4.8.3. ElastoDyn Theory

Note this document is work in progress and is greatly incomplete. This documentation was started to document some code changes to the the tail furl and rotor furl part of ElastoDyn. Please refer to the different ressources provided in Section 4.8 for additional documents.

4.8.3.1. Notations

Points

The following (partial) list of points are defined by ElastoDyn:

- `Z` : the platform reference point
- `O` : the tower-top/base plate point
- `W` : the specified point on the tail-furl axis
- `I` : the tail boom center of mass
- `J` : the tail fin center of mass

Bodies

The following (partial) list of bodies are defined by ElastoDyn:

- `E` : the earth/inertial frame
- `X` : the platform body
- `N` : the nacelle body
- `A` : the tail-furl body

4.8.3.2. Kinematics

ElastoDyn computes the position, velocity and accelerations of key points of the structure, starting from the platform reference point *Z* and going up in the structure.

The different position vectors are available in the data stucture `RtHSdat`. For instance, the global position of point J is given by:

$$\boldsymbol{r}_\mathrm{J} = \boldsymbol{r}_\mathrm{Z} + \boldsymbol{r}_\mathrm{ZO} + \boldsymbol{r}_\mathrm{OW} + \boldsymbol{r}_\mathrm{WJ} \tag{4.184}$$

The translational displacement vector (how much a point has moved compared to its reference position) is calculated as follows: $\boldsymbol{r}_\mathrm{J} - \boldsymbol{r}_{\mathrm{J,ref}}$.

The coordinate systems of ElastoDyn are stored in the variable `CoordSys`. The orientation matrix of a given coordinate system can be formed using the unit vectors (assumed to be column vectors) of a given coordinate system expressed in the inertial frame. For instance for the tailfin coordinate system:

$$\boldsymbol{R}_\mathrm{Ai} = \begin{bmatrix} \hat{\boldsymbol{x}}^t_\mathrm{tf}\big|_\mathrm{i} \\ \hat{\boldsymbol{y}}^t_\mathrm{tf}\big|_\mathrm{i} \\ \hat{\boldsymbol{z}}^t_\mathrm{tf}\big|_\mathrm{i} \end{bmatrix} \tag{4.185}$$

Angular velocities are stored in variables `RtHSdat%AngVelE*` with respect to the initial frame ("Earth", *E*). For instance, the angular velocity of the tail-furl body (body *A*) is:

$$\boldsymbol{\omega}_\mathrm{A/E} = \boldsymbol{\omega}_\mathrm{X/E} + \boldsymbol{\omega}_\mathrm{N/X} + \boldsymbol{\omega}_\mathrm{A/N} \tag{4.186}$$

where $\boldsymbol{\omega}_\mathrm{N/X} = \boldsymbol{\omega}_\mathrm{B/X} + \boldsymbol{\omega}_\mathrm{N/B}$

Linear (translational) velocities of the different points are found in the variables `RtHSdat%LinVelE*`, and are computed based on Kane's partial velocities (which are Jacobians of the velocity with respect to the time derivatives of the degrees of freedom). For instance, the linear velocity of point J is computed as:

$$\boldsymbol{v}_\mathrm{J} = \sum_\mathrm{j} \frac{\partial \mathrm{v}_\mathrm{J}}{\partial \dot{\mathrm{q}}_\mathrm{j}} \dot{\mathrm{q}}_\mathrm{j} \tag{4.187}$$

where the Jacobians $\frac{\partial \mathrm{v}_\mathrm{J}}{\partial \dot{\mathrm{q}}_\mathrm{j}}$ are stored in `RtHSdat%PLinVelEJ(:,0)`

Translational accelerations are computed as the sum of contribution from the first and second time derivatives of the degrees of freedom. For instance, the acceleration of point *J* is computed as:

$$\begin{aligned} \tilde{\boldsymbol{a}}_\mathrm{J} &= \sum_{\mathrm{j}\in\mathrm{PA}} \frac{\partial \mathrm{a}_\mathrm{J}}{\partial \dot{\mathrm{q}}_\mathrm{j}} \dot{\mathrm{q}}_\mathrm{j} \\ \boldsymbol{a}_\mathrm{J} &= \tilde{\boldsymbol{a}}_\mathrm{J} + \sum_{\mathrm{j}\in\mathrm{PA}} \frac{\partial \mathrm{v}_\mathrm{J}}{\partial \dot{\mathrm{q}}_\mathrm{j}} \ddot{\mathrm{q}}_\mathrm{j} \end{aligned} \tag{4.188}$$

where $\frac{\partial \mathrm{a}_\mathrm{J}}{\partial \dot{\mathrm{q}}_\mathrm{j}}$ are stored in `RtHSdat%PLinVelEJ(:,1)`

Angular accelerations requires similar computations currently not documented.

4.8.3.3. Rotor and tail furl

The user can select linear spring and damper models, together with up- and down-stop springs, and up- and down-stop dampers.

The torque applied from the linear spring and damper is:

$$Q_{\mathrm{lin}} = -k\theta - d\dot{\theta} \tag{4.189}$$

where θ is the degree of freedom (rotor or tail furl), k is the linear spring constant (`RFrlSpr` or `TFrlSpr`) d is the linear damping constant (`RFrlDmp` or `TFrlDmp`).

The up-/down- stop spring torque is defined as:

$$Q_{\mathrm{stop,\,spr}} = \begin{cases} -k_{US}(\theta - \theta_{k_{US}}), & \text{if } \theta > \theta_{k_{US}} \\ -k_{DS}(\theta - \theta_{k_{DS}}), & \text{if } \theta < \theta_{k_{DS}} \\ 0, & \text{otherwise} \end{cases} \tag{4.190}$$

where k_{US} is the up-stop spring constant (`RFrlUSSpr` or `TFrlUSSpr`), $\theta_{k_{US}}$ is the up-stop spring angle (`RFrlUSSP` or `TFrlUSSP`), and similar notations are used for the down-stop spring.

The up-/down- stop damping torque is defined as:

$$Q_{\mathrm{stop,\,dmp}} = \begin{cases} -d_{US}\dot{\theta}, & \text{if } \theta > \theta_{d_{US}} \\ -d_{DS}\dot{\theta}, & \text{if } \theta < \theta_{d_{DS}} \\ 0, & \text{otherwise} \end{cases} \tag{4.191}$$

where similar nnotations are used. The total moment on the given degree of freedom is:

$$Q = Q_{\mathrm{lin}} + Q_{\mathrm{stop,spr}} + Q_{\mathrm{stop,dmp}} \tag{4.192}$$

4.8.3.4. Yaw-friction

A yaw-friction model is implemented in ElastoDyn based on a Coulomb-viscous approach. The yaw-friction moment as a function of yaw rate (ω) is shown below in Fig. 4.48

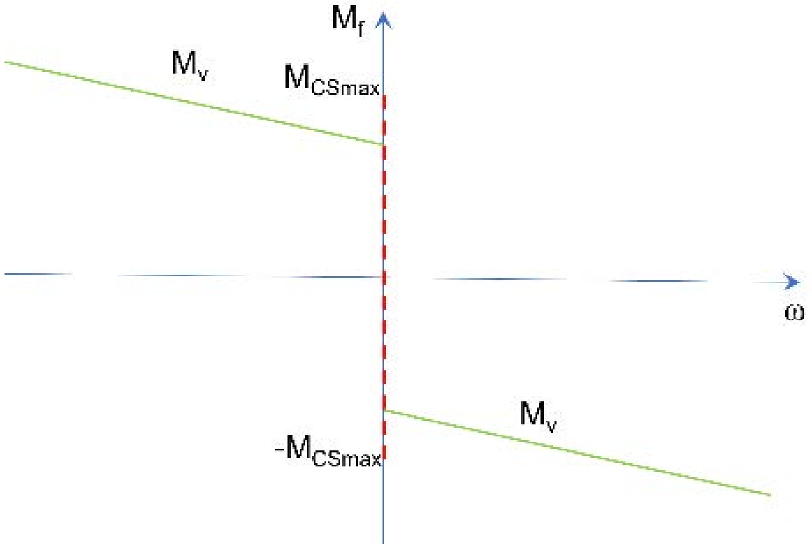

Fig. 4.48 *Yaw-friction model*

The yaw-friction torque M_f can be calcualated as follows.

if $\omega \neq 0$:

$$M_f = \mu_d \bar{D} \times \min(0, F_z) \times \mathrm{sign}(\omega) - \sigma_v \times \omega$$

if $\omega = 0$ and $\dot{\omega} \neq 0$:

$$M_f = \mu_d \bar{D} \times \min(0, F_z) \times \mathrm{sign}(\dot{\omega})$$

if $\omega = 0$ and $\dot{\omega} = 0$:

$$M_f = -\min(\mu_s \bar{D} \times |\min(0, F_z)|, |M_z|) \times \mathrm{sign}(M_z)$$

where $\bar{D}$ is the effective yaw-bearing race diameter, μ_d is the dynamic friction coefficient, μ_s is the static friction coefficient,:math:*F_z* is effective axial load on yaw-bearing, $M - z$ is the external torque on yaw-bearing. The static ‘stiction’ (where the static contribution exceeds the dynamic Coulomb friction) is only applied if both the yaw rotational velocity and acceleration at the current time-step are zero. The static portion of the friction is omitted if the rotational acceleration is not null, (sign(0) is taken as 1). This is to account for the fact that a ‘warm’ joint may not feel stiction when crossing through zero velocity in a dynamic sense [ed-HWS23]

4.8.4. Developer notes

Internal coordinate systems

The different coordinate systems of ElastoDyn are stored in the variable `CoordSys`. The coordinate systems used internally by ElastoDyn are using a different convention than the OpenFAST input/output coordinate system.

For instance, for the coordinate system of the nacelle, with unit axes noted $\mathrm{x_n}, \mathrm{y_n}, \mathrm{z_n}$ in OpenFAST, and $\mathrm{d_1}, \mathrm{d_2}, \mathrm{d_3}$ in ElastoDyn, the following conversions apply: $\mathrm{d_1} = \mathrm{x_n}, \mathrm{d_2} = \mathrm{z_n}$ and $\mathrm{d_3} = -\mathrm{y_n}$.

The following (partial) list of coordinate systems are defined internally by ElastoDyn:

- *z* : inertial coordinate system
- *a* : tower base coordinate system
- *t* : tower-node coordinate system (one per node)
- *d* : nacelle coordinate system
- *c* : shaft-tilted coordinate system
- *rf* : rotor furl coordinate system
- *tf* : tail furl coordinate system
- *g* : hub coordinate system