Virtual Robots Module add-in for Robotics Toolbox in MATLAB

(Developed at Indian Institute of Technology Delhi by Prof. S.K. Saha & team)

Virtual Robots Module is a visualization tool that allows the visualization and simulation of industrial robots, by making use of the computational capabilities of Robotics Toolbox in MATLAB. Using the available CAD models, commonly available industrial robots can be analyzed using the Robotics Toolbox and visualized using the Virtual Robots Module.

System Requirements:

- Windows OS (XP/7/8) (32/64bit)
- .NET 2.0

Using Virtual Robots Module in MATLAB

A **VRM_Robot()** class is implemented to integrate the computational capabilities of Robotics Toolbox and the visualization capabilities of Virtual Robots Module. The **SerialLink** object of the Robotics Toolbox is implemented as a property of the **VRM_Robot()** class, that can be accessed and used for analysis. The .NET COM object corresponding to the Virtual Robots Module is also implemented as a property of the **VRM_Robot()** class, whose functionalities can be accessed by the user. The documentation for the **VRM_Robot()** class is given below:

VRM_Robot

Virtual Robots Module robot object, which holds the information regarding the robot. The **SerialLink** object and the Virtual Robots Module are handled by the **VRM_Robot**() object.

Methods

VRM_Robot	create a robot object
AvailableRobots	displays the available robots in the installation directory
LoadRobot	load one of the available robot's properties and model
DisplayRobot	display the active robot
MoveRobot	move the robot from its initial position to the final position
ForwardKinematics	display forward kinematics between two joint configurations
CartesianMotionRelative	move the robot in Cartesian space relative to the current state

Properties

jointState	current values of the loaded robots's joint angles
initialJointState	initial position of joints for forward kinematics
finalJointState	final position of joints for forward kinematics

`rtbSerialLinkObject`	SerialLink object class from Robotics Toolbox
`vrmServer`	Virtual Robots Module object for displaying the robots.

VRM_Robot.VRM_Robot

Virtual robot object constructor

`robot = VRM_Robot()` creates a virtual robot object

VRM_Robot.AvailableRobots

Displays the available virtual robot models

`robot.AvailableRobots()` lists the available virtual robot models in the installation directory.

VRM_Robot.LoadRobot

Loads an available virtual robots

`robot.LoadRobot(‘robotName’)` loads the robot and sets up the `SerialLink` object in Robotics Toolbox corresponding to the related Denavit Hartenberg parameters.

`robotName` – String corresponding to name of one of the available robots

---

VRM_Robot.DisplayRobot

Displays the loaded robot in the Virtual Robots Module

`robot.DisplayRobot()` opens the Virtual Robots module and shows the currently loaded robot. The robot is shown in the configuration corresponding to the current value of robot.jointState.

`robot.DisplayRobot(jstate)` opens the Virtual Robots module and shows the currently loaded robot. The robot is shown in the configuration corresponding to the joint angles mentioned in jstate (1xn array, n= no.of joint angles). Jstate is given in radians

---

VRM_Robot.MoveRobot

Move the robot from initial to final joint configuration

`robot.MoveRobot()` moves the robot from the initial configuration specified in the property robot.initialJointState to the final configuration specified in the property robot.finalJoint State.

robot.MoveRobot(timesteps) moves the robot from the initial configuration specified in the property robot.initialJointState to the final configuration specified in the property robot.finalJoint State within the specified timesteps.

VRM_Robot.ForwardKinematics

Compute and display the forward kinematics simulation from one joint configuration to another.

robot.ForwardKinematics() displays the forward kinematics simulation of the robot while moving between the initial and final position of the robot given by robot.initialJointState and robot.finalJointState, respectively.

robot.ForwardKinematics(js_init, js_final, timesteps) displays the forward kinematics simulation of the robot while moving between the initial and final position of the robot given by js_init and js_final respectively, for the mentioned timesteps. js_init and js_final are 1 × n vectors (n = no.of joint angles).

VRM_Robot.CartesianMotionRelative

Move the robot in the Cartesian space relative to the current position.

robot.CartesianMotionRelative(changeInConfig) displays the motion of the robot in Cartesian space form the current position to the new position, as per the relative change in the end effector configuration.

changeInConfig is a 6 × 1 vector representing the relative change in Roll-Pitch-Yaw angles (in degrees) and relative change in position of the end effector(in meter) of the end effector.