DEVELOPING AN INTERACTIVE COMPUTER PROGRAM TO ENHANCE STUDENT LEARNING OF DYNAMICAL SYSTEMS

Daniel K. Howe and Oscar Barton, Jr.
Mechanical Engineering Department
George Mason University

Abstract

Today's students are quite accustomed to availing themselves of the latest in computer innovations and technology to aid in learning and the attainment of student outcomes. For example, use of tablets and cellphones in the classroom to take notes, collaborate on projects and to search the web for information is commonplace. Likewise, advancements in computer software and tools afford in-depth simulations of both mechanical and thermal systems. MATHEMATICA, with its symbolic and visual capabilities, is one such tool that, despite its robustness, is less utilized in classroom environments and is more integrated as a tool for those who pursue research in every discipline from economics to engineering. In this paper, the capabilities of MATHEMATICA are explored as a tool to model and visualize the forced mechanical response of viscoelastically-damped, multiple degree of freedom systems obtained through a Newtonian approach. Proportional damping facilitates the diagonalization of the resulting Eigen-value problem using a Cholesky Decomposition approach. In addition, multiple harmonics can be included as part of the forcing function. Displacement results for each mass permit the generation of graphical results and also provide the needed inputs for animated motions of all included masses. While the ultimate goal is to solve the dynamic response of general n^{th} degree of freedom systems, explicit results are presented for the second, fourth, and tenth order degree of freedom systems to demonstrate the versatility of the software. The program utilized an interactive and user friendly interface developed in MATHEMATICA.

Introduction

The development of low cost, high power computers continues to revolutionize how engineers perform their work in industry, academia, and research oriented positions. Through enhanced graphical capabilities, these systems support the modelling and analysis of complex systems that bring the solution of real world problems to the desktop. Universities typically provide these systems for students, recognizing the direct benefit towards the attainment of student outcomes, especially in engineering disciplines that need to comply with EAC-ABET criteria. Johannesen suggests that "When understood, more interesting and complicated situations can be explored with the help of computational tools"[1]. Tajvidi et al note that "Particularly in engineering dynamics, Computer Simulation and Animation [CSA] modules can demonstrate motion of particles and rigid bodies through computer animations, helping students picture the concepts taught in the course" [2]. Computers have their greatest impact not by displacing the entire course, but by providing an additional tool for aiding conceptual understanding, particularly by providing visualizations of complicated systems which are otherwise difficult to obtain and employ [2]. Therefore, the continuing focus on developing effective tools to exploit both the power and potential of computers for the benefit of student understanding offers an excellent opportunity.

Continuing to leverage the power of computers requires the development of software tools accessible to students. One platform for these tools is Wolfram Research's MATHEMATICA, a computational software program widely used for its versatility and robustness. Importantly, MATHEMATICA operates on personal computers and possesses a large existing user base and documentation of its features. This paper presents the work of one student's research project, which is focused on using the visual and computational tools in MATHEMATICA to develop a program to study dynamical systems. The goal of this project is to enhance the attainment of student learning outcomes. Using forced spring-damper-mass systems as test cases, a generalized program is demonstrated that features graphical user input of system parameters, interactive graphics to display mechanical responses, and user control of animations.

Background

This project originates from work performed by Roush as part of the Trident Scholar Program in the study of dynamical systems [4]. In that work, Roush used a Lagrangian approach to develop and solve models for the vibrations of a rigid plate mounted on viscoelastic supports under a variety of loads and initial conditions.

While that project features the use of MATHEMATICA to solve large systems of equations, it does not develop the process into a generalized program, instead only focusing on the problem at hand. As a starting point, the dynamics of a two degree of freedom mass-spring-damper system are modelled using Newtonian mechanics, and the system is subsequently expanded to include additional degrees of freedom. This approach enables the exploration of increasingly complex dynamic system, which a key advantage for students of computer-driven modelling.

The derivation of the governing equations used in this work follows a similar approach to that presented by Roush [4]. In short, the governing equations are developed, here using a Newtonian approach instead of a Lagrangian approach, the Eigenvalue problem is formed incorporating proportional damping, and finally, Cholesky decomposition is used to determine the system's mechanical response. All of these steps are performed using MATHEMATICA.

Problem Formulation

In this section, the equilibrium formulation yielding the governing equations and the preferred method to determine the mechanical response for a two degree of freedom system is presented.

Figure 1 presents the two degree of freedom system containing dampers, masses and springs under consideration.

While there are several approaches to determine the governing equations of a dynamical system, a Newtonian approach was used for this problem. By holding mass m_1 fixed and summing forces on mass m_2, the governing equation becomes

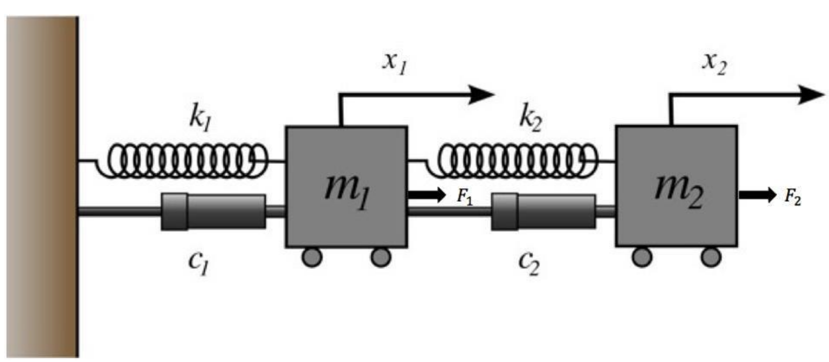

Figure 1: Spring-Mass-Damper System.

$$m_2\ddot{x}_2 + c_2(\dot{x}_2 - \dot{x}_1) + k_2(x_2 - x_1) = F_2(t) \quad \text{Eqn.(1)}$$

Repeating this process for the mass m_2 provides the following equation

$$m_1\ddot{x}_1 + (c_1 + c_2)\dot{x}_1 - c_2\dot{x}_2 + (k_1 + k_2)x_1 - k_2x_2 = F_1(t) \quad \text{Eqn.(2)}$$

Combining Eqn (1) and Eqn (2) into a matrix system of equations provides

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix}\begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{Bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 \\ -c_2 & c_2 \end{bmatrix}\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} + \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}\begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} F_1(t) \\ F_2(t) \end{Bmatrix} \quad \text{Eqn.(3)}$$

Equation (3) is a system of equations coupled both dynamically and statically. Damping complicates determining a solution in several ways. First, it is challenging to experimentally measure, and second, with damping present, there are no assurances that the associated eigenvectors would diagonalize the damped system, thus decoupling the system. In fact, Caughey and O'Kelly [13] have shown exactly that

$$[k][m]^{-1}[c] = [c][m]^{-1}[k] \quad \text{Eqn.(4)}$$

$$[m][k]^{-1}[c] = [c][k]^{-1}[m] \quad \text{Eqn.(5)}$$

$$[m][c]^{-1}[k] = [k][c]^{-1}[m] \quad \text{Eqn.(6)}$$

must be satisfied in order for the system to be diagonalized.

Whenever Eqn.(4), Eqn.(5), or Eqn.(6) do not hold, then a proportional damping model is recommended in which the damping matrix [c] is expressed as a linear combination of the mass and stiffness matrix given by

$$[c] = \alpha[m] + \beta[k] \quad \text{Eqn.(7)}$$

Constants α and β are selected to produce a desired damping ratio, based upon experimental results or design considerations.

Method of Solution

In light of the complicated structure of the governing equations, a modal solution is the preferred method of solution. An overview of this approach is presented here as it is not the focus of the paper, but demonstrates the method for solving the mechanical system response. This approach incorporates several steps including

1. A coordinate transformation using Cholesky Decomposition.
2. A modal transformation.
3. Solution of the problem in modal space.
4. Transformation back to original coordinate space.

The code that was developed incorporates these steps efficiently and can solve most systems quickly.

Application of MATHEMATICA

The significant contribution provided by MATHEMATICA to a student rests on fact that regardless of a student's ability or inclination to solve a complicated, coupled system of equations by hand, with a single program any student can rapidly create and analyze as complicated a system as he or she wishes. Naturally, an understanding of the mathematics performed by the program in the background is essential to the holistic understanding of the subject, but the program here functions similarly to an experiment: armed with theoretical knowledge, the student can rapidly explore numerous variations of the problem, unencumbered by the need to perform repetitious calculations. Equipping the student with this code enables a more efficient exploration of the dynamics at play in these systems.

The necessary computer code, which mirrors the process of solving these types of problems manually, are encapsulated into a software package that any user can load into their copy of MATHEMATICA. Once loaded into MATHEMATICA, users can then call the appropriate function for their operation. The function will then prompt the user to input system parameters, beginning with the number of degrees of freedom. Having stated the requisite system size, the user then inputs the remaining required system parameters – masses, stiffnesses, forcing functions, and initial conditions. In short, the user has to provide the physical data describing the system, as well as any initial conditions and forcing functions. Subsequently, the program automatically generates the matrix system of governing equations and applies the methods mentioned above to produce a system of solutions. Students therefore do not have to actually perform calculations on the system, merely describe the system, to analyze a system.

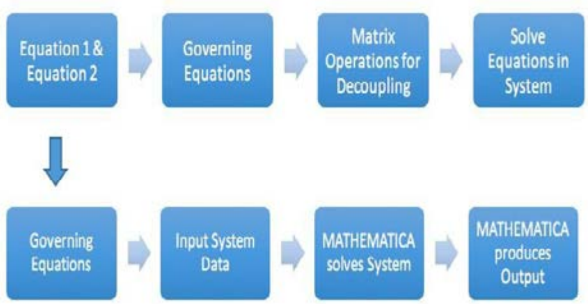

Figure 2: Upper Row details Solution Process, Lower Row Details MATHEMATICA Interaction.

A test example is generated using data from Inman [3]. Here, the system parameters are

$$m_1 = 9\,kg \quad k_1 = \frac{24N}{m}$$

$$m_2 = 1\,kg \quad k_2 = \frac{3N}{m}$$

With initial velocities of each mass taken as zero, the initial displacements for each mass are

$$x_1 = 1\,m \qquad\qquad x_2 = 0\,m$$

In addition, no forcing functions are applied and the damping coefficients are quite small:

$$\alpha = 0.00125\frac{1}{kg * s} \qquad\qquad \beta = 0.00125\frac{s}{kg}$$

Upon entering the information into the program, it executes the code required to simulate the system. Upon completion, it outputs (Figure 3):

- system information input by the user to achieve this result is displayed;
- the governing system of equations is presented in matrix form;
- the solution for the mechanical response of each mass;
- a displacement versus time graph.

Figure 3: Program Output for 2 DOF free, moderately damped system.

Several features of the program improve its usability and accessibility for the user. Notably, this graph has an adjustable time axis, allowing the user to dynamically adjust the length of time displayed between 0-100 seconds, including both the start and end times, to functionally zoom in on different parts of the graph as desired. Each of the output sections also includes the ability to hide that portion of the output as desired by using the arrow next to each section header. This feature provides significant value for very large systems where hiding the governing equations and their solutions can efficiently control the size of the output while preserving all information for future use. The program manages information this way to allow the user to prioritize which pieces of the information are most relevant without destroying the rest.

The most useful feature, however, is the animation section. For any size, nth degree system, a simple representative animation is produced where the masses move according to their respective solutions. The animation features provide an excellent visualization feature for students to fully understand the motion of these systems.

This base case displays the capabilities of the MATHEMATICA driven software. Operations that, on paper, require tedious computation have been streamlined such that, with available data, any user can analyze a system in a timely manner. Furthermore, the software here can perform these operations reliably time and time again, minimizing the opportunity for unseen mathematical error. These two properties unlock a world of potential for a student. While a sufficient knowledge of the mathematics required to the produce the solutions remains essential for proper understanding, a student with this software can experimentally explore these systems relieved of the cost of constructing a physical experiment and unburdened by the need for tedious and repetitive matrix calculations. Unlocking the potential for students to explore large systems such as these allows them to garner an improved understanding they might not otherwise garner from traditional methods of instruction.

To see how this software allows the user to experience how changing variables affects behavior, let us consider the previous example, but with both damping coefficients raised to 0.75. Entering this data produces Figure 4.

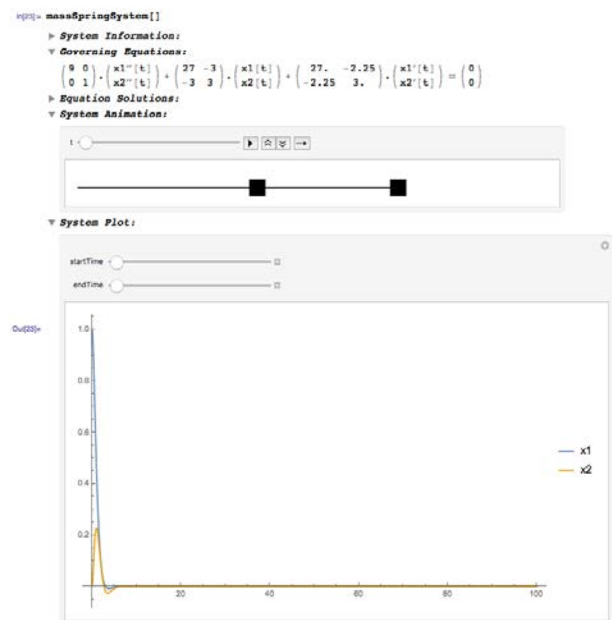

Figure 4: Program Output for 2 DOF free, heavily damped system.

While this is a rather extreme shift in performance, it does amply demonstrate how the software operates. The change of only two variables dramatically altered the behavior of the system, information which the program provides readily to the user.

For an additional exploration of these abilities, let us examine a system featuring the same mass and spring terms, and with high proportional damping coefficients of 0.75 but now add a multiple harmonic forcing function to further demonstrate the program's functionality, the results of which are seen in Figure 5.

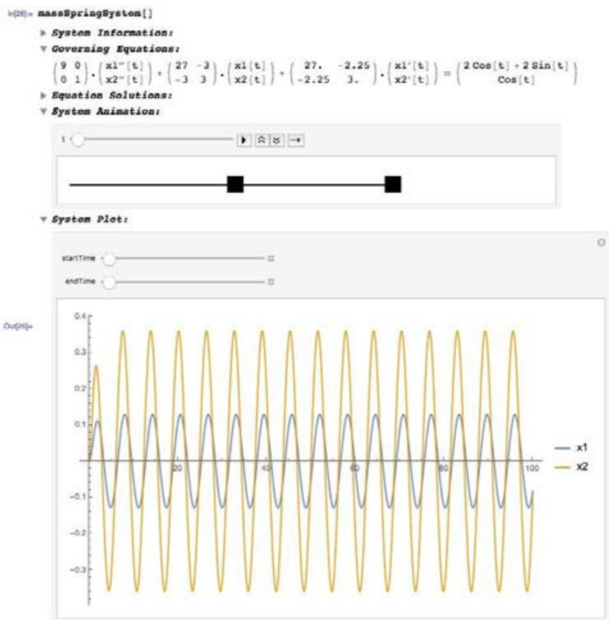

Figure 5: Program Output for 2 DOF forced, heavily damped system.

In this system, the addition of the forcing functions overwhelms even the strong damping coefficients. When shown in the actual program, each of these three related systems would appear next to each other, allowing for the behavior of the systems to be compared quickly. A student, therefore, can explore the changes in behavior caused by adjusting the values of variables and go back and see the prior behavior on demand.

For a sample system containing additional degrees of freedom, Inman provides the following data representing a simplified model of a building [3]:

$$m_1 = 4000\ kg \quad k_1 = \frac{5\ kN}{m} \quad x_1(0) = 0.025\ m$$

$$m_2 = 4000\ kg \quad k_2 = \frac{5\ kN}{m} \quad x_2(0) = 0.020\ m$$

$$m_3 = 4000\ kg \quad k_3 = \frac{5\ kN}{m} \quad x_3(0) = 0.010\ m$$

$$m_4 = 4000\ kg \quad k_4 = \frac{5\ kN}{m} \quad x_4(0) = 0.001\ m$$

Again here the initial mass velocities are all zero. Inputting this information to the program, produces the output shown in Figure 6.

Figure 6: Multiple DOF system, undamped, freely vibrating.

Although solving this system does not pose any challenge to a computer, it nonetheless illustrates the capability to solve larger systems and thus prove more useful as an educational tool.

To demonstrate the real usefulness of computers in this analysis, we present a large, ten degree of freedom system wherein the mass and spring terms do not equal each other, the damping constants are non-symmetric, and the initial displacements vary. The only simplification permitted is no forcing functions. This is intended as a stress test of the software to demonstrate its ability, not to model a real-world phenomenon, and the results are seen in Figure 7.

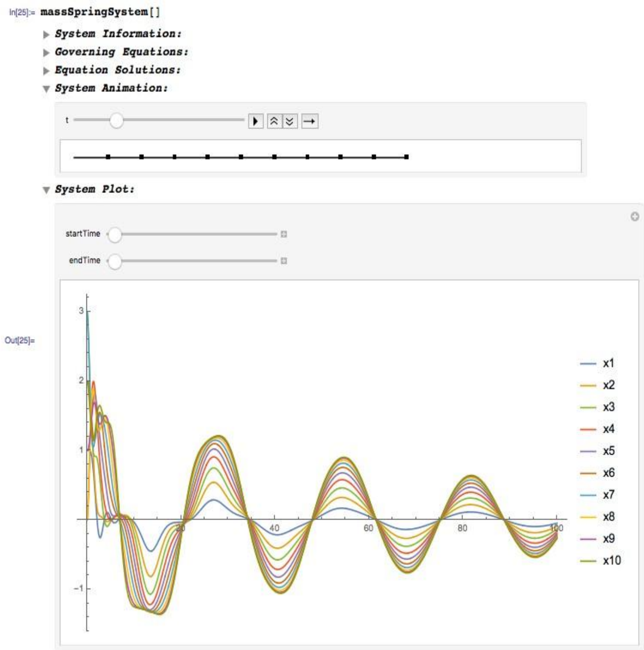

Figure 7: Multiple DOF system with varying system parameters.

Despite the increased size, the pattern continues of consistently scaling the size of the program output to the size of the system. Importantly, due to the size of this system, the governing equations and solutions have been minimized to demonstrate the output control features of the program. As explored in the previous examples, the program captures the properties of the mathematics and combines them with the efficiency and power of a computer to produce software that simplifies the work required of the user, particularly a student not yet well-versed in these systems.

Several features included in the software improve the flexibility of the simulation. First, the user can control the display of the solutions. Thus, for a relatively simple system a user may want to see the equations for immediate analysis, while for a system with many degrees of freedom the user may wish to hide the output in the interests of space until required. As mentioned before, the solutions are contained within a dropdown section which can be opened or closed at any time. Second, as touched on before, each system's output contains a summary of the of the system's data. This makes comparing two systems straightforward, since all the relevant data needed to describe it appears with the corresponding system response plot. Third, the solution of these problems on a computer often gives rise to functions with extremely small coefficients, once witnessed with an order of magnitude of 10^{-29}. Functions with coefficients this small contribute nothing significant to the system, and as a form of trimming unnecessary excess, the program excises any function with a coefficient of 10^{-10} or smaller. This allows for the impactful functions to appear at the fore, uncluttered by miniscule numbers with equally miniscule physical significance. These features aim to improve the ability of a user to actually use the program without losing track of the objective in tiny details that detract from the main goal of the model. Especially from a student perspective, cutting the waste generated from using the software greatly enhances the potential for deriving understanding of the system without a glut of useless data.

Implementation

While the code development is in its early stages, components are sufficiently well-developed for use in the department's elective vibration course ME 499. During its offering, feedback from students will be captured after demonstrating features of the code during class and used to modify the code. Extension to distributed systems – rods, plates and shells – is possible with continued support for the OSCAR undergraduate research support program.

Conclusion

Engineering students must accumulate and assimilate a truly staggering quantity of information throughout the course of their studies. While traditional teaching methods remain effective in imparting the necessary knowledge, technology increasingly offers new techniques which can add to the methods already employed. Specifically, research grade instruments such as MATHEMATICA can add substantially to the learning process for students by enabling them to explore areas of knowledge in ways unavailable through a whiteboard. As a representative case, software capable of modelling dynamic spring-mass systems of various sizes has been developed and presented, software which allows the student to run numerous experiments in the pursuit of understanding. By enabling the student to run these virtual experiments without the restrictions of cost and time requisite for building similar physical systems, the bounds of his or her understanding can be steadily increased by leveraging technology in this manner as a useful addition to teaching methods already widely deployed.

Bibliography

1. Johannesen, I. G. (2015, June), Stimulating the Learning Process in Mathematics and Numerics by Use of Computer Programs Like Mathematica. Paper presented at 2015 ASEE Annual Conference and Exposition, Seattle, Washington. 10.18260/p.24739

2. Tajvidi, M., & Fang, N. (2015, June), Application of Computer Simulation and Animation (CSA) in Teaching and Learning Engineering Mechanics. Paper Presented at 2015 ASEE Annual Conference and Exposition, Seattle, Washington. 10.18260/ p.23560

3. Inman, Daniel J. (2001). Engineering Vibrations. Upper Saddle River, NJ: Prentice-Hall Inc.

4. Roush, Midshipman 1/C Angela Marie. (2010) Dynamic Analysis of an Optical Laser Platform. (Report No. 393). Annapolis, MD: United States Naval Academy.

5. Caughey, T.K., & O'Kelly, M.E.J. (1965). Classical Normal Modes in Damped Linear Dynamic Systems. Journal of Applied Mechanics. 583 – 588. Retrieved from http://appliedmechanics.asmedigital collection .asme.org/

6. Caughey, T.K. & O'Kelly, M.E.J.. (1961). Effect of Damping on the Natural Frequencies of Linear Dynamic Systems. The Journal of the Acoustical Society of America. 33(11). 1458 – 1461.

Biographical Information

Daniel Howe is a Junior in Mechanical Engineering at George Mason University. A native of Fairfax, VA, he enrolled in the major in January 2015. In addition to the curriculum, he researches the mechanics of dynamic systems as a research assistant to the Department Chair, Oscar Barton, Jr., PhD, PE. In particular, his research focuses on the computer modeling of vibrations in dynamic systems. He also provides academic support as a tutor for mathematics, science, and engineering in the Volgeneau School of Engineering, and is the Secretary of the George Mason University Chapter of the American Society of Mechanical Engineers.

Oscar Barton, Jr., Ph.D., P.E. is a Professor of Mechanical Engineering at George Mason University. A native of Washington, D.C., Professor Barton received his B.S in Mechanical Engineering from Tuskegee (Institute) University, his M.S in Mechanical Engineering and Ph.D. degree in Applied Mechanics from Howard University. He joined the faculty of Mechanical Engineering Department at George Mason University fall 2014, after completing a 22 year career at the U.S. Naval Academy. His research focuses on the development of approximate closed form solutions for linear self-adjoint systems, those that govern the responses of composite structures, and the analysis of dynamic systems. More recently, he has mentored numerous midshipmen through independent research projects and has directed two Tri-dent Scholars, the Naval Academy's flagship research program. He has published over 50 journal and conference articles on these topics. He is actively involved in curriculum development and program assessment. He chairs ASME Committee on Engineering Accreditation. He serves as Commissioner for Engineering Accreditation Commission of ABET, Inc. and was a program evaluator for six years prior to joining the commission. He holds a professional engineering license in the State Maryland. He is a member of the Board of Education, ASME.