

Method cheat sheet for class VortexSystem

What it does	Function	Jump to
Calculates the coordinates of the bound vortices of one blade and saves them to <code>coordinates_blade_bound_elementwise</code> . It does that by purely rotating the given blade structure to the current " <code>blade_rotation</code> ".	<code>_blade_bound_elementwise</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L348
Calculates the coordinates of the trailing vortices of one blade and saves them to <code>coordinates_blade_trailing_elementwise</code> . It does that by calculating the first trailing vortex which is fixed to the blade. This vortex is parallel to the chord and goes from 1/4 chord to 5/4 chord. After that its helix geometry.	<code>_blade_trailing_elementwise</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L361
Combines elementwise coordinates into one numpy array. If the coordinates do not exist run the input function " <code>if_not_do</code> " (which should then calculate these coordinates). After that, continue with the combining.	<code>_combine_elementwise</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L434
Rotates combined coordinates. If the coordinates do not exist run the input function " <code>if_not_do</code> " (which should then calculate these coordinates). After that, continue with the rotating.	<code>_rotate_combined</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L455
Calculate the induced velocities at " <code>control_point</code> " of a vortex with <code>circulation=1</code> that spans from " <code>vortex_star</code> " to " <code>vortex_end</code> ".	<code>_vortex_induction_factor</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L485
Build up a matrix that relates the induced velocities at a control point with all existing bound vortices. The nested loops are (1. outermost, 3. innermost): <ol style="list-style-type: none"> 1. Bound vortex systems of all blades 2. Control points 3. Bound vortices of the current bound vortex system 	<code>bound_induction_matrices</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L185
Build up a matrix that relates the induced velocities at a control point with all existing trailing vortices. The nested loops are (1. outermost, 4. innermost): <ol style="list-style-type: none"> 1. Trailing vortex system of all blades 2. Control points 3. Blade elements 4. Trailing vortices of the current blade element of the current trailing vortex system 	<code>trailing_induction_matrices</code>	https://github.com/JonasMPN/LiftingLine/blob/master/src/vortex_system.py#L217
All other functions are either for error reporting, setting values of the class, plotting the vortex system, or changing the data format of method inputs.		

