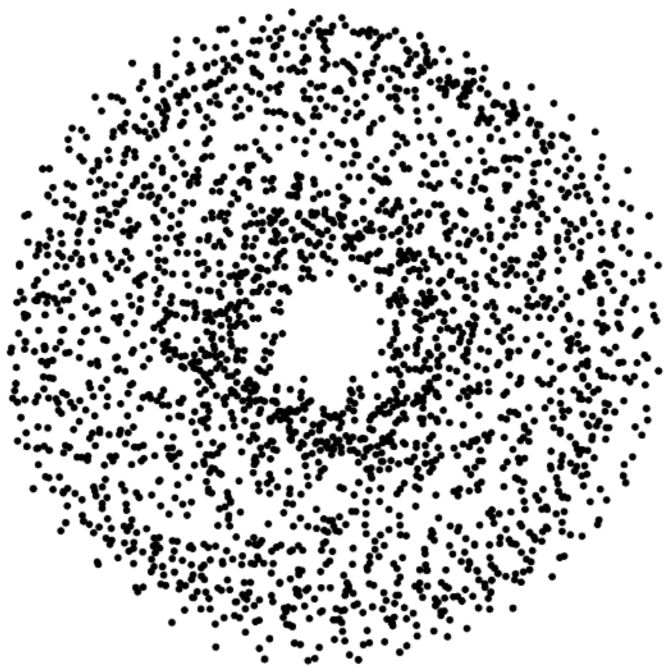


Vehicle skin points all initialize at some locations leaving empty envelope sectors

Envelope Creation

- Vehicle skin points generated for each outer drift point selected from initial envelope of projected base points

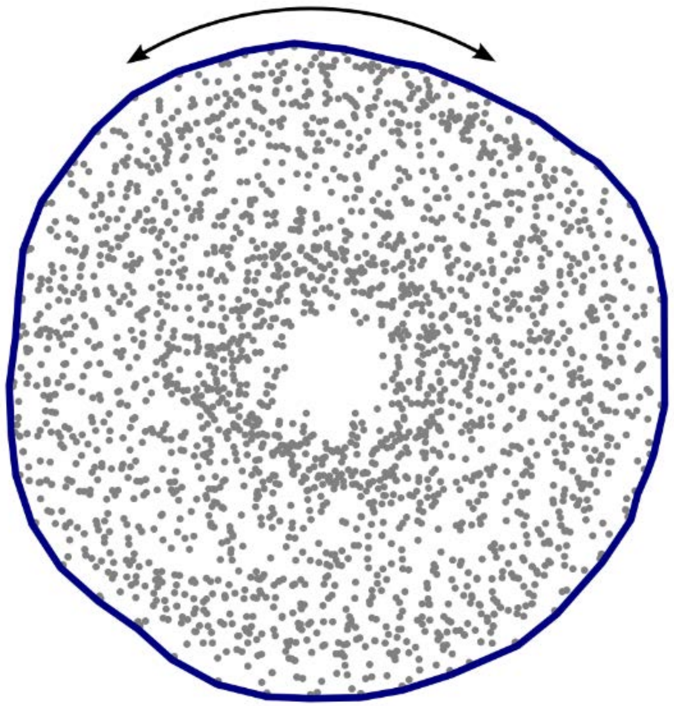
Time Slice Data



Envelope Creation

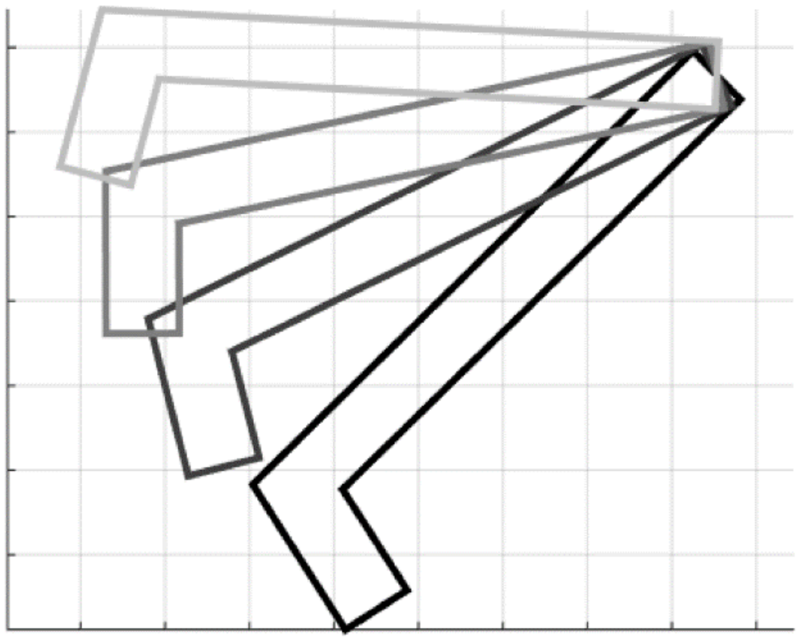
- Envelope algorithm used on vehicle skin points
- Vehicle envelope defined by outermost points

Envelope Curve



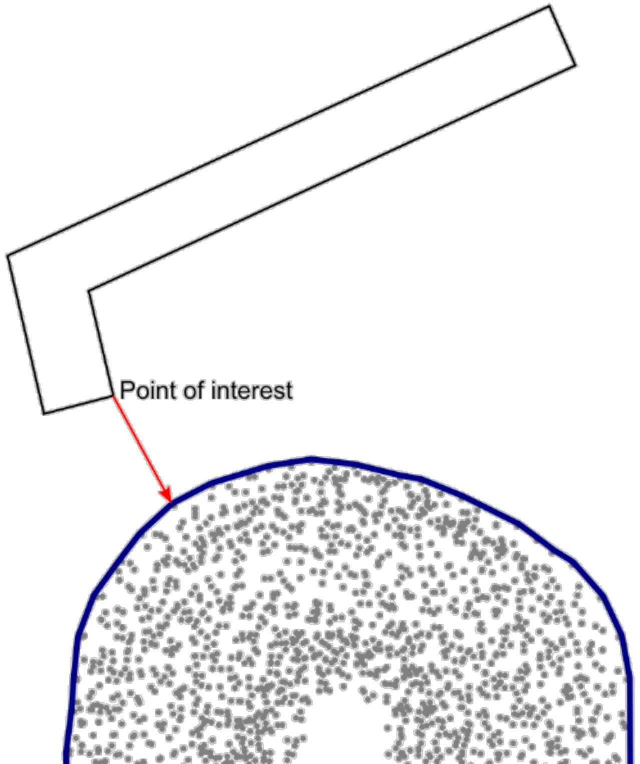
Generate Swing Arm Skin

- In separate analysis umbilical retract dynamics evaluated
  - Simulation Output: Umbilical orientation with respect to time
- Umbilical envelope used with orientation to determine points of interest

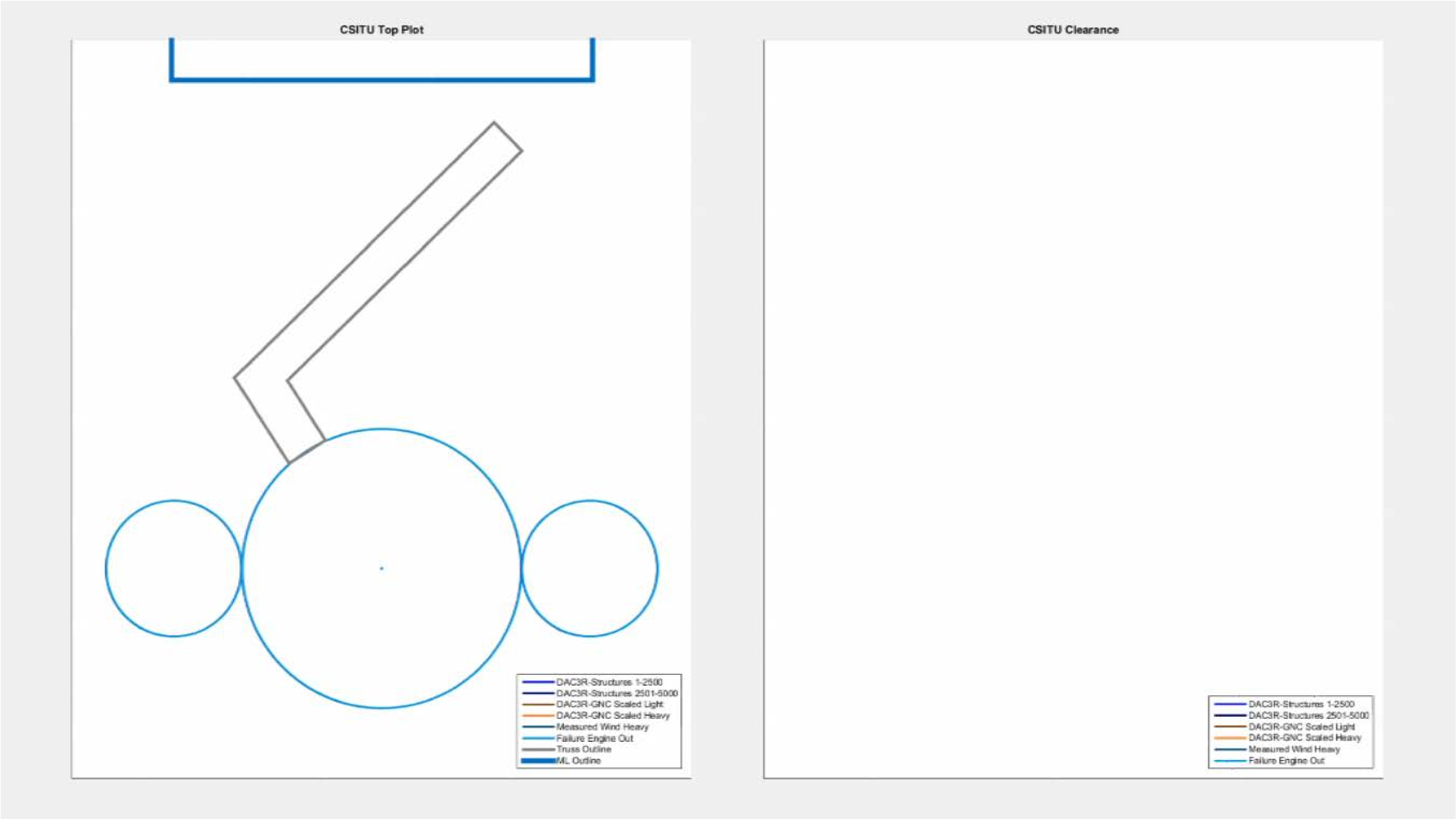


Calculate Clearance

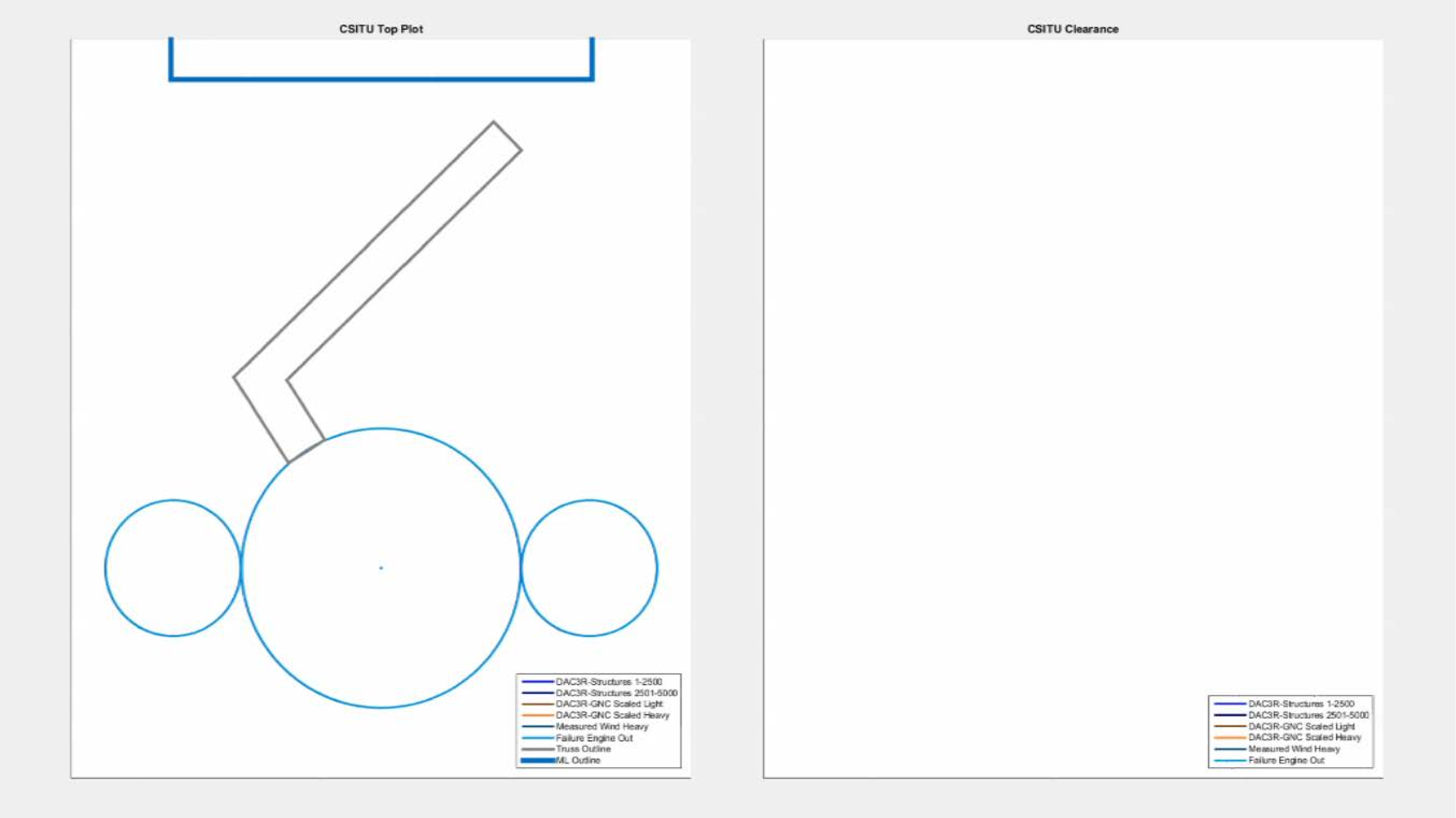
- In clearance simulation compare swing arm point of interest to vehicle skin envelop curves w.r.t time
- Find closest vehicle skin point to swing arm POI
  - – Clearance outputs:
    - Coordinates of POI and closest vehicle skin point
    - Clearance associated to two points
    - Simulation IDs corresponding min clearance points



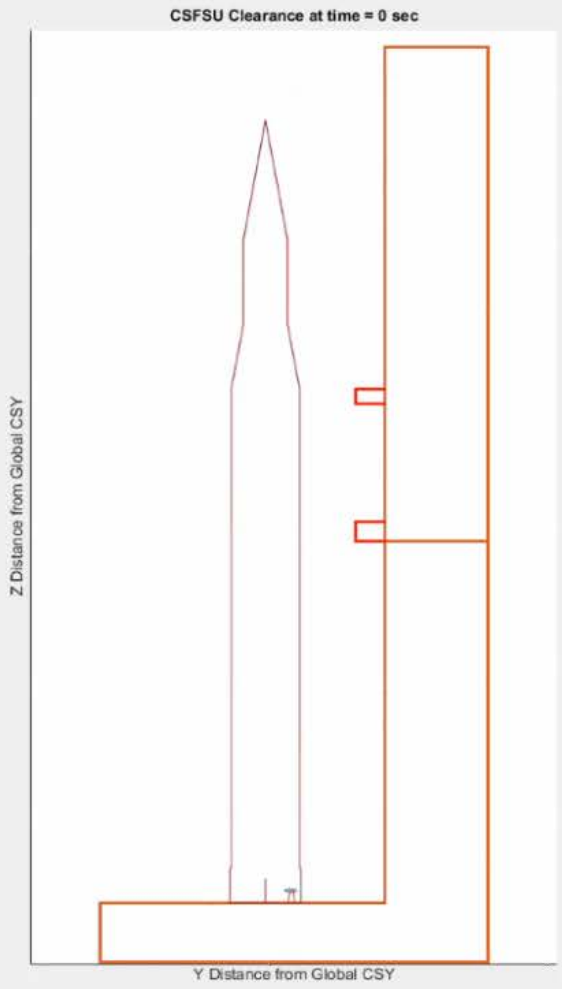
Clearance Plot Real Time



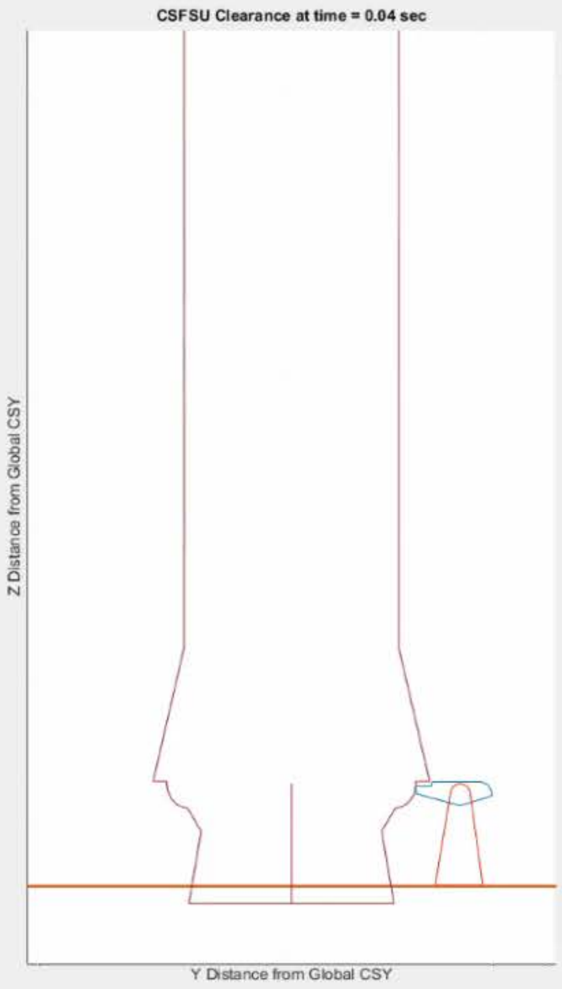
Clearance Plot Half Speed



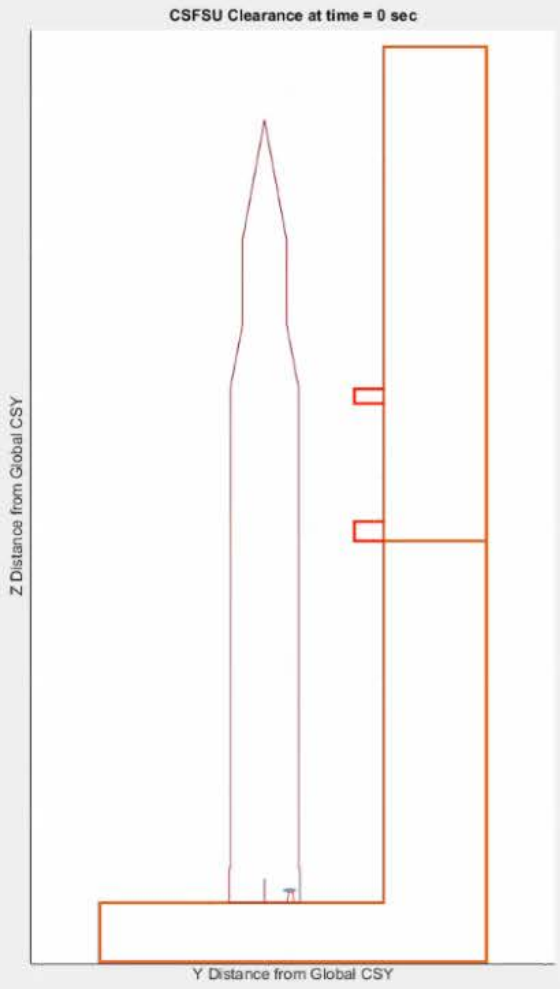
East View Check



Light/Fast



ASEU Retract



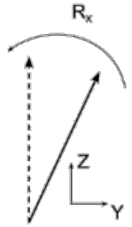
Heavy/Slow

BACKUP SLIDES

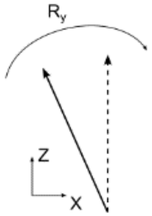
Flight Vehicle Skin Analysis

- MATLAB Function Name: scurve.m
- Steps to calculate ellipse curve:
  1. Determine space fixed rotations to orient Uvect in up (Z) direction

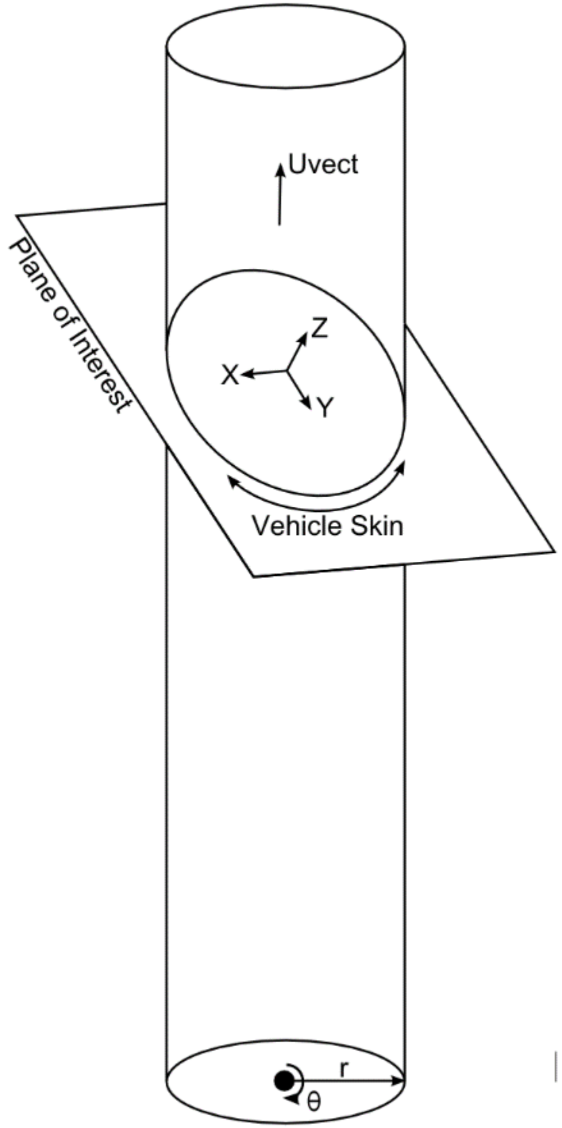
$$Uvect = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \qquad Find\ [R_x]\ such\ that: \begin{bmatrix} a' \\ 0 \\ c' \end{bmatrix} = [R_x] \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$



$$Find\ [R_y]\ such\ that: \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = [R_y] \begin{bmatrix} a' \\ 0 \\ c' \end{bmatrix}$$



$$[R_x] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & \sin(\theta_x) \\ 0 & -\sin(\theta_x) & \cos(\theta_x) \end{bmatrix}, [R_y] = \begin{bmatrix} \cos(\theta_y) & 0 & -sin(\theta_y) \\ 0 & 1 & 0 \\ sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix}$$



Flight Vehicle Skin Analysis

- Steps to calculate ellipse curve:
  2. Within the flight vehicle linear space create the plane of interest equation

$$\textit{Find plane of interest normal within flight vehicle linear space}$$

$$\begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix}^{veh} = \left[[R_x][R_y]\right]^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{abs} = \begin{bmatrix} cx & sxsy & cxsy \\ 0 & cx & -sx \\ -sy & sxcy & cxcy \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^{abs}$$

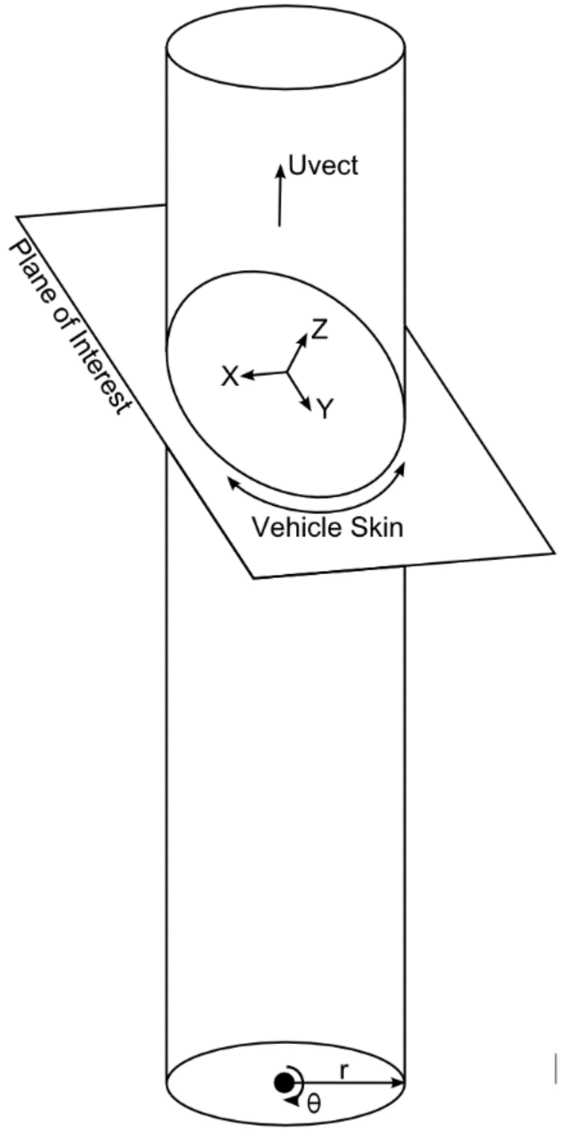
$$\textit{Create plane equation in flight vehicle linear space}$$

$$n_x x + n_y y + n_z z = 0$$

$$or$$

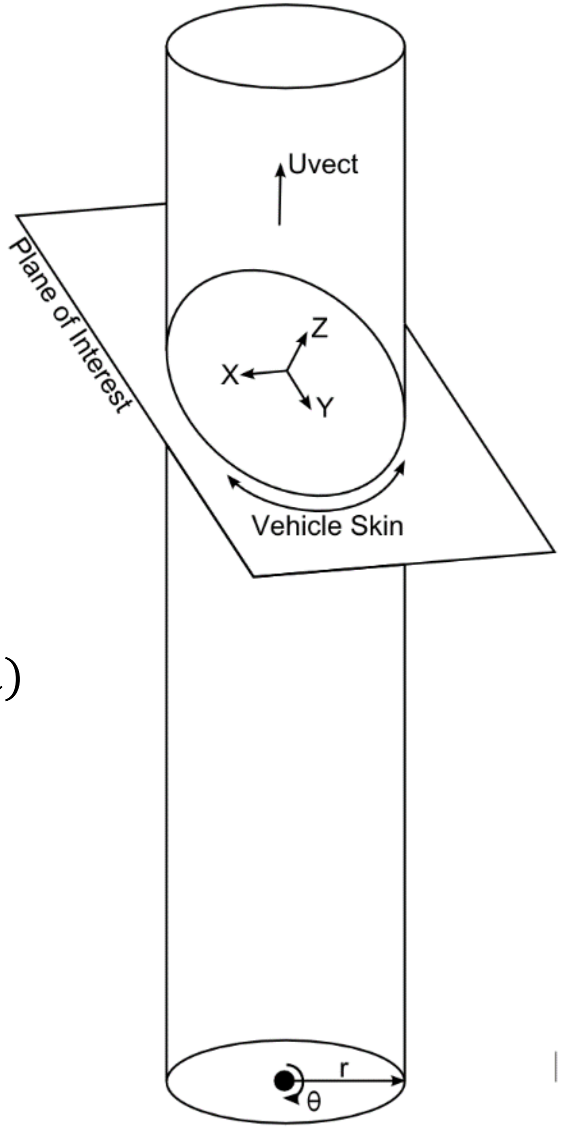
$$(cxsy)x + (-sx)y + (cxcy)z = 0$$

$$Note: sx, sy = \sin(\theta_x), \sin(\theta_y), \text{and } cx, cy = \cos(\theta_x), cos(\theta_y)$$



Flight Vehicle Skin Analysis

- Steps to calculate ellipse curve:
  3. Create a cylinder equation oriented up using vehicle radius at plane of interest



$$Define\ a\ subspace\ for\ a\ cylinder\ representing\ the\ rocket$$

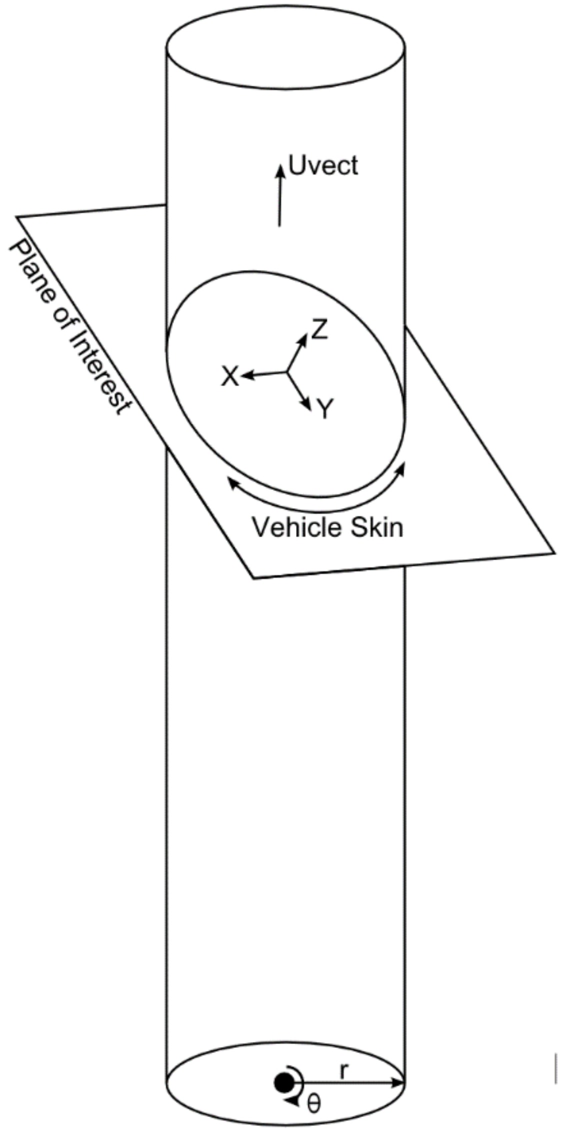
$$\begin{bmatrix} r * cos(\theta) \\ r * sin(\theta) \\ z \end{bmatrix}^{veh} \quad for\ r = rocket\ skin\ radius\ at\ plane\ of\ interest\ (rh\_lookup.m)$$

Flight Vehicle Skin Analysis

- Steps to calculate ellipse curve:
  4. Intersect the two equations to define vehicle skin linear space

$$Resulting\ \ subspace:$$

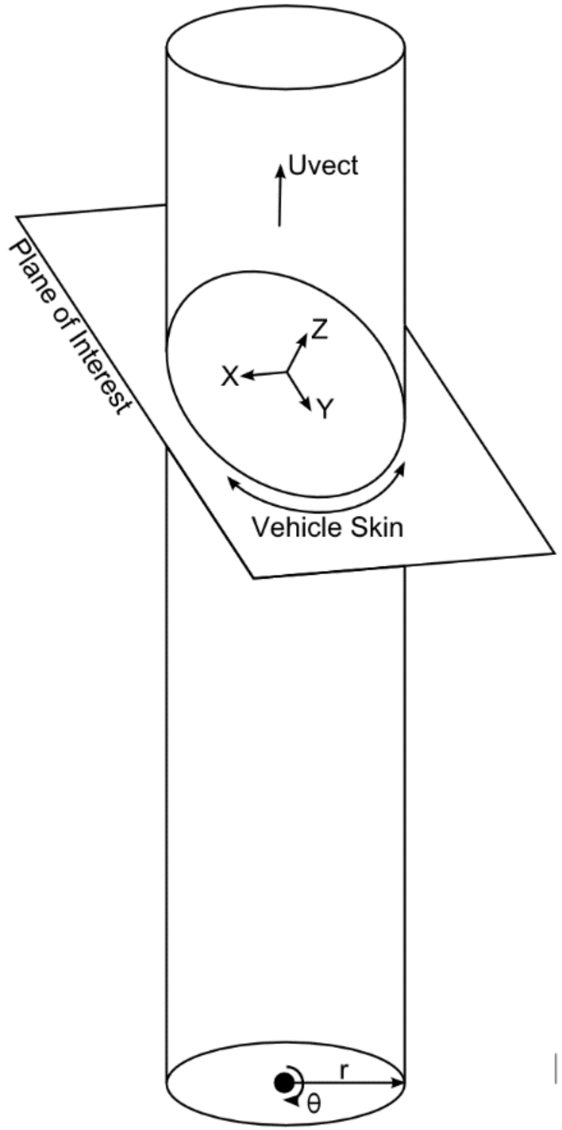
$$\begin{bmatrix} s_x(r,\theta) \\ s_y(r,\theta) \\ s_z(r,\theta) \end{bmatrix}^{veh} = \begin{bmatrix} \mathrm{r} * \cos(\theta) \\ \mathrm{r} * \sin(\theta) \\ \dfrac{cx * sy * r * \cos(\theta) - sx * r * \sin(\theta)}{-cx * cy} \end{bmatrix}^{veh}$$



Flight Vehicle Skin Analysis

- Steps to calculate ellipse curve:
  5. Transform vehicle skin linear space to global coordinate system using space fixed rotations calculated in step 1

$$\begin{bmatrix} s_x(r,\theta) \\ s_y(r,\theta) \\ s_z(r,\theta) \end{bmatrix}^{abs} = [R_x][R_y]\begin{bmatrix} s_x(r,\theta) \\ s_y(r,\theta) \\ s_z(r,\theta) \end{bmatrix}^{veh}$$

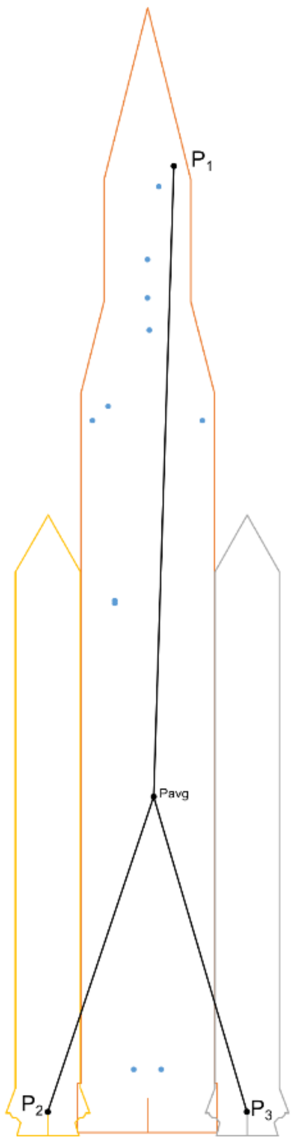


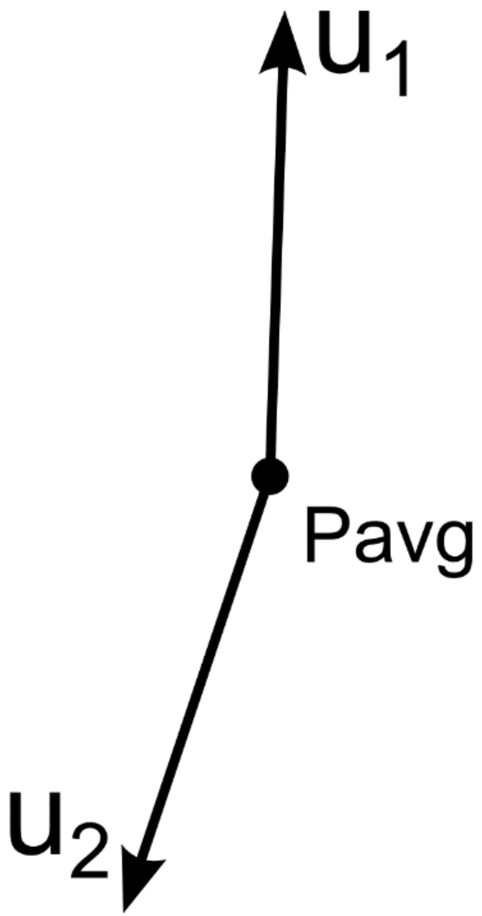
Marshall Data Synthesis

- The supplied structures data from the August 2015 technical interchange meeting is in the form of a set of points moving relative to their nominal position with respect to time.
- Each point represents a flex body grid point location in the SLS X, Y, and Z coordinate frame.
- Task:
  - Determine vehicle position and orientation with respect to time from point data

Marshall Data Synthesis

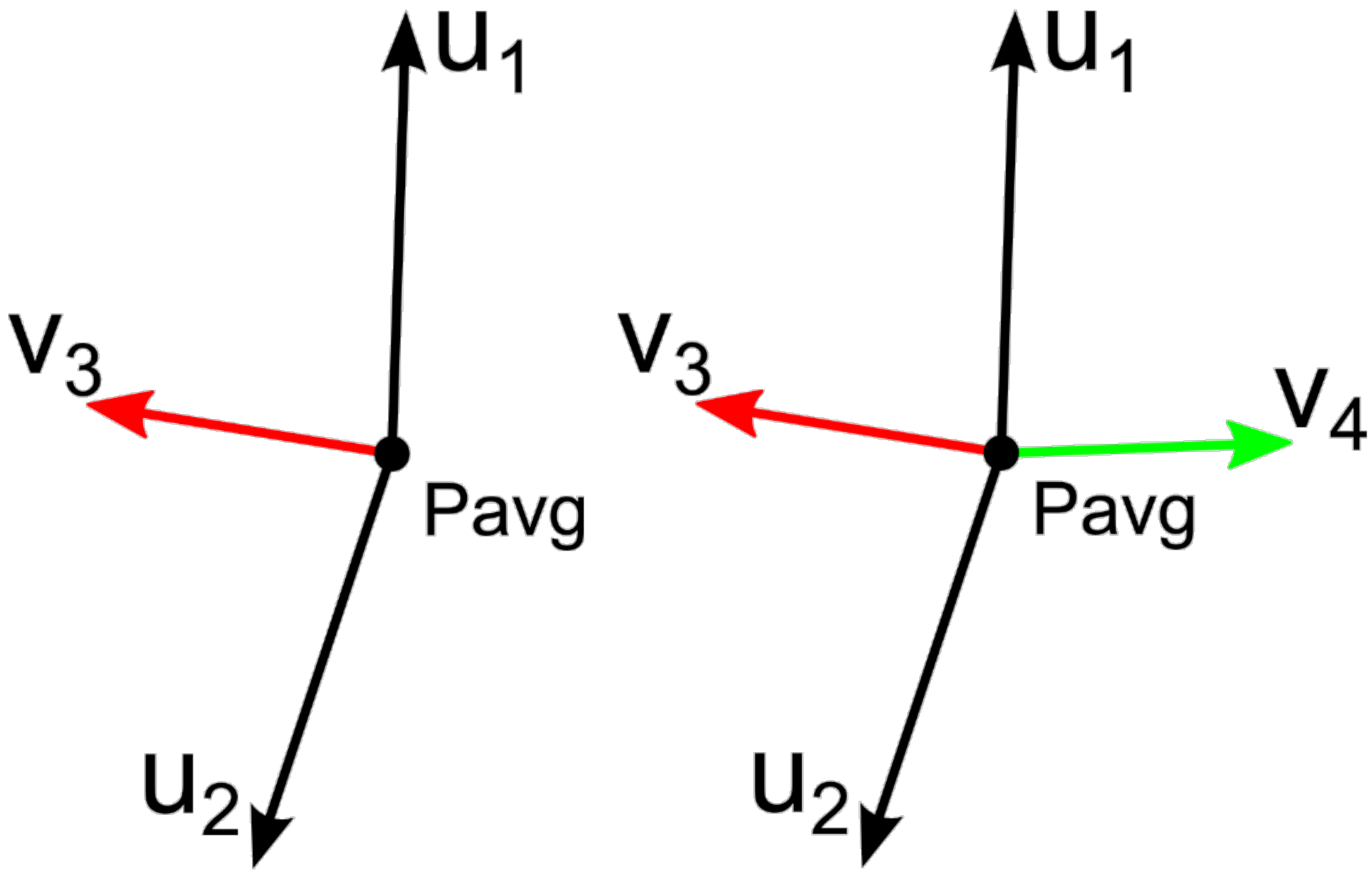
- Pick 3 points to represent flight vehicle in the nominal position (P_1,P_2, and P_3)
- Average P_1,P_2, and P_3 point coordinates to find P_{avg}
- Create unit vectors u_1 and u_2 from P_{avg} to P_1 and P_2





Marshall Data Synthesis

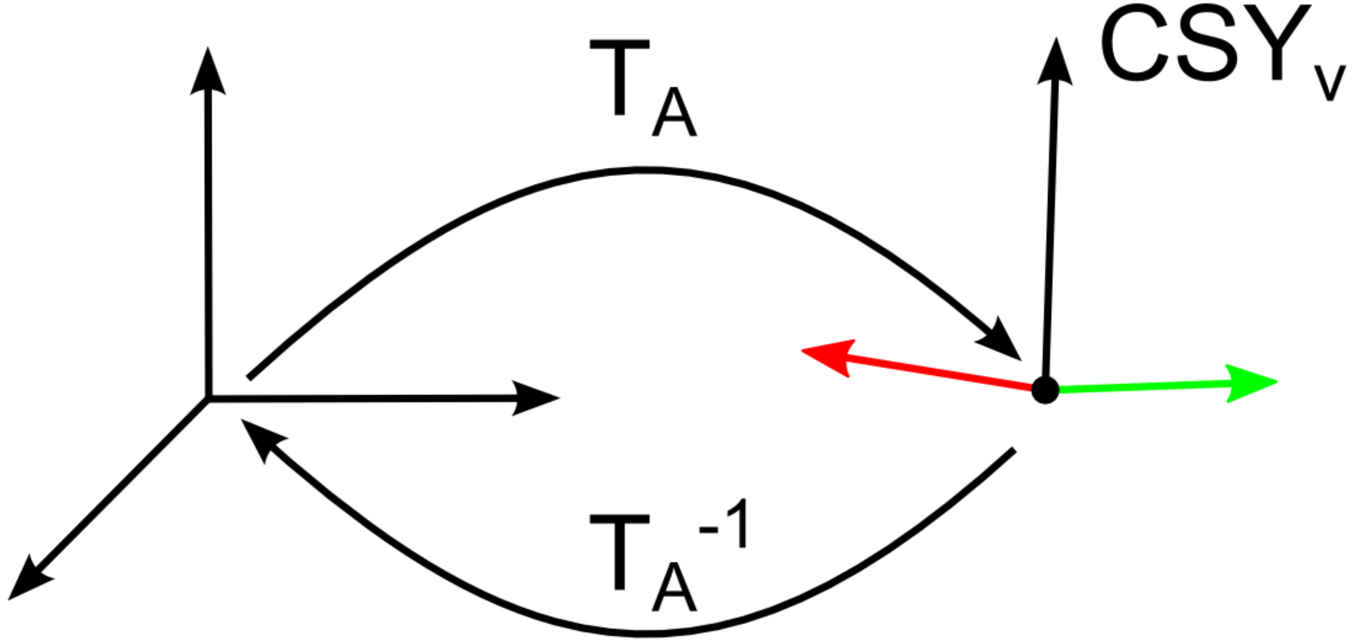
- Normalize the cross product of u_1 and u_2 to determine v_3
- Take the cross product of u_1 and v_3 to determine v_4
- Assemble u_1, v_3, and v_4 to form T_A



Marshall Data Synthesis

- ${}_{I}^{A}T$ is the matrix which will transform a coordinate system in the default coordinate frame orientation to CSY_v

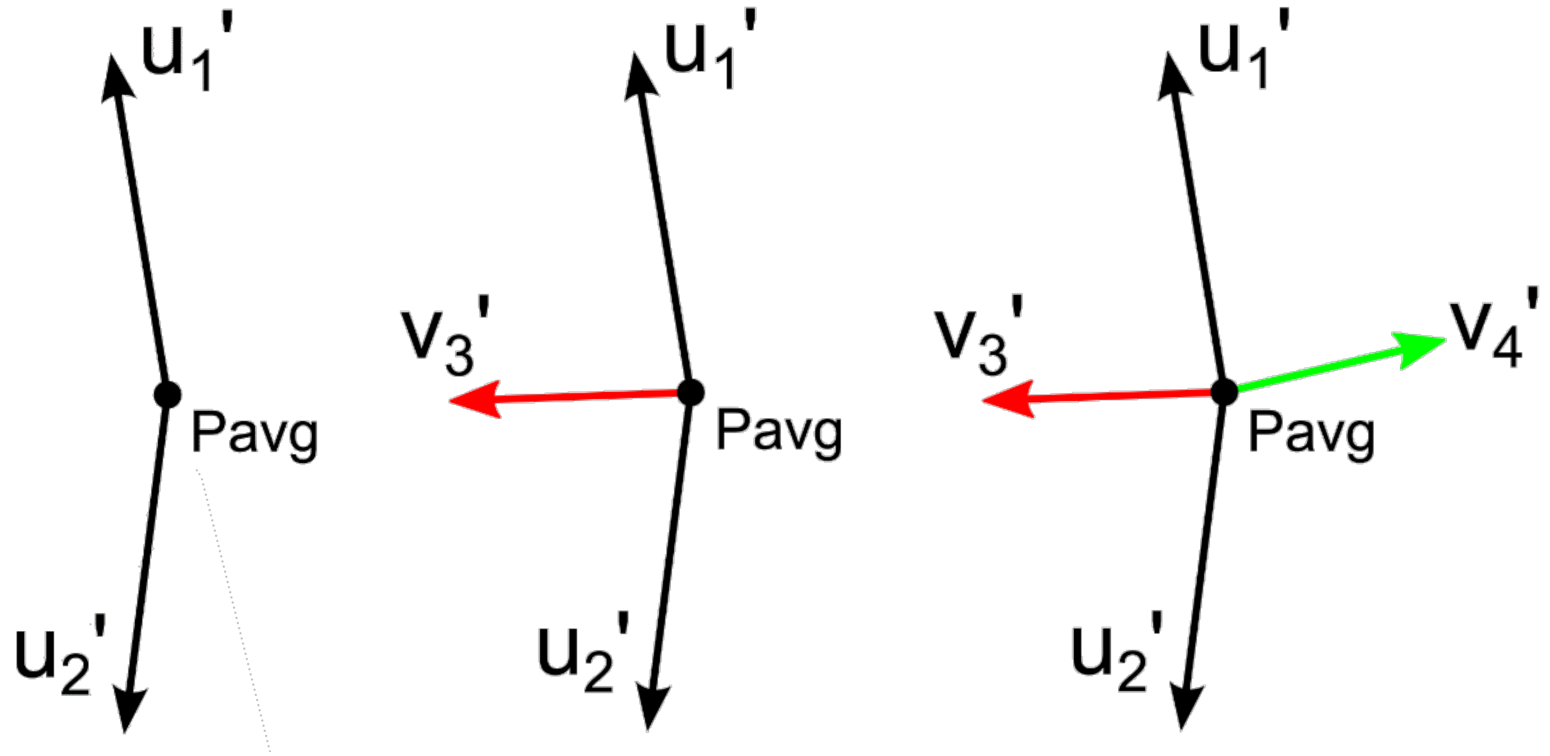
$$
{}_{I}^{A}T = \begin{bmatrix} u_1 \\ v_4 \\ v_3 \end{bmatrix} = \begin{bmatrix} u_{x1} & u_{y1} & u_{z1} \\ v_{x4} & v_{y4} & v_{z4} \\ v_{x3} & v_{y3} & v_{z3} \end{bmatrix}
$$




```
vx4=(ux2*uy1^2 - ux1*uy2*uy1 + ux2*uz1^2 - ux1*uz2*uz1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
vy4=(uy2*ux1^2 - ux2*uy1*ux1 + uy2*uz1^2 - uy1*uz2*uz1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
vz4=(uz2*ux1^2 - ux2*uz1*ux1 + uz2*uy1^2 - uy2*uz1*uy1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
vx3=(uy1*uz2 - uy2*uz1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
vy3=-(ux1*uz2 - ux2*uz1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
vz3=(ux1*uy2 - ux2*uy1)/(abs(ux1*uy2 - ux2*uy1)^2 + abs(ux1*uz2 - ux2*uz1)^2 + abs(uy1*uz2 - uy2*uz1)^2)^(1/2)
```

Marshall Data Synthesis

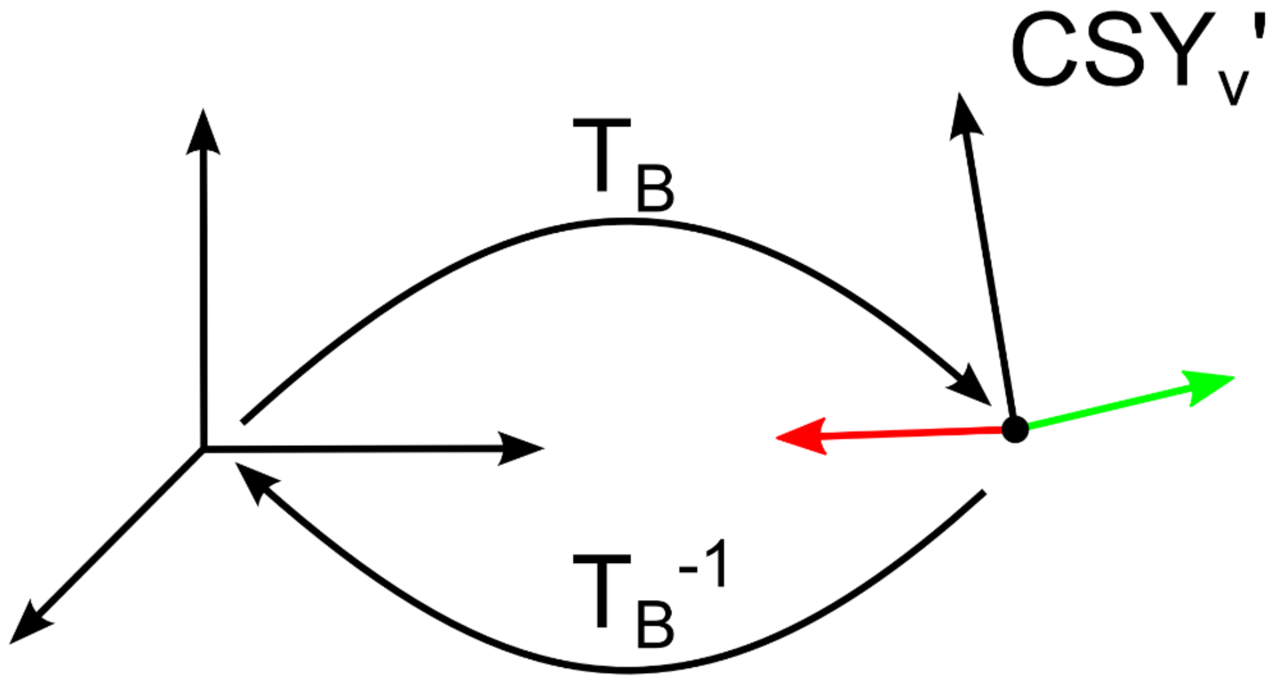
- Repeat process for the 3 corresponding flight vehicle points in the deflected position P_1',P_2', and P_3' to create P_{avg}', u_1', u_2', v_3', and v_4'



Marshall Data Synthesis

- ${}^{B}_{I}T$ is the matrix which will transform a coordinate system in the default coordinate frame orientation to the deflected vehicle frame $CSY_{v'}$

$$ {}^{B}_{I}T = \begin{bmatrix} u_1' \\ v_4' \\ v_3' \end{bmatrix} = \begin{bmatrix} u_{x1}' & u_{y1}' & u_{z1}' \\ v_{x4}' & v_{y4}' & v_{z4}' \\ v_{x3}' & v_{y3}' & v_{z3}' \end{bmatrix} $$



vx4p=(ux2p*uy1p^2 - ux1p*uy2p*uy1p + ux2p*uz1p^2 - ux1p*uz2p*uz1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)
vy4p=(uy2p*ux1p^2 - ux2p*uy1p*ux1p + uy2p*uz1p^2 - uy1p*uz2p*uz1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)
vz4p=(uz2p*ux1p^2 - ux2p*uz1p*ux1p + uz2p*uy1p^2 - uy2p*uz1p*uy1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)
vx3p=(uy1p*uz2p - uy2p*uz1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)
vy3p=-(ux1p*uz2p - ux2p*uz1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)
vz3p=(ux1p*uy2p - ux2p*uy1p)/(abs(ux1p*uy2p - ux2p*uy1p)^2 + abs(ux1p*uz2p - ux2p*uz1p)^2 + abs(uy1p*uz2p - uy2p*uz1p)^2)^(1/2)

Marshall Data Synthesis

- To determine the transformation matrix that will orient the vehicle from the nominal position to the deflected position we use the equation:

$$ {}_{A}^{B}T = {}_{I}^{B}T \, {}_{A}^{I}T = {}_{I}^{B}T \, {}_{I}^{A}T^{T} $$

$$ {}_{A}^{B}T = \begin{bmatrix} r_{x1} & r_{y1} & r_{z1} \\ r_{x2} & r_{y2} & r_{z2} \\ r_{x3} & r_{y3} & r_{z3} \end{bmatrix} = \begin{bmatrix} {u_{x1}}' & {u_{y1}}' & {u_{z1}}' \\ {v_{x4}}' & {v_{y4}}' & {v_{z4}}' \\ {v_{x3}}' & {v_{y3}}' & {v_{z3}}' \end{bmatrix} \begin{bmatrix} u_{x1} & u_{x4} & u_{x3} \\ v_{y1} & v_{y4} & v_{y3} \\ v_{z1} & v_{z4} & v_{z3} \end{bmatrix} $$

Marshall Data Synthesis

- Equate T to Rxyz
- Solve for θ_y, θ_x, and θ_z

$$R_{xyz} = [R_z][R_y][R_x] = \begin{bmatrix} cz & sz & 0 \\ -sz & cz & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cy & 0 & -sy \\ 0 & 1 & 0 \\ sy & 0 & cy \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cx & sx \\ 0 & -sx & cx \end{bmatrix}$$

$$R_{xyz} = \begin{bmatrix} cy * cz & cx * sz + cz * sx * sy & sx * sz - cx * cz * sy \\ -cy * sz & cx * cz - sx * sy * sz & cz * sx + cx * sy * sz \\ sy & -cy * sx & cx * cy \end{bmatrix} = \begin{bmatrix} r_{x1} & r_{y1} & r_{z1} \\ r_{x2} & r_{y2} & r_{z2} \\ r_{x3} & r_{y3} & r_{z3} \end{bmatrix} = {}^{B}_{A}T$$

$$\theta_y = \pm \text{atan2}\left(\mathrm{r}_{x3}, \sqrt{1 - \mathrm{r}_{x3}^2}\right) \qquad \theta_x = \text{atan2}\left(\frac{\mathrm{r}_{y3}}{-cy}, \frac{\mathrm{r}_{z3}}{cy},\right) \qquad \theta_z = \text{atan2}\left(\frac{\mathrm{r}_{x2}}{-cy}, \frac{\mathrm{r}_{x1}}{cy},\right)$$

$$Note: sx, sy, sz = \sin(\theta_x), \sin(\theta_y), \sin(\theta_z) \text{and } cx, cy, cz = \cos(\theta_x), cos(\theta_y), cos(\theta_z)$$