# ServoDyn

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

Alternatively using logarithmic maps, for inputs that are small rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively. Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that , , and  are written as , , and , respectively, or equivalently are written as , , and , respectively.

The default perturbation sizes are as follows:

*  for rotational inputs in rad, rad/s, and rad/s^2

# ElastoDyn

For outputs that are rotations in 3D, i.e.,  and , it is implied that  and  are written as  and , respectively. Note:  for outputs that are rotations in 3D because rotational outputs in ElastoDyn don’t depend directly on the inputs.

Alternatively using logarithmic maps, for outputs that are small rotations in 3D, i.e.  and , it is implied that  is written as , or equivalently, . Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that  is written as . Note that for the platform rotatoinal states, which are made up of small angles in 3D, , and this simplifies to . So, for the platform rotation DOFs, .

The default perturbation sizes are as follows:

*  for rotational states (platform, yaw, rotor-furl, tail-furl, generator, drivetrain, teeter) in rad and rad/s

# BeamDyn

For states that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively, where the conversion between W-M parameters and DCM and its functional inverse is: .

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

For outputs that are rotations in 3D, i.e.  and , it is implied that  and  are written as  and , respectively.

The default perturbation sizes are as follows:

*  for rotational states in rad and rad/s
*  for rotational inputs in rad, rad/s, and rad/s^2

# AeroDyn

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

Alternatively using logarithmic maps, for inputs that are small rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively. Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that , , and  are written as , , and , respectively, or equivalently are written as , , and , respectively.

The default perturbation sizes are as follows:

*  for orientations, rotational velocities, and airfoil control inputs in rad, rad/s, and user-defined units

# HydroDyn

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

Alternatively using logarithmic maps, for inputs that are small rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively. Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that , , and  are written as , , and , respectively, or equivalently are written as , , and , respectively.

The default perturbation sizes are as follows:

*  for rotational inputs in rad, rad/s, and rad/s^2

# SubDyn

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

Alternatively using logarithmic maps, for inputs that are small rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively. Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that , , and  are written as , , and , respectively, or equivalently are written as , , and , respectively.

The default perturbation sizes are as follows:

*  for rotational inputs in rad, rad/s, and rad/s^2

# MoorDyn

For states that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively, where the conversion between rotation parameters and DCM and its functional inverse is: .

For inputs that are rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively.

Alternatively using logarithmic maps, for inputs that are small rotations in 3D, i.e.  and , it is implied that , , and  are written as , , and , respectively. Of course for small rotations, where  is always be made up of small angles, , and thus it is implied that , , and  are written as , , and , respectively, or equivalently are written as , , and , respectively.

The default perturbation sizes are as follows:

*  for rotational states in rad and rad/s
*  for rotational inputs in rad, rad/s, and rad/s^2

Since the perturbations to form the linearization are one at a time, can we use the standard Euler angles from global to perturb the system?

