Robotic Manipulation Using an Open-Architecture Industrial Arm: A Pedagogical Overview

By Robert J. Wood

Robotics education at the undergraduate level is most effective as a coupling between theoretical concepts and tangible experiments. Making this connection effective requires a pragmatic way of applying the traditional robotic material to exciting laboratory exercises. A course recently offered in Harvard's School of Engineering and Applied Sciences, simply titled "Introduction to Robotics," utilizes an open-architecture robotic arm to give students hands-on experience with topics that they encounter in lecture. None of the experiments conducted in this course are wholly novel; however, the use of an open-architecture hardware or software system enables the instructor to rapidly prototype lab exercises with minimal effort. This column will give an overview of the apparatus and experiments used for this course.

Apparatus Overview

The majority of existing industrial arms are not conducive to education: the user interface (software or teach pendant) is typically oriented to repetition of precise tasks. Although the physical instantiation of the arm is not a primary concern, the software interface to the arm is of quintessential importance. Students should not spend an inordinate amount of time learning a proprietary motion description language specific to any given manufacturer. Instead, we settled on the six degrees of freedom (DoF) open-architecture robot from Quanser. This system consists of a 5 DoF CRS CataLyst-5 from Thermo Electron Corporation mounted to a linear track (for the sixth axis). The existing CRS controller is supplemented with a Quanser control board, allowing the user to switch between the industrial controller and an open-architecture controller in which the user has access to everything from high-level commands to individual joint signals. The open-architecture configuration uses a Matlab or Simulink interface that includes libraries for common functions such as kinematics and control. At the base of the workspace, a peg board was installed, which enabled the lab instructors to interchangeably place objects and obstacles for the latter labs. Additionally, an overhead camera, with its primary axis anti-parallel with the inertial z-axis, is used for vision.

Lab Overview

Prior to each lab, students write Matlab functions to solve tasks as prelab exercises. Each successive lab builds on tools that students developed for the previous exercise while maintaining a close connection to the material presented in class.

Digital Object Identifier 10.1109/MRA.2008.928281

Lab 1: Forward Kinematics

Given the Denavit-Hartenberg convention and the geometry of the arm (taken from data sheets), the students first write Matlab functions for the homogeneous transformations and a script to calculate position and orientation of the tool frame. During lab time, the students input various joint angles into both the arm controller and their script. They are then required to physically measure the location of the tool frame and compare to their predictions while using observations of the arm to debug any discrepancies. It is important that the scripts consider joint limitations, and thus some of the joint angles given to the students are outside the physical limits, so as to test the robustness of their code. Furthermore, the students use this script to evaluate the extent of the workspace by varying the joint angles through the configuration space.

Lab 2: Inverse Kinematics

The prelab requires students to write a function to calculate all solutions to the inverse kinematics when given the position and orientation of the tool frame. Furthermore, their code must check that each solution in the configuration space does not violate joint limits and discard erroneous solutions. During lab time, the students are given various position and orientation values for the tool frame. Using their inverse kinematics function, they first evaluate how many, if any, solutions exist. They must then implement all valid solutions for joint configurations on the arm and physically measure the difference between actual and desired tool frame position and orientation. As with the first lab, they use this comparison to iteratively debug their function.

Lab 3: Velocity Kinematics and Singularities

The third lab involves an exploration of the relationship between velocities in the workspace and velocities in the configuration space. To do this, students first construct the manipulator Jacobian using the forward kinematics module from the first lab. From their numerical Jacobian matrices, they predict the singularities of the arm. In lab, the students run the arm close to its singular configurations by choosing trajectories in the workspace appropriately. Simultaneously, they observe the joint velocities and watch where a finite workspace velocity corresponds to large joint velocities (limited by motor torque and current saturation). The purpose of observing singular configurations becomes apparent in the next lab where students must incorporate singularities as obstacles in the configuration space.

Lab 4: Path Planning and Obstacle Avoidance

The students write algorithms to generate safe way points in the configuration space when given physical descriptions of

This report on robotic teaching in academia addresses the important issues related to undergraduate courses. In this case, robotics is a very good tool to ground the theoretical concepts that students may be facing for the first time, to their physical and practical significance. As the author points out, there is no need of complete novelty in these courses, rather a clear pedagogical structure that can take students from theory to practice and vice versa, in preparation for the more challenging courses to come.

—Paolo Fiorini, RAS Education Committee Cochair

obstacles and a goal in the workspace. Students are told about the geometry of the peg board and that it would contain cylindrical obstacles within the workspace boundary. For path planning and obstacle avoidance, they could use any viable method; however, lectures covered the gradient descent and probabilistic road map methods. This is the first competitive lab; students were separated in teams and competed against each other based on speed (i.e., minimum number of way points) and number of obstacles hit (including singularities). Teams were then told about the position and radius of obstacles in the workspace along with the start and goal positions. Once the teams generated safe way points, these were loaded into the arm controller, which interpolated the points and ran the trajectory while scoring each team.

Lab 5: Vision and Object Manipulation

The final lab entailed aspects of each of the preceding labs. In addition, students were given the position and orientation of an overhead video camera (with respect to the inertial frame). In class, vision algorithms were presented to segment an image and return the coordinates of objects (in the image plane). The students were told that the peg board would contain two objects: the smaller of the two is the object that is to be manipulated, and the larger object is the goal. Using the overhead camera, they were first required to determine the centroid of each object (by segmentation and a simple camera calibration) and their relative sizes. For this lab, the end effector from the previous lab was replaced with a simple bellows-actuated gripper. Students would generate way points in the configuration space that would bring the gripper over the smaller object while requiring an orientation of the gripper that will facilitate grasping. A close command is given to the gripper to pickup the object. The second set of way points should lift the object and move it to a safe position above the second object (goal) and release. In this lab, the goal object was a basket so that a successful trial is one that puts the smaller object in the basket.

Potential Additions

Once the open-architecture infrastructure is in place, multiple additional exercises could be readily developed into labs. For example, labs on individual joint control, visual servoing, force feedback for manipulation, and force versus position control would all be natural progressions of the previous labs.

Robert J. Wood is an assistant professor in Harvard School of Engineering and Applied Sciences. He completed his M.S. degree in 2001 and Ph.D. degree in 2004 in the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley. At Harvard, he founded the Harvard Microrobotics Lab and has demonstrated the world's first robotic insect capable of generating sufficient thrust to take off. His current research interests involve the creation of biologically inspired aerial and ambulatory microrobots, minimal control of underactuated, computationally limited, nonlinear dynamical systems, and decentralized control of multiagent systems. He is the winner of 2007 DARPA Young Faculty award, 2008 NSF CAREER award, 2008 ONR Young Investigator Program award, and many best paper and video awards.

Address for Correspondence: Robert J. Wood, School of Engineering and Applied Sciences, Harvard University, 33 Oxford St., Cambridge, MA 02138, USA. E-mail: rjwood@seas.harvard.edu.