Teaching mechanism dynamics using a haptic device - II

Majid Koul, Subir Saha, M Manivannan

Abstract

Haptics feedback has interested several researchers in using the technology for teaching dynamics associated with many physical systems. Mechanism dynamics is one such area in which the effects of changes in kinematic and dynamic parameters of a virtual mechanism can be realized physically with a haptic device.

In this work, an in-house developed inverse dynamics algorithm for closed-loop systems is utilized in order to compute the forces/torques associated with the complex mechanisms in motion. The proposed work is a generalization to the concept of 'teaching mechanism dynamics using haptics' demonstrated in our previous work [1]. In this part, the protocol to integrate virtual mechanisms, generalized dynamics algorithm, the control causality and the haptic device are discussed.

The development of such technologies is contemporary and is expected to greatly transform the pedagogy of teaching dynamics.

Keywords: Haptics, Dynamics, Education, Higher-DOF mechanisms

1 Introduction

Mechanisms form an integral part of any small or big machine that usually performs a primary or secondary function in the system. They are ubiquitously found in vehicles, aircraft's, robots, medical devices, as well as in production machineries. Mechanisms are inherently closed-loop in structure and usually have planar or spatial nature. Since mechanisms form a basic component of any complex machine, the understanding of the same is important for good design. In general, engineering students face difficulties in realizing the physics associated with these mechanisms. This has been largely due to the typical classroom based teaching which relies on visual display of such systems only on the blackboard or a video. Improving upon such methodologies of teaching, a laboratory with several physical models of mechanisms always comes handy for the students to understand the physics behind them. However, there is always a need to develop and conceive new ideas or mechanisms, which may be practically difficult to realize and time consuming. Haptics based teaching can fill-up this gap by providing physical understanding of any given mechanism. The effect of change in inertial and kinematic properties of the virtual mechanism can be conveyed in the form of

Majid Koul (Corresponding author)
School of Technology, Islamic University of Science and Technology, J&K, majidkoul@gmail.com.

Subir Saha
Department of Mechanical Engineering, Indian Institute of Technology Delhi, New Delhi, saha@mech.iitd.ac.in.

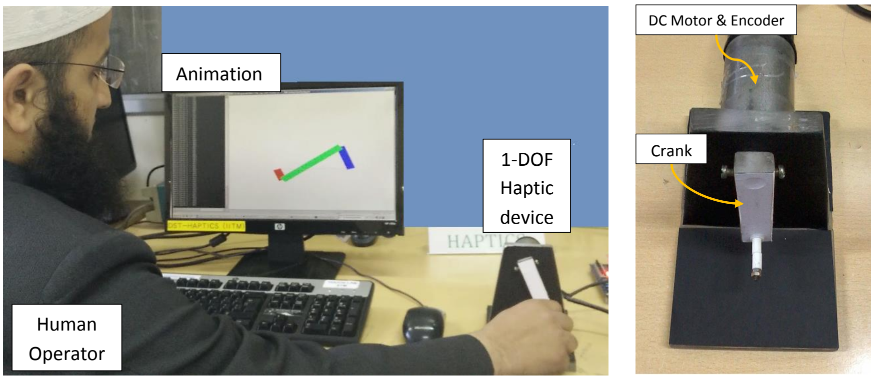
M Manivannan
Department of Applied Mechanics, Indian Institute of Technology Madras, Chennai, TN, mani@iitm.ac.in.

torque/force feedback at the haptic device. In fact, vision along with the haptics feedback is an essential and promising tool for the design process of such mechanisms [2].

In this work, a demonstration is made to show how a simple low cost 1- and 2-DOF haptic interfaces can be utilized to teach a typical course on mechanism dynamics. The methodology is extended to higher-DOF systems as well even though they may not physically exist. The rest of the paper is organised as follows: Section 2 describes the hardware and software development, while Section 3 provides illustration of the concept using several single and multi-DOF mechanisms. Section 4 details the summary of the work.

2 Hardware and Software development

This section details the necessary hardware and software required and their development for the mechanism dynamics simulation.



(a) Operator receiving both visual and haptic cues (b) 1-DOF haptic device

Figure 1: A typical teaching mechanism dynamics setup

2.1 Haptic interface

Haptic interface is popularly defined in the literature as a system consisting of the haptic device (mechanical part) and the associated electrical and electronics part (driver circuits, controller, etc.). Figure 1(a) depicts the typical setup for teaching mechanism dynamics using a haptic device. A human operator holds the haptic device while his eyes focus on the animation of the mechanism on the screen.

An acrylic-based low-cost 1-DOF haptic device used for this purpose is shown in Fig. 1(b), allowing a full 360° swing about its rotation axis. This is generally a desirable feature for driving several 1-DOF mechanisms completely. This setup had a low-cogging DC motor with collocated optical encoders for position sensing. A crank was attached to the motor axis directly without any gear in-between. Gears are usually avoided in haptic devices on account of backlash and high back-drive friction, while capstan drive arrangement does not allow full rotation about the axis.

Digital signals from optical encoder were sent to the controller that utilized quadrature decoding algorithm along with a counter module to calculate the current position of the crank. This positional information was sent to the PC via an enhanced parallel port protocol. On the other hand, the controller received necessary torque commands from the PC and converted it to an equivalent PWM signals along-with the suitable direction of the applied torque. Appropriate handshaking commands were written in the controller and on the PC side for bidirectional communication.

2.2 Generalized inverse dynamics algorithm

For determining the generalized forces/torques associated with the mechanisms in motion, a generalized inverse dynamics program was utilized. The program would directly/indirectly utilize the human operators input trajectory at the active joints of the mechanism, besides the corresponding velocity and acceleration. In this work, the generalized inverse dynamics algorithm for closed-loop systems proposed in [3] was used. The algorithm had been written in the commercial software MATLAB.

For the above algorithm, angular velocity of the active joint of the haptic device was estimated by differentiating the position information of corresponding samples in MATLAB. Similarly, angular acceleration was estimated from the corresponding samples of the angular velocity information. Appropriate signal filtering was carried out in MATLAB (using several inbuilt functions) before inputting the information to the inverse dynamics algorithm.

2.3 Mechanism animation

Next, a complex mechanism had to be made visible and working as if the real mechanism was being driven. Making such animations using Open Graphics Libraries (OpenGL) of VC++ software proved quite handy, given the amount of literature and variety available in using this software platform. Animations required positional information of the links as input, which were scaled or equated to the compatible dimensions on the screen. Active joint information was sent to the inverse kinematics function to calculate the passive joint trajectories of the mechanism.

2.4 Hardware and software integration

Next, in order to integrate the various components of the mechanism dynamics setup, the following procedure was followed: The Enhanced Parallel Port (EPP) communication protocol was written in Visual C++ for data in and out from the PC along with the animation of the mechanism in OpenGL (VC++). On the other hand, the inverse dynamics program was available in MATLAB.

In order to communicate between the MATLAB and VC++ programs, a readily available tool viz. *matlabengine* [4] was used initially. However, the communication speed was considerably slow ($\sim$ 20 Hz) which caused instability to the haptic device.

A TCP/IP (Transmission Control Protocol, Internet Protocol) communication protocol was alternatively used. For this protocol, socket programming was required that worked based on the definition of a client and a host. More details can be found in [5]. This protocol enabled a high speed inter process communication (IPC) between the MATLAB and VC++ programs used for basic dynamic calculations and animations, respectively ($\sim$ 500 Hz). This data transfer speed was enough for the envisaged experiments enabling stable operations with the haptic device.

In order to reduce the delay between the position and torque commands received (or sent) from (or to) the haptic device, the concept of thread programming was used in VC++. A processor has the capability to spawn threads of processes which are processed serially one after another. Each process is accessed in between, which is equivalent to processing these entities in parallel although they are serial in access. Microsoft Visual C++ provides such facility. A piece of code was written in VC++ which is similar to writing three processes. One was for the data acquisition and output through EPP protocol. The second one was for graphic output using OpenGL. The third one was for TCP/IP communication protocol using socket programming.

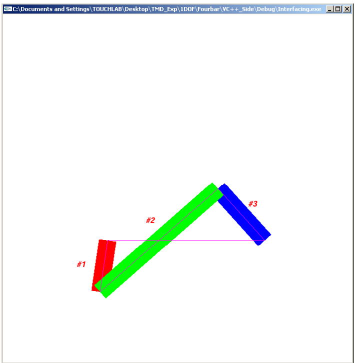
3 Illustrations

In all the experiments performed using the proposed haptic device, it was important that the user was made familiar with the setup and its working. This would avoid possible damage to the device during the experiments and ensure smooth operation. During the experimental trials, a user was asked to concentrate on the animation and avoid looking at the actual setup. The experiments were conducted with the user holding the handle of the crank with the dominant hand (right hand). Position profile from the encoder data was logged into a file along-with the required torque from the inverse dynamics algorithm.

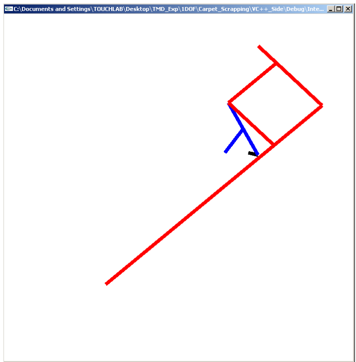
For comparison of the desired torque, the magnitude of electric current to the DC motor was measured using an in-built current sensing circuit in the LMD18200T H-bridge. Since the torque applied by a DC motor is proportional to its current, an appropriate comparison was done rather easily. A suitable electric resistance (based on LMD18200T specifications) was attached and the analog signal from the sensor was logged to the PC using National Instruments data acquisition card (USB-6001). The voltage levels were then measured and converted to the corresponding current values. Only absolute values of the currents were provided and measured thereby. Negative values were implemented using a direction bit in the H-bridge that would change the direction of rotation of the DC motor.

3.1 Mechanisms with one-DOF

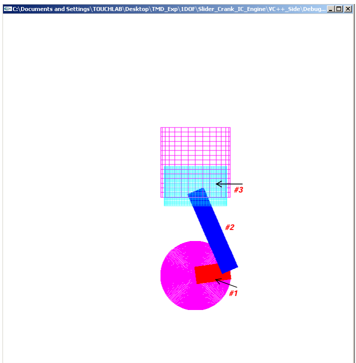
In this sub-section, experiments were carried out with several planar and spatial mechanisms having one-degree-of-freedom only. Figure 2(a) shows animation window of the crank-rocker four-bar mechanism while Fig. 2(b) shows a carpet scrapping mechanism. Animation window of a slider-crank mechanism used in IC engine is shown in Fig. 2(c), while Fig. 2(d) shows a spatial 1-DOF RSUR mechanism. These animated windows would pop up on the screen after running each corresponding programs.



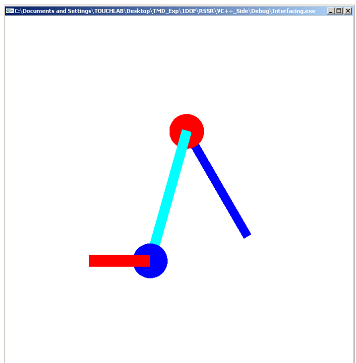
(a) Four-bar mechanism



(b) Carpet scrapping mechanism



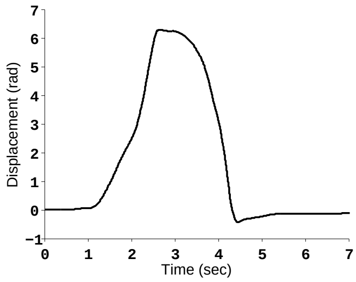
(c) Slider-crank - IC Engine



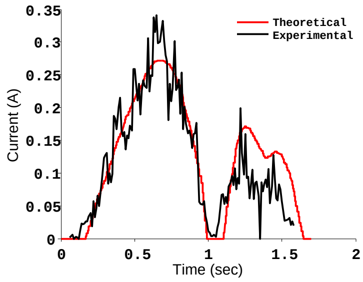
(d) RSUR mechanism (Spatial)

Figure 2: Animation windows of one-DOF mechanisms

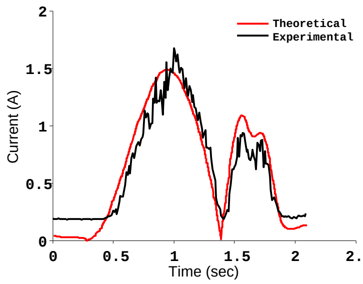
In the experiments for each mechanism, the crank of the haptic device was first given a full 360° rotation anticlockwise and later clockwise. For the four-bar mechanism, current requirements for the low and high masses of the coupler are evident from Figs. 3(b) and 3(c), respectively. Higher mass of the coupler would require more torque at the crank or current to the motor, respectively. Also, the difference in the nature of the curves obtained in Figs. 3(c) and 3(d) demonstrates the change in dynamics associated with a system in two directions. A comparative results of the current requirements at the motor and the actual measured values for the above trajectories for the other mechanisms could not be reported due to space constraint. Note that the velocity and acceleration components of the driving crank were of lesser magnitudes since high angular velocities were difficult to achieve with manual crank motion. An amplification factor for each mechanism was provided based on the required torque. This would however vary depending upon the limitations of the motor used.



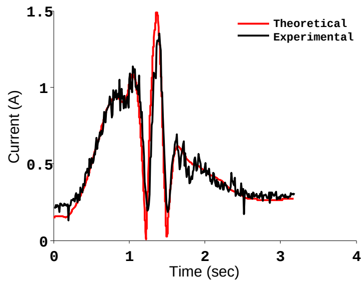
(a) Position profile



(b) Current profile for the low mass of the coupler



(c) Current profile for the high mass of the coupler



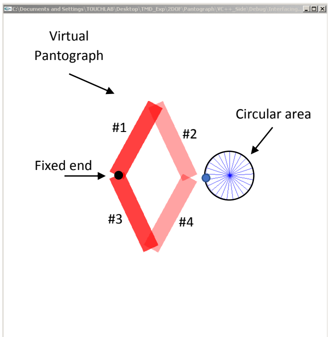
(d) Case (c) with reverse rotation

Figure 3: Experimental outcomes for the crank-rocker four-bar mechanism

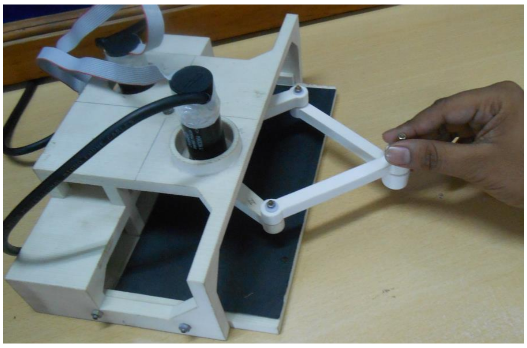
3.2 Multi-DOF mechanisms

A similar strategy was used for the simulation of 2-DOF Pantograph mechanism with congruent joints, for which a 2-DOF haptic device was used. The change in the kinematic and dynamic properties of the links was reflected at the joint torques towards the end-effector of the 2-DOF haptic device. Figure 4(a) shows the animation window of the Pantograph connected to the 2-DOF haptic device shown in Fig. 4(b). A task to move the end-effector manually along the circular periphery shown in Fig. 4(a) was given. Joint torques are plotted in Fig. 5 with a low mass (3 kg) and high mass (5 kg) of the passive links. The joint torques need to be matched with the enhanced amount ($\times$ 10) due to the torque enhancers in the actual device.

Figures 5(a) and 5(b) depict the computed end-effector profiles of the haptic interface or the animated Pantograph. The plot is a region between the two Cartesian axes where the manual task resulted into an approximate circle. Figures 5(c) and 5(d) depict the torque at the active joint 1 of the Pantograph, while the torques at the active joint 3 of the Pantograph is not shown owing to the space constraint. An increase in the torque requirement for the high masses of the passive links is observed from above and similarly felt by the operator. Corresponding current values were not shown here due to the noise in the measurements on account of the low value. Since torque enhancer

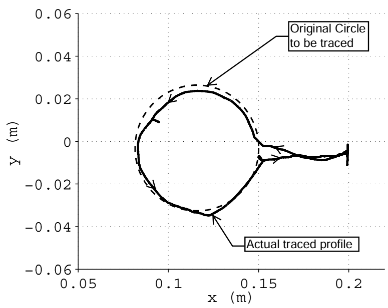


(a) Animation of the Pantograph

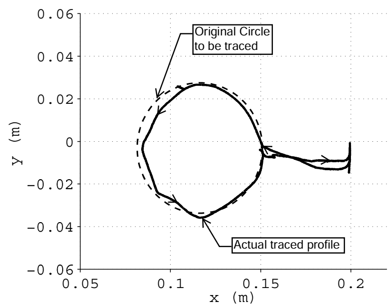


(b) The 2-DOF haptic device - Congruent joint Pantograph

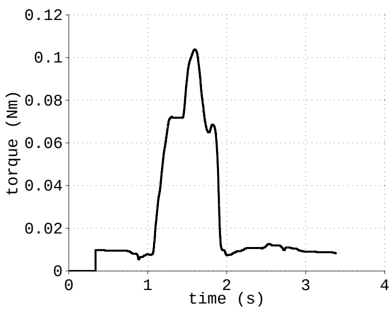
Figure 4: Two-DOF mechanism dynamics simulation



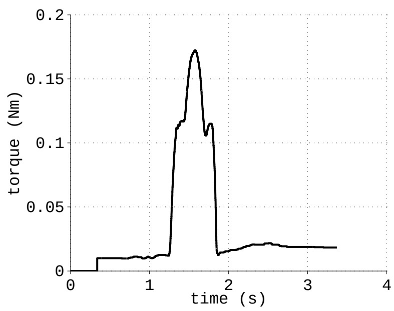
(a) End-effector profile with low masses



(b) End-effector profile with high masses



(c) Joint 1 (low masses)



(d) Joint 1 (high masses)

Figure 5: End-effector and active joint torque profiles for drawing circle with the 2-DOF haptic device

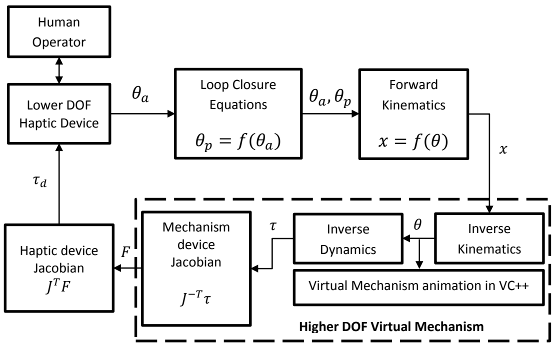


Figure 6: Control strategy for higher-DOF simulation

increased the torque, corresponding current was low compared to the one-DOF case, where no torque enhancer was used.

3.3 Driving parallel manipulators

As an interesting application of this work, the 2-DOF haptic device was utilized to partially drive a higher-DOF parallel manipulator. A typical strategy for such an application is shown in Fig. 6. The operator manipulated the haptic device whose end-effector was assumed to control the same-DOF as that of the higher-DOF manipulator. For example, the two translatory movements of the end-effector of the planar 3-RRR manipulator were easily mapped to the 2-DOF haptic device.

As seen in Fig. 6, the forward kinematics block computed the end-effector position of the 2-DOF haptic device. This information was mapped to the workspace of the higher-DOF manipulator, where an inverse kinematics module computed the joint information of the higher-DOF manipulator. These joint values were used for the

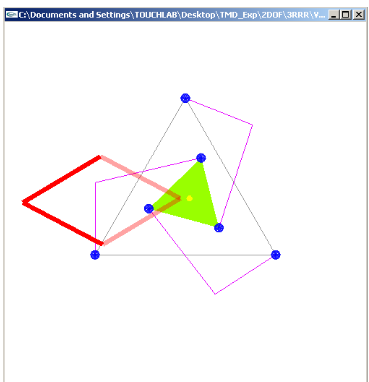


Figure 7: 3-RRR platform driven by Pantograph

necessary animation of the manipulator on the computer screen. The same were differentiated for estimating velocities and accelerations, which were utilized by the inverse dynamics module to calculate the required torque/forces at the joints. Using the Jacobian of the higher-DOF manipulator, the equivalent forces at the end-effector were computed, which were multiplied by Jacobian of the haptic device to generate torques at the actuators. Figure 7 depicts the animation window of the 3-RRR manipulator being manipulated by the 2-DOF haptic device.

4 Summary

In this work, a generalized strategy to teach mechanism dynamics using a haptic device was presented. The protocol for hardware and software integration was detailed. Several examples of mechanisms were provided in which a generalized inverse dynamics algorithm proposed elsewhere was used to compute the necessary forces/torques. A strategy to reflect partial dynamics of higher-DOF systems on the lower-DOF haptic device was also proposed.

A psycophysics study of the effect of such teaching pedagogy on the learning capabilities of a student is a part of our future work.

Acknowledgement

The first author would like the thank BRNS/ BARC Mumbai for the financial support during this work under the sponsored project Adaptive Force Control of Industrial robot using Force/Torque sensor.

References

[1] M. H. Koul, S. K. Saha, and M. Manivannan, "Teaching mechanism dynamics using a haptic device," in *Proceedings of the 1st International and 16th National Conference on Machines and Mechanisms (iNaCoMM2013), IIT Roorkee, India*, 2013.

[2] G. Ye, J. J. Corso, G. D. Hager, and A. M. Okamura, "Vishap: Augmented reality combining haptics and vision," in *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, vol. 4, pp. 3425–3431, IEEE, 2003.

[3] M. H. Koul, *Dynamics of closed-loop multi-body systems and their application to haptic interfaces*. PhD thesis, IIT Delhi, 2015.

[4] Y. Cheng, S. Li_ran, and L. Xue_yao, "The application of matlab engine to vc++ 6.0 and its object-oriented programming [j]," *Applied Science and Technology*, vol. 11, p. 013, 2001.

[5] K. R. Fall and W. R. Stevens, *TCP/IP illustrated, volume 1: The protocols*. addison-Wesley, 2011.