2006-942: INNOVATIVE METHODS IN TEACHING FUNDAMENTAL UNDERGRADUATE ENGINEERING COURSES

Amir Rezaei, West Virginia University Inst. of Tech.
Dr. Amir Rezaei is an Associate Professor of Mechanical Engineering at California State Polytechnic University,Pomona. His research interests include Anisotropic Elasticity, Composite Materials, Vibration, and Stability. He is an active member of American Society of Engineeirng Education (ASEE) and is currently serving in Design Division (DEED) of this society. He has taught across the mechanical engineering curriculum as well as developing new courses in graduate and undergrduate levels.

Marco Schoen, Idaho State University

Gurdeep Hura, West Viginia University Inst. of Tech.

Innovative methods in teaching fundamentals of undergraduate engineering courses

Amir G. Rezaei, Ph.D.
Marco P. Schoen Ph.D.
Gurdeep Hura, Ph.D.
Umesh Korde, Ph.D.

Mechanical Engineering Department
California State University, Pomona
Idaho State University
West Virginia University Institute of Technology
South Dakota School of Mines and Technology

Abstract

Some of the fundamental theories in undergraduate engineering tend to be highly theoretical and mathematically complex such as Vibration, Strength of Material, Controls, Electrical and Kinematics Theories. The traditional treatment in teaching these theories may not be beneficial for some undergraduate students who can be classified as visual learners or poorly prepared in early courses in science or mathematics. Engineering coursework should be multi-disciplinary, collaborative, active learning and accommodate students from a variety of backgrounds. Our engineering graduates should realize that a good portion of the engineering skills is dedicated to actually applying the fundamental theory and asking the right questions, in addition to trying to find a solution.
With the availability of fast computers and software available for every subject, it is now possible to teach students much more than what we could in the past using the traditional tools. Through the use of modern technology we could help students to develop intuition and judgment for problems, which could not been possible through traditional chalk and board techniques. These tools can make student who is "weak" analytically to understand the fundamental theories in engineering more fully and be able to apply them when they are practicing in the filed. It is the intent of this paper to gather and present some of the useful demonstrations of simulation of the fundamental theories in engineering created in new software tools and share the experience with other engineering educators to enhance teaching in the classroom.

I. Introduction

Basic animation and simulation can significantly improve the instructional process relative to the traditional blackboard approach[1]. Textbook homework problems and examples presented to the students on the board during the class period are helpful but even then, a student may have difficulty visualizing what a mathematical solution to a problem means. Using computational software equipped with simulation capability during the class period will enhance student understanding of the fundamental theory significantly and help them to explore more about the theory being discussed[2,3]. The approach presented in this paper is the use of computer animation as the vehicle for experimentation which is presented to students during a fifty minute lecture period. This paper is an integration of many recommendations on enhancing the undergraduate engineering education and improving teaching in the classroom by the authors in different institution across the United States.

II. Interactive Computational Tools

There are many general-purpose computational tools such as MATLAB™, SIMULINK™, ANSYS©, Etc already in use for computational purposes in higher education. Most of these software tools are becoming more user-friendly and interactive to use on the computer screen. It is the intent of this paper to present some of these computational tools capability and their effectiveness during the classroom to the audience in ASEE conference. These software include but not limited to:

1. Working Mode©, which is a general purpose Simulation/ Animation package.
2. MathCAD©, which is a comprehensive design environment that enables one to combine formulas, text and interactive graphics in a single worksheet.
3. ANSYS©, which is a Finite Element Analysis software.
4. Etc.

These simulation demonstrations will prepare the students to understand and better visualize the fundamental concept being taught in the classroom.

In the next sections a few concepts in engineering has been chosen to apply the effectiveness of simulation and animation to improve teaching effectiveness in the classroom. The classes that these tools are used are mostly upper level engineering courses were the theoretical concepts are more difficult to understand by the students. The class sizes where these methods are applied range from 15 to 23 students.

1. Robot Kinematics Visualization using Matlab™

The treatment of kinematics and inverse kinematics in Robotics and Mechatronics classes can become rather abstract. Using industrial or educational robotic manipulators to demonstrate for example the Denavit-Hartenberg parameters in class or laboratory environments is very restrictive, since each robot has only one set of these parameters, and these type of equipment tends to be rather expensive. Visualization of different parameter selection, the resulting

workspace computation, and the exploration of the effects of variable limitations are of great value to students in these types of classes. Traditionally, a robotics or mechatronics class will provide the fundamentals in kinematics and the inverse kinematics computation, and solutions are most often found through the help of software such as Matlab[©]. Programming is as essential to this field as the fundamentals in computing the resulting parameters of a manipulator configuration. Combining the need to practice programming, designing robotic manipulators, visualizing the resulting robots, and playing with the available parameters to arrive at some optimal configuration can lead to a much improved experience for the student. Matlab© has a rather user-friendly GUI (graphic user interface) toolbox. Students use this software package already to solve kinematics and inverse kinematics problem for these classes. Introducing GUI programming using Matlab[©] in these classes to design, analyze, and visualize their robotic concepts is very effective and cost efficient. In the following, an example of an undergraduate student's work is presented, which addresses the above mentioned ideas.

The GUI developed allows the user to select the number of links the robotic manipulator will posses. After that, a window for the definition of the variables is generated, as seen in Figure 1a. Once the joint variables are defined, the Denavit-Hartenberg parameters are selected in a new window. The querying for the link parameters is depicted in Figure 1b.
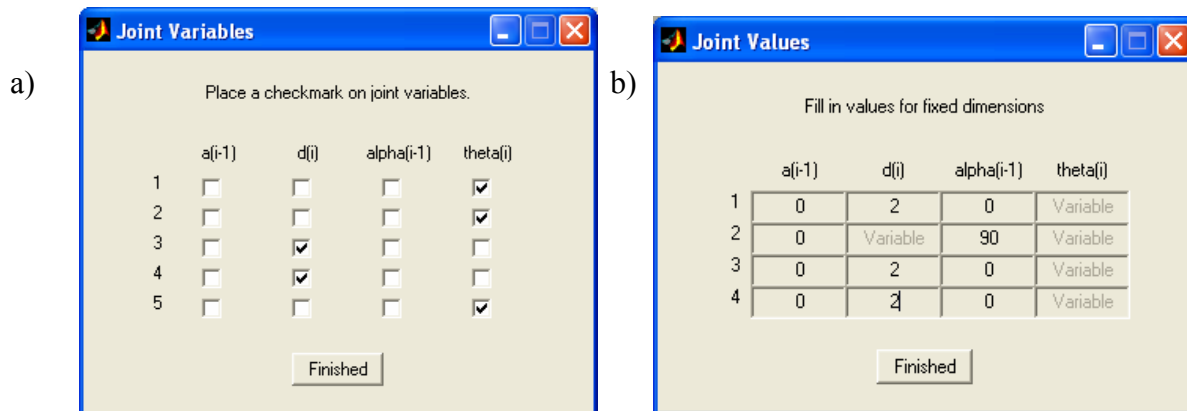
a)
b)

Figure 1: user interface for a) defining the joint variables, and b) selecting the link parameters a conceptual robotic manipulator.

The resulting robotic manipulator is generated using the traditional kinematic relationships, which is easily programmed in Matlab™. An example of such a visualization is given in Figure 2, where the different links are depicted in different colors.
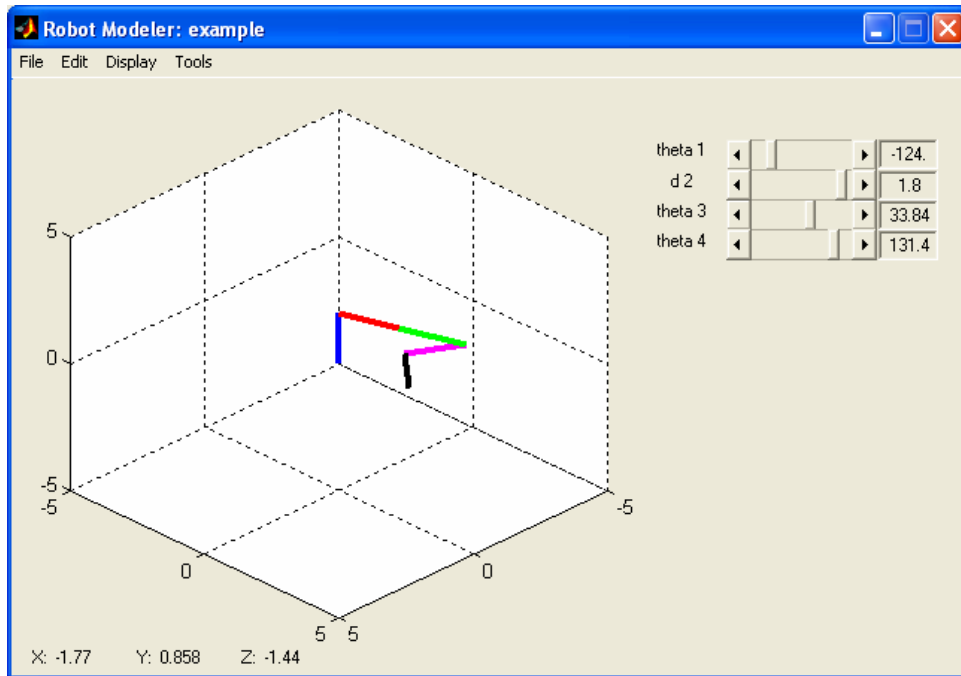
Figure 2: GUI interface with resulting robotic manipulator, with slides to simulated different configurations.

2. Simulation of Rotating Unbalance in Theory of Vibration

Rotating unbalance is a source of troublesome vibration in rotating machinery [6]. It is also very important to understand its fundamental theory for engineers since many machines and devices in real-life applications do experience this phenomenon. For application engineers it is most important to visualize the behavior of these machines when the unbalanced mass induces undesired vibration. Some of most common machines that experience rotating unbalance are Gyroscopes, Cryo-coolers, Tires, and Washing machines. A schematic of such a rotating unbalance of mass m_0, a distance e from the center of rotation is given in Figure 3. The simulated model in Working Model$^©$ is shown in Figure 2. Both models have exactly the same parameters and material constants. The frequency of rotation of unbalance is donated by ω and the natural frequency of the machine itself is calculated to be ω_n.
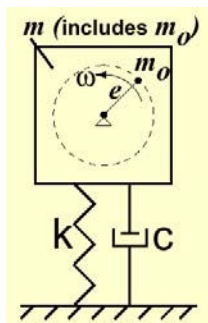

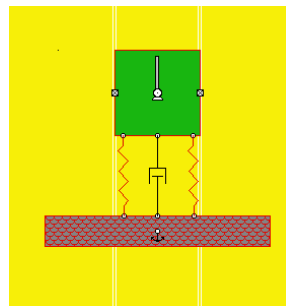**Figure 3.** Rotating Unbalance Simulation

Figure 4. Model created in Working Model 2D for

The forces imparted by the unbalance on the structure are shown on Figure 5 with two components R_x and R_y.
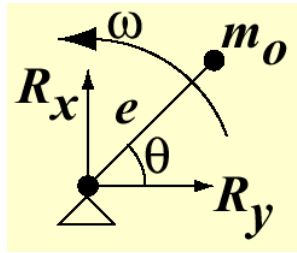

Figure 5. Forces imparted by the unbalance

The x component of the unbalance force is cancelled by the reactions from the frictionless wall and the y component will induce a force, which will make the problem just like any other single Degree of Freedom System with harmonic excitation with governing differential equation of motion given by

$$m \ddot{x} + c \dot{x} + kx = m_0 e \omega^2 \sin \omega t$$

The total solution to the above equation is given by:

$$x(t) = A \sin(\omega_d t + \phi) + \frac{(m_0 e / m) r^2}{\sqrt{(1-r^2)^2 + (2\zeta r)^2}} \sin(\omega t - \tan^{-1}(2\zeta r / (1-r))) \qquad (1)$$

Where ω_d is the damped natural frequency, r is the frequency ratio, ζ is the damping factor of the system and A and ϕ are determined from the initial conditions for the rotating unbalance.

To understand the behavior of the system, one needs to perform parametric studies by obtaining solutions to equation (1) for different values of system's parameters. One way is to create the physical model and obtain experimental data and produce graphs of the data to demonstrate the solution to students. Building a physical model will require hardware and most of all time in the laboratory. Using computational tools there is no need for a physical model to be built to perform parametric studies. Among those computational software for parametric studies is Mathcad©. MathCAD© will produce graphical results of equation (1) for different inputs of the parameters of the system and allow the user to obtain the solution for different values of the system's parameters. The results are produced graphically which can be used in the classroom for presentations. The results of inputs into MathCAD© have produced graphs shown in Figure 6 below. Using such software, students' understanding of problems such as rotating unbalance can be improved considerably. A survey by the author in engineering vibration course and a class size of 23 students showed improvement in scores students on topics which computational software was used during instructions.
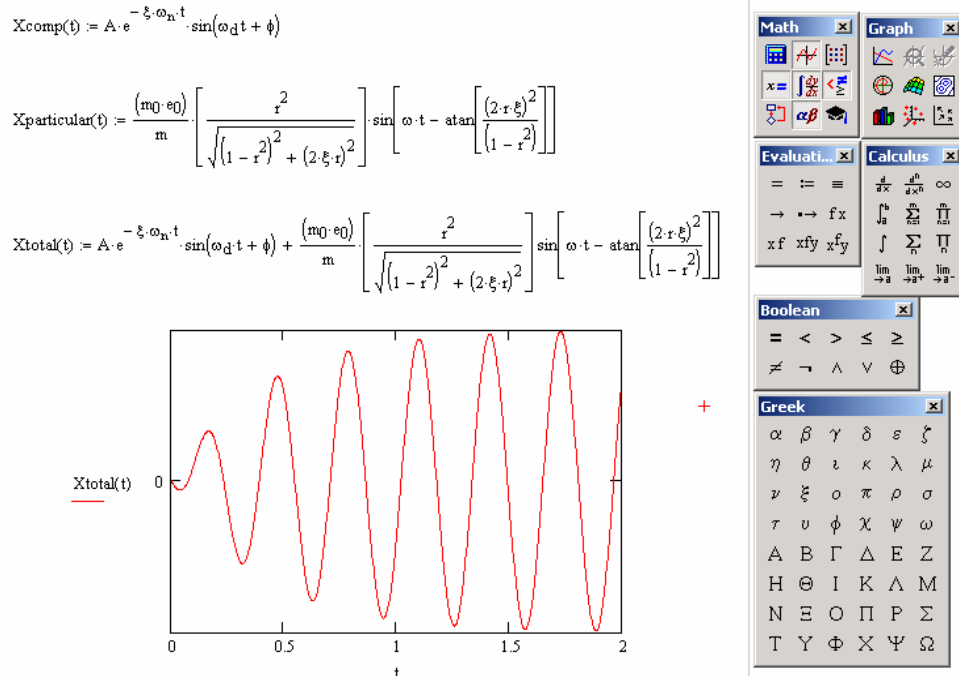
$$X_{comp}(t) := A \cdot e^{-\xi \cdot \omega_n \cdot t} \cdot \sin(\omega_d t + \phi)$$

$$X_{particular}(t) := \frac{(m_0 \cdot e_0)}{m} \cdot \left[\frac{r^2}{\sqrt{(1-r^2)^2 + (2 \cdot \xi \cdot r)^2}} \right] \cdot \sin\left[\omega \cdot t - \text{atan}\left[\frac{(2 \cdot r \cdot \xi)^2}{(1-r^2)} \right] \right]$$

$$X_{total}(t) := A \cdot e^{-\xi \cdot \omega_n \cdot t} \cdot \sin(\omega_d t + \phi) + \frac{(m_0 \cdot e_0)}{m} \cdot \left[\frac{r^2}{\sqrt{(1-r^2)^2 + (2 \cdot \xi \cdot r)^2}} \right] \cdot \sin\left[\omega \cdot t - \text{atan}\left[\frac{(2 \cdot r \cdot \xi)^2}{(1-r^2)} \right] \right]$$

Figure 6. MathCAD output for a simple parametric studies of Rotating Unbalance

The author's investigation also revealed that using computational software such as MATCAD©
will only help those students that already have good mathematical understanding of the problem
being discussed in class.   Combining mathematical computational tools with simulation tools in
class to introduce problems such as rotating unbalance will help to improve teaching
effectiveness more than just using the computational tools. One of these simulation software that
can be successfully used during class is Working Model©. Working Model© is an advanced
motion simulation package with sophisticated editing capabilities. The program makes it possible
to build and analyze dynamic mechanical systems with essentially no programming or
mathematical understanding about the dynamics of the problem. The software allows one to
design and test without building a physical model for the problem or system. The windows
interface is intuitive and easy to learn. The dynamics engine is physics-based and enables the
program to accurately calculate the behavior of even highly complex models. Figure 7 shows the
model created for rotating unbalance problem and the results of the motion due to the unbalance
mass. This simulation model allows the user to change the system's parameters on the screen
and the results are shown to match perfectly with those results obtained from the computation
software tools. The authors experience in using such motion simulation model during the class
helps students to be interested and attracted to understand the mathematics involved in motion of
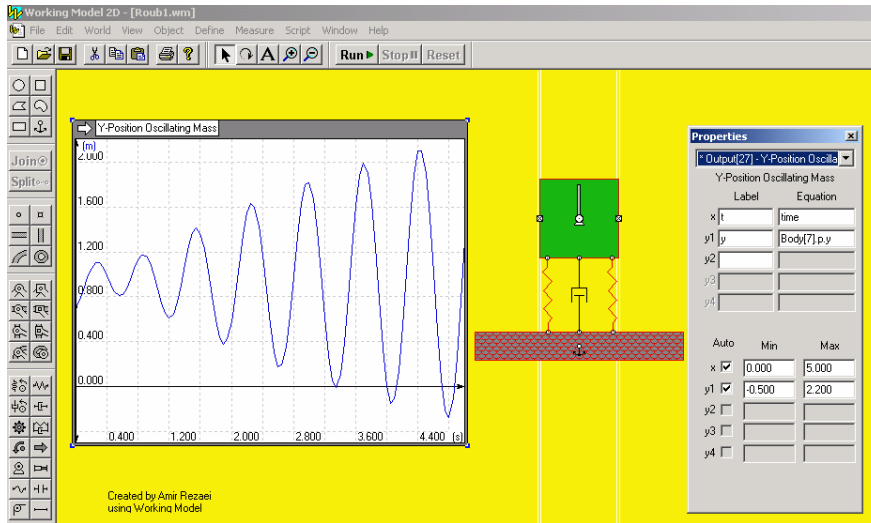such structures.

Figure 7. Simulation model created in Working Model© for Rotating Unbalance

3. Single and two degree of Freedom System Simulation

Vibration of a single degree freedom system, a spring-mass system, with damping in theory of vibration is a fundamental concept to teach to students in theory of vibration. It is also very important for engineers since many machines and devices in real-life applications do experience vibratory phenomenon. For application engineers it is most important to visualize the behavior of these machines with lack of enough damping produces undesired vibration. A schematic of such single degree freedom system with damping with mass m is given in Figure 8. The simulated model in Working Model is shown in Figure 9. Both models have exactly the same parameters and material constants.
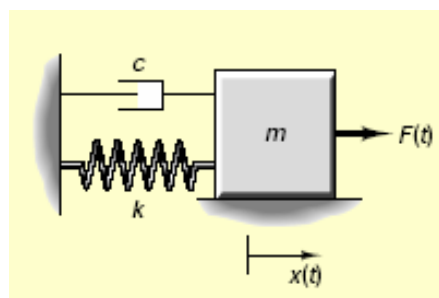

Figure 8. Spring mass system with damping
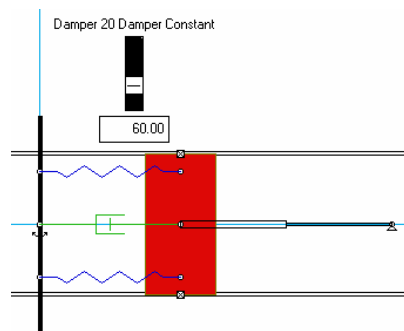

Figure 9. Model created in Working Model 2D© for Simulation

These models represent a single degree freedom system with harmonic excitation and its governing differential equation of motion is given by

$$m\ddot{x} + c\dot{x} + kx = F_0 \sin \omega t$$

The total solution to the above equation is also given by equation (1).

To understand the behavior of this system, one needs to perform parametric studies by obtaining solutions to equation (1) for different values of system's parameters. One way is to create the physical model and obtain experimental data and produce graphs of the data to demonstrate the solution to students. The physical model of this system is available at the control laboratory which will be demonstrated to the students later.   MathCAD$^{©}$ will produce graphical results of equation (1) for different inputs of the parameters of the system and allow the user to obtain the solution for different values of the system's parameters. These parameters will be important in understanding the control action of a controller like the damper. The results are produced graphically which can be used in the classroom for presentations. The results of inputs into MathCAD$^{©}$ have produced graphs shown in Figure 10. Using such software, students' understanding of problems such as the spring mass system can be improved considerably. A survey obtained in engineering vibration course and a class size of 23 students showed improvement in scores students on topics which computational software was used during instructions.
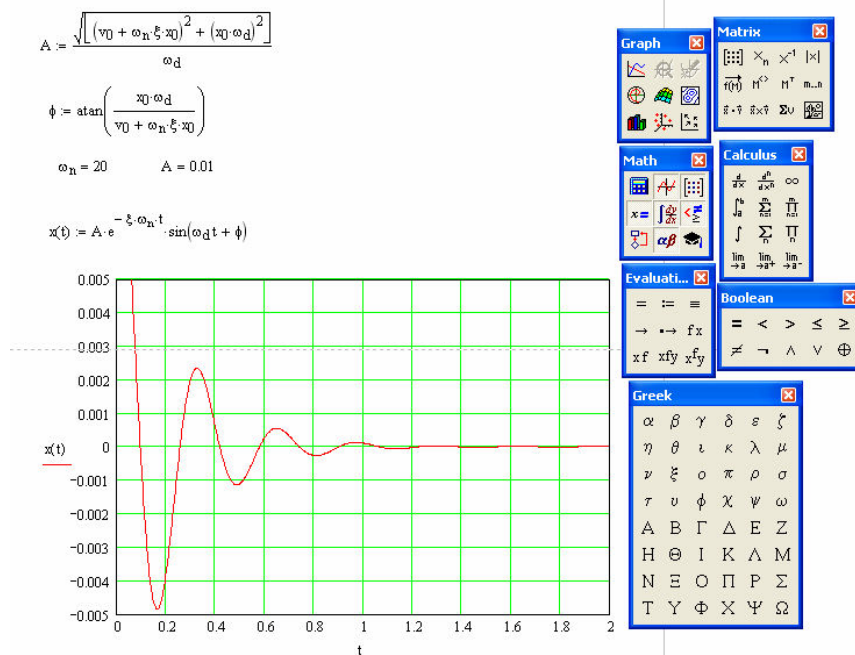

$$A := \frac{\sqrt{\left(v_0 + \omega_n \cdot \xi \cdot x_0\right)^2 + \left(x_0 \cdot \omega_d\right)^2}}{\omega_d}$$

$$\phi := \mathrm{atan}\left(\frac{x_0 \cdot \omega_d}{v_0 + \omega_n \cdot \xi \cdot x_0}\right)$$

$$\omega_n = 20 \qquad A = 0.01$$

$$x(t) := A \cdot e^{-\xi \cdot \omega_n \cdot t} \cdot \sin\left(\omega_d t + \phi\right)$$

Figure 10. MathCAD$^{©}$ output for a simple parametric studies of spring-mass with damping

The investigation also revealed that using computational software such as MathCAD$^{©}$ will only help those students that already have a good mathematical understanding of the problem being discussed in class. Combining mathematical computational tools with simulation tools in class to introduce problems such as the spring mass system will help to improve teaching effectiveness more than just using the computational tools. One of these simulation software that can be successfully used during class is Working Model. Working Model is an advanced motion simulation package with sophisticated editing capabilities. The program makes it possible to build and analyze dynamic mechanical systems with essentially no programming or mathematical understanding about the dynamics of the problem. The software allows one to design and test without building a physical model for the problem or system. The windows interface is intuitive and easy to learn. The dynamics engine is physics-based and enables the

program to accurately calculate the behavior of even highly complex models. Figure 11 shows the model created for spring mass problem and the results of the motion for its free vibration due to a step input to the system. This simulation model allows the user to change the system's parameters on the screen and the results are shown to match perfectly with those results obtained from the computation software tools or by experiments.
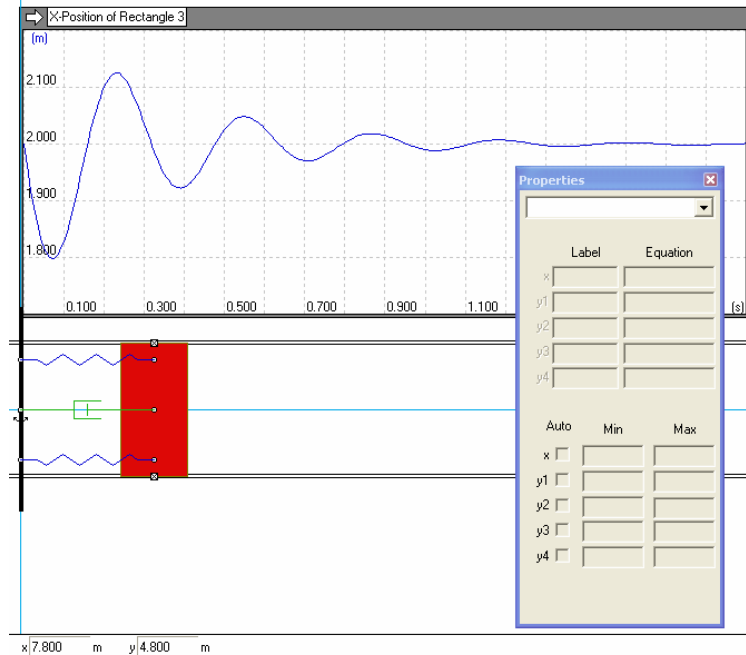

Figure 11. Simulation model created in Working Model© for the SDF spring mass system

Figures 10 represent simulation of under damped, critically damped, and over damped cases, respectively for the motion of the spring mass system. These are demonstrated during the lecture portion of vibrations course.
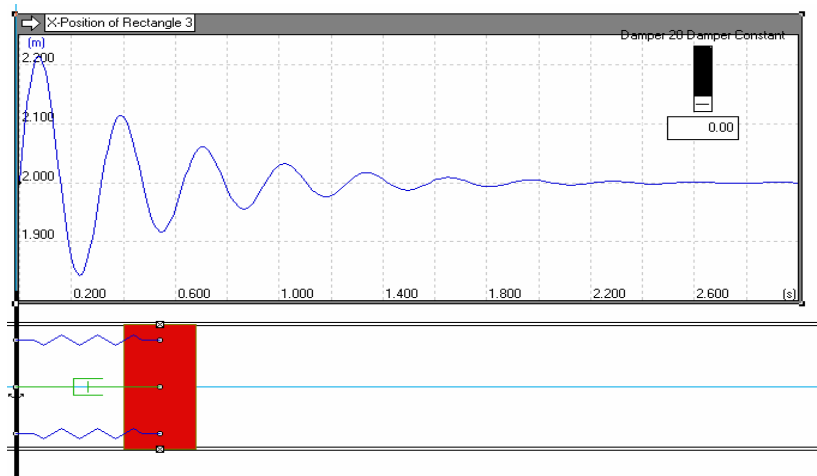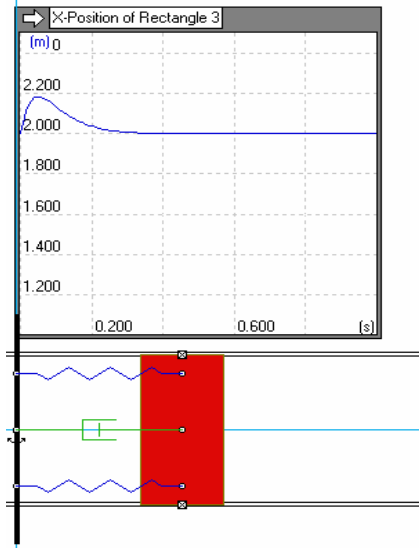

Figure 12 i. Under damped SDF system

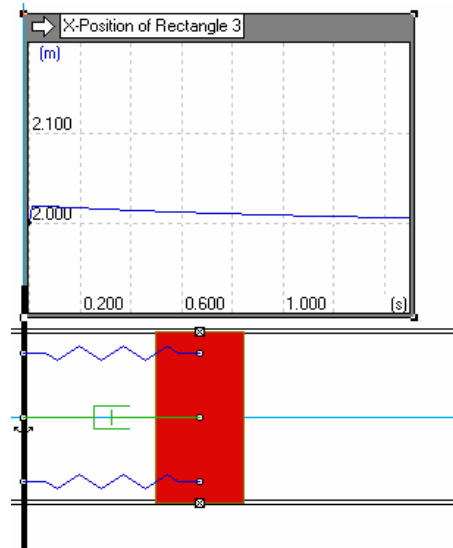**Figure 12 ii** Critically Damped system

Figure 12 iii Over damped system

Figure 13 shows a typical two degree of freedom spring mass system with damping. The system is under excitation of two forcing factions. These forcing functions could have different forms such as step, sinusoidal, triangular, etc. The task is to demonstrate to the students affect of all of the parameters for each component during the vibration of this system and later be able to use the experimental model , Model 210 Rectilinear plant shown in Figure 14, to demonstrate the control aspect involved in the system is the experimental setup for the same system in consideration.
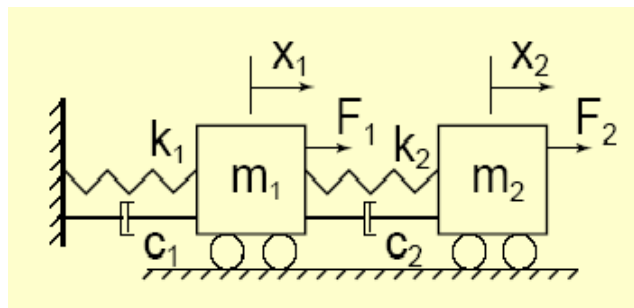

Figure 13. Two degree of freedom spring mass system with damping
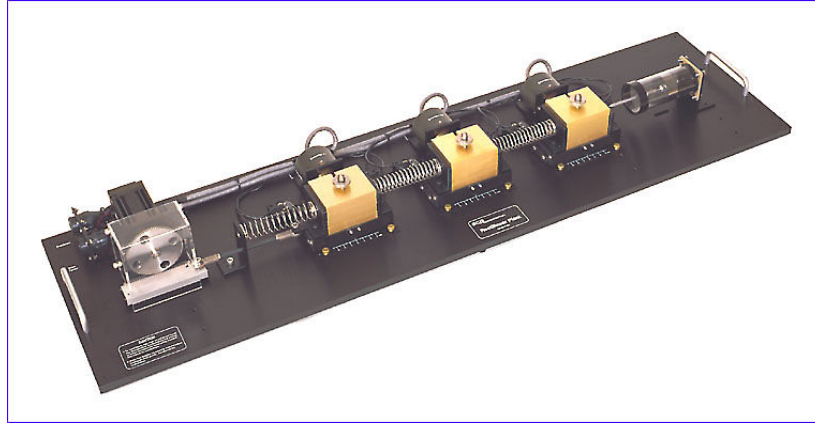
Figure 14. Model 210 Rectilinear plant used in Control Laboratory

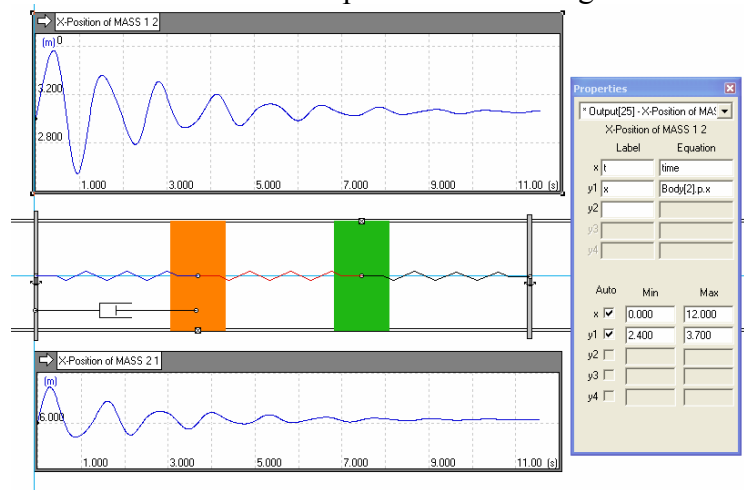Figure 15 represents simulation of one of the plants shown in Figure 14.

Figure 15 Simulation for one of the plants of Rectilinear system created in Working Model[©] for the 2DF spring mass with damping system

III. Conclusion

Modern software tools have potential to play an increasingly important role in engineering education. In this paper the implementation of recent computational, simulation and animation, and solid modeling software tools to engineering concepts has been presented. Animation, and simulation has been used to explain mathematically advanced topics in engineering to undergraduate students and increase teaching effectiveness of the faculty in the classroom.

Bibliography

[1] Kurt Gramoll, Rob Abbanat, "Interactive Multimedia for Engineering Dynamics," ASEE

Conf. Proc., Anaheim, CA, June 1995.

[2] Kraige, L.G., P.R. Gagnon, and T. Lin, "Motion Simulation Software for Personal Computer Use in Undergraduate Mechanics Courses," ASEE Conf. Proc. pp. 1569-1574, 1985.

[3] Richard T. O'Brien Jr. *"On Teaching State-Space Design Using A Matlab Real-Time Experiment"* SSST March 2003 Morgantown West Virginia.

[4] Bernstein, D. *"A plant taxonomy for designing control experiments"* Control system Magazine IEEE Vol. 21, issue 3, Jun 2001 pg 7-14.

[5] Mathcad, Mathsoft Engineering & Education, Inc., Cambridge, MA, http://www.mathsoft.com/.
    MATLAB, TheMathWorks, Inc., Natick, MA, http://www.mathworks.com/.

[6] D.J.Inman, 'Engineering Vibration,' 2nd Ed. Prentice Hall, NJ, 2003.

[7] Engineering and Engineering Technology Accreditation, Accreditation Board for Engineering and Technology, Inc., 111 Market Place, Suite 1050, Baltimore, MD, 21202, http://www.abet.org/criteria.htm.