
REVIEWED BY GORDON R. PENNOCK

As stated in the preface, the goal of this book is two-fold: (i) to explore the underlying principles of kinematic geometry which are so important for an understanding of rigid body displacements and velocities in a robotic manipulator; and (ii) to explore the principles of the geometry of force systems in as much as they relate to the understanding of the kinematics. The book emphasizes important and long-established principles which provide the reader with a basis for a deeper understanding of the capabilities and the limitations of robot motion. The authors believe that this knowledge can be used effectively to design and control robotic manipulators.

The key to the treatment on robotics presented...
Extension of Graph Theory to the Duality Between Static Systems and Mechanisms


**Related Proceedings Papers**

A Study of the Duality Between Epicyclic Gear Trains and Beam Systems

IDETC-CIE2006

The Duality Between Planar Kinematics and Statics

IDETC-CIE2005

Geometrical Kinematic Analysis of a Planar Serial Manipulator Using a Barycentric Formula

IDETC-CIE2015

**Related Chapters**

Statics and Stiffness of a Body

Mechanics of Accuracy in Engineering Design of Machines and Robots Volume II: Stiffness and Metrology

A New Structure of High Precision Extensometer

International Conference on Instrumentation, Measurement, Circuits and Systems (ICIMCS 2011)

Molecular Statics Study of the Effect of Hydrogen on Edge Dislocation Motion in Alpha-Fe

International Hydrogen Conference (IHC 2012): Hydrogen-Materials Interactions
All standard robots in DQ Robotics are defined inside the folder [root_folder]/robots and have the static method `kinematics()` that returns a DQ_Kinematics object. Therefore, a KUKA LWR 4 robot manipulator can be defined analogously: `lwr4 = KukaLwr4Robot.kinematics();`. All DQ_Kinematics subclasses have common functions for robot kinematics such as the ones used to calculate the forward kinematics, the Jacobian matrix that maps the configuration velocities to the time-derivative of the end-effector pose, as well as other Jacobian matrices that map the configuration velocities to the time derivative of o Group of Robotics, Automation and Biomechanics University of Bologna Italy. IFAC 2017 World Congress, Toulouse, France. Kinematics. Statics. Freedoms. 1. First-order Rigid-Body Kinematics 2. Statics of Rigid Bodies 3. Freedoms and Constraints 4. Screw Systems 5. Invariant and Persistent Screw Systems 6. Applications—Mobility Analysis 7. Application—Design 8. Application—Singularity Analysis 9. Conclusions. Kinematics. Statics. Robots and Screw Theory: Applications of Kinematics and Statics to Robotics [Davidson, Joseph K., Hunt, the late Kenneth H.] on Amazon.com. “This text by Davidson (mechanical and aerospace engineering, Arizona State U.) and the late Hunt (emeritus, mechanical engineering, Monash U., Australia) treats the capabilities and limitations of the mechanics of robot motion, particularly relying on the mathematical application of screw theory, which underlies areas of the mechanics of statics and first-order kinematics. After introducing screw theory, they explore applications related to spatial serial robot-arms, assembly-configurations of serial robot-arms, in-parallel actuation, and static stability in legged vehicles.”—SciTec It has been proven screw theory is an efficient approach for kinematic and dynamic analysis of parallel mechanism. In this paper detailed description of reciprocal screw is described. Reciprocal screws play an important role in robotics. Screw theory is an emerging approach in the field of robotics. It has been proven screw theory is an efficient approach for kinematic and dynamic analysis of parallel mechanism. It has been developed by $ = (; 0 ),..(3)$ Sir Robert Stawell Ball in 1876 for application in kinematics 3) Reciprocal Product and statics of mechanisms (rigid body mechanics). [1] A The reciprocal product of two screws, say $1 = (1 ; 10 )$, screw is a six-dimensional vector constructed from a pair of three-dimensional vectors.