

SE3

SE3 Representation of 3D rigid-body motion

This subclass of RTBPose is an object that represents rigid-body motion in 2D. Internally this is a 3x3 homogeneous transformation matrix (4x4) belonging to the group SE(3).

Constructor methods::

SE3 general constructor
SE3.angvec rotation about vector
SE3.eul rotation defined by Euler angles
SE3.exp exponentiate an se(3) matrix
SE3.oa rotation defined by o- and a-vectors
SE3.Rx rotation about x-axis
SE3.Ry rotation about y-axis
SE3.Rz rotation about z-axis
SE3.rand random transformation
SE3.rpy rotation defined by roll-pitch-yaw angles
new new **SE3** object

Display and print methods::

animate ^graphically animate coordinate frame for pose
display ^print the pose in human readable matrix form
plot ^graphically display coordinate frame for pose
print ^print the pose in single line format

Group operations::

***** ^mtimes: multiplication (group operator, transform point)
.* ^^times: multiplication (group operator) followed by normalization
/ ^mrdivide: multiply by inverse
./ ^^rdivide: multiply by inverse followed by normalization
^ ^mpower: xponentiate (integer only)
.^ ^power: exponentiate followed by normalization
inv inverse
prod ^product of elements

Methods::

det determinant of matrix component
eig eigenvalues of matrix component
log logarithm of rotation matrix $r \geq 0$ && $r \leq 1$
simplify ^apply symbolic simplication to all elements
Ad adjoint matrix (6x6)
increment update pose based on incremental motion
interp interpolate poses
velxform compute velocity transformation
interp interpolate between poses
ctraj Cartesian motion
norm normalize the rotation submatrix

Information and test methods::

dim* returns 4
isSE* returns true
issym* test if rotation matrix has symbolic elements
isidentity test for null motion
SE3.isa check if matrix is SE(3)

Conversion methods::

char convert to human readable matrix as a string
SE3.convert convert **SE3** object or SE(3) matrix to **SE3** object
double convert to SE(3) matrix
R convert rotation part to SO(3) matrix
S03 convert rotation part to S03 object
T convert to SE(3) matrix
t translation column vector
toangvec convert to rotation about vector form

todelta	convert to differential motion vector
toeul	convert to Euler angles
torpy	convert to roll-pitch-yaw angles
tv	translation column vector for vector of SE3
UnitQuaternion	convert to UnitQuaternion object

Compatibility methods::

homtrans	apply to vector
isrot	^returns false
ishomog	^returns true
t2r	^convert to rotation matrix
tr2rt	^convert to rotation matrix and translation vector
tr2eul	^^convert to Euler angles
tr2rpy	^^convert to roll-pitch-yaw angles
tranimate	^animate coordinate frame
transl	translation as a row vector
trnorm	^^normalize the rotation matrix
trplot	^plot coordinate frame
trprint	^print single line representation

Other operators::

+	^plus: elementwise addition, result is a matrix
-	^minus: elementwise subtraction, result is a matrix
==	^eq: test equality
~=	^ne: test inequality

- ^ inherited from RTBPose
- ^^ inherited from SO3

Properties::

n	get.n: normal (x) vector
o	get.o: orientation (y) vector
a	get.a: approach (z) vector
t	get.t: translation vector

For single **SE3** objects only, for a vector of **SE3** objects use the equivalent methods

t	translation as a 3x1 vector (read/write)
R	rotation as a 3x3 matrix (read)

Notes::

- The properties R, t are implemented as MATLAB dependent properties. When applied to a vector of **SE3** object the result is a comma-separated list which can be converted to a matrix by enclosing it in square brackets, eg [T.t] or more conveniently using the method T.transl

See also

[SO3](#), [SE2](#), [RTBPose](#).

Other functions named SE3

[SE2/SE3](#) [SO3/SE3](#)

Class Details

Superclasses [SO3](#)
Sealed false
Construct on load false

Constructor Summary

[SE3](#) Create an SE(3) object

Property Summary

[a](#)

[n](#)

[o](#)

ref	string, name of reference coordinate frame Help for SE3/ref is inherited from superclass RTBPOSE
t	SO3.T Get homogeneous transformation matrix Help for SE3/t is inherited from superclass SO3
target	string, name of target coordinate frame Help for SE3/target is inherited from superclass RTBPOSE

Method Summary

	Ad	SE3.Ad Adjoint matrix
	R	SO3.R Get rotation matrix Help for SE3/R is inherited from superclass SO3
Static	Rx	SE3.Rx Construct SE3 from rotation about X axis
Static	Ry	SE3.Ry Construct SE3 from rotation about Y axis
Static	Rz	SE3.Rz Construct SE3 from rotation about Z axis
	SO3	SE3.SO3 Convert rotational component to SO3 object
	T	SO2.T Get homogeneous transformation matrix
	Twist	SE3.Twist Convert to Twist object
	UnitQuaternion	SO3.UnitQuaternion Convert to UnitQuaternion object Help for SE3/UnitQuaternion is inherited from superclass SO3
Static	angvec	SE3.angvec Construct SE3 from angle and axis vector
	animate	RTBPose.animate Animate a coordinate frame Help for SE3/animate is inherited from superclass RTBPOSE
	char	RTBPose.char Convert to string Help for SE3/char is inherited from superclass RTBPOSE
Static	convert	SE3.check Convert to SE3
	ctraj	SE3.ctrj Cartesian trajectory between two poses
	ctranspose	
Static	delta	SE3.delta Construct SE3 object from differential motion vector
	det	SO3.inv Determinant Help for SE3/det is inherited from superclass SO3
	dim	RTBPose.dim Dimension Help for SE3/dim is inherited from superclass RTBPOSE
	disp	
	display	RTBPose.display Display pose in matrix form Help for SE3/display is inherited from superclass RTBPOSE
	double	RTBPose.double Convert to matrix Help for SE3/double is inherited from superclass RTBPOSE
	eig	SO3.eig Eigenvalues and eigenvectors Help for SE3/eig is inherited from superclass SO3
	eg	
Static	eul	SE3.eul Construct SE3 from Euler angles
Static	exp	SE3.exp Construct SE3 from Lie algebra
	homtrans	SE3.homtrans Apply transformation to points (compatibility)
	increment	SE3.increment Apply incremental motion to an SE3 pose
	interp	SE3.interp Interpolate SE3 poses
	inv	SE3.inv Inverse of SE3 object
	isSE	RTBPose.isSE Test if rigid-body motion Help for SE3/isSE is inherited from superclass RTBPOSE
Static	isa	SE3.ISA Test if matrix is SE(3)
	isequal	
	ishomog	Test if SE3 class (compatibility) Help for SE3/ishomog is inherited from superclass RTBPOSE
	ishomog2	Test if SE2 class (compatibility) Help for SE3/ishomog2 is inherited from superclass RTBPOSE
	isidentity	SE3.identity Test if identity element
	isrot	Test if SO3 class (compatibility) Help for SE3/isrot is inherited from superclass RTBPOSE
	isrot2	Test if SO2 class (compatibility) Help for SE3/isrot2 is inherited from superclass RTBPOSE
	issym	RTBPose.issym Test if pose is symbolic Help for SE3/issym is inherited from superclass RTBPOSE
	isvec	Test if vector (compatibility) Help for SE3/isvec is inherited from superclass RTBPOSE

	log	SE3.log Lie algebra
	logs	SE3.log Lie algebra in vector form
	minus	RTBPose.minus Subtract poses Help for SE3/minus is inherited from superclass RTBPOSE
	mpower	RTBPose.mpower Exponential of pose Help for SE3/mpower is inherited from superclass RTBPOSE
	mrdivide	RTBPose.mrdivide Compound SO2 object with inverse Help for SE3/mrdivide is inherited from superclass RTBPOSE
	mtimes	RTBPose.mtimes Compound pose objects Help for SE3/mtimes is inherited from superclass RTBPOSE
	ne	
	new	SE3.new Construct a new object of the same type
	norm	SE3.norm Normalize rotation submatrix (compatibility)
Static	oa	SE3.oa Construct SE3 from orientation and approach vectors
	plot	TRPLOT Draw a coordinate frame (compatibility) Help for SE3/plot is inherited from superclass RTBPOSE
	plus	RTBPose.plus Add poses Help for SE3/plus is inherited from superclass RTBPOSE
	power	RTBPose.power Exponential of pose Help for SE3/power is inherited from superclass RTBPOSE
	print	RTBPose.print Compact display of pose Help for SE3/print is inherited from superclass RTBPOSE
	prod	RTBPose.prod Compound array of poses Help for SE3/prod is inherited from superclass RTBPOSE
Static	rand	SE3.rand Construct random SE3
	rdivide	SO3.mrdivide Compose SO3 object with inverse and normalize Help for SE3/rdivide is inherited from superclass SO3
Static	rpy	SE3.rpy Construct SE3 from roll-pitch-yaw angles
	simplify	RTBPose.simplify Symbolic simplification Help for SE3/simplify is inherited from superclass RTBPOSE
	subs	RTBPose.subs Symbolic substitution Help for SE3/subs is inherited from superclass RTBPOSE
	t2r	Get rotation matrix (compatibility) Help for SE3/t2r is inherited from superclass RTBPOSE
	times	SO3.times Compose SO3 objects and normalize Help for SE3/times is inherited from superclass SO3
	toangvec	SE3.toangvec Convert to angle-vector form
	todelta	SE3.todelta Convert SE3 object to differential motion vector
	toeul	SE3.toeul Convert to Euler angles
	torpy	SE3.RPY Convert to roll-pitch-yaw angles
	tr2eul	SO3.tr2eul Convert to Euler angles (compatibility) Help for SE3/tr2eul is inherited from superclass SO3
	tr2rpy	SO3.tr2rpy Convert to RPY angles (compatibility) Help for SE3/tr2rpy is inherited from superclass SO3
	tr2rt	Split rotational and translational components (compatibility) Help for SE3/tr2rt is inherited from superclass RTBPOSE
	tranimate	Animate a 3D coordinate frame (compatibility) Help for SE3/tranimate is inherited from superclass RTBPOSE
	tranimate2	Animate a 2D coordinate frame (compatibility) Help for SE3/tranimate2 is inherited from superclass RTBPOSE
	transl	SE3.transl Get translation vector
	transpose	
	trnorm	SE3.trnorm Normalize rotation submatrix (compatibility)
	trplot	Draw a 3D coordinate frame (compatibility) Help for SE3/trplot is inherited from superclass RTBPOSE
	trplot2	Draw a 2D coordinate frame (compatibility) Help for SE3/trplot2 is inherited from superclass RTBPOSE
	trprint	Compact display of 3D rotation or transform (compatibility) Help for SE3/trprint is inherited from superclass RTBPOSE

trprint2	Compact display of 2D rotation or transform (compatibility) Help for SE3/trprint2 is inherited from superclass RTBPOSE
tv	SE.tv Return translation for a vector of SE3 objects
velxform	SE3.velxform Velocity transformation
vpa	RTBPose.vpa Variable precision arithmetic Help for SE3/vpa is inherited from superclass RTBPOSE