Simulation and Visualization of Dynamic Systems

Michael P. Hennessey, Ph.D.
School of Engineering
University of St. Thomas
101 O'Shaughnessy Science Hall
2115 Summit Avenue
St. Paul, MN 55105-1079

Abstract

Ideas from both the system dynamics and CAD perspective were integrated together to create a new course entitled *Simulation and Visualization of Dynamic Systems*, a technical elective in mechanical engineering at the University of St. Thomas. One may think of such a course as a type of modern system dynamics, in which CAD and animation are brought in for significant visualization to enhance understanding of mechanical systems that incorporate motion as well as to appeal to a wide audience, some of whom may not be technically oriented. This paper describes such a course and its advantages. Sample student project work is shown that emphasizes applied mathematical modeling, CAD modeling, modeling and simulation using MATLAB® and Simulink®, and animation of specific scenarios of interest using SolidWorks® with the Animator add-on. Projects show-cased include: (1) the zero-friction skiing brachistochrone problem, (2) the kinematic car with both front and rear wheel steering, (3) the N-pendulum chain approximation of a weighted string, and (4) a tracked vehicle.

Introduction

Traditionally, computer-aided-design (CAD) related topics have been taught in courses separate from *System Dynamics* in a BSME program. CAD is often taught in one or more freshman or sophomore courses and focuses largely on static geometric issues associated with parts, assemblies, and drawings. It is commonly paired with a track on drafting. *System Dynamics* is often a sophomore or junior level mechanical engineering course and emphasizes applied mathematical modeling of systems from different energy domains, incorporates such simulation software as MATLAB® and Simulink®, and could be enhanced by CAD. At the University of St. Thomas, currently *System Dynamics* is not a required course in mechanical engineering largely due to the existence of the liberal arts core curriculum comprised of 84 semester credits outside the ME major. Also, it is the opinion of the author that engineering students could greatly benefit from exposure to system dynamics topics and there has been occasional anecdotal evidence to support this claim. In response to the above observations, a new 4 semester credit technical elective course entitled *Simulation and Visualization of Dynamic Systems* was created that integrates aspects of visualization (including animation) with a traditional system dynamics course. Prerequisites include *Linear Algebra and Differential Equations* and *Engineering Graphics* with 3 hours per week being allocated for lecture and 3 hours per week for a simulation laboratory (lab) that utilizes PCs equipped with software such as SolidWorks (2005), including the add-on Animator, MATLAB®, and Simulink®.

The course objective, description, and related courses are given by:

Objective: Gain experience in dynamic modeling, simulation, and visualization of many different mechanical systems using applied mathematical techniques and modern software important to mechanical, electrical, and systems engineers working in industry or students preparing for graduate school in engineering.

Description: Many engineering systems are inherently dynamic in nature. Characterizing and designing such systems requires mathematical modeling, simulation, and visualization using modern software such as MATLAB®, Simulink®, and SolidWorks, possibly with add-on modules. Lectures focus on the detailed applied mathematical modeling of a variety of systems from different energy domains, including mechanical translational, mechanical rotational, hydraulic (fluid), thermal, and electrical. The simulation lab has 4 essential components to it: (1) software training, including use of "add-ons" and "toolboxes" as needed, (2) developing dynamic models using MATLAB® and Simulink®, (3) creating CAD models of mechanical systems, and (4) integrating the dynamic models with the visualization to create computer animations of the resulting motions of the mechanical systems. Students also work on a team-based dynamic simulation and visualization of mechanical systems project.

Related Courses: At the University of St. Thomas there are several related courses in the sense that CAD, mechanics, or dynamic modeling has been used to some extent: *Engineering Graphics* (Hennessey et al., 2002), *Kinematics & Mechanism Design* (Hennessey, 2003; Hennessey & Kumar, 2006c), *Mechanics II* (Hennessey & Hacker, 2006b), and *Machine Design and Synthesis* (Hennessey, 2001).

Lecture Topics and Demonstration Aids

Lecture topics (along with homework) in the course were traditional topics in system dynamics selected from the course's textbook (Close et al., 2002) and included: (1) introduction and mathematics review, (2) translational mechanical systems, (3) standard forms for system models, (4) rotational mechanical systems, (5) electrical systems, (6) transform function analysis, (7) developing a linear model, (8) electromechanical systems, (9) thermal systems, (10) fluid systems, and (11) various applications. Since this is intended to be a first course in system dynamics, bond graph theory (Karnopp et al., 2006) was not discussed, although the idea of unification was emphasized. Analogies between 5 different energy domains minus of course the thermal inductor, since it doesn't exist, were discussed. Applications introduced systems with elements from multiple energy domains present. Some theory was presented but the main focus was on techniques and applications. A few demonstration aids were utilized during lecture; these include a model steam engine to better understand rotational inertia and a time constant for a first order system, a highly contoured plastic water bottle to understand variable fluid capacitance, and a lead fishing sinker of approximately spherical shape to better understand transient heat transfer, heat capacitance, along with thermal resistance.

Simulation Lab Topics

A total of 10 weekly 3 hour simulation labs were utilized in the course, with the remaining 3 labs devoted exclusively to the simulation and visualization project. The simulation labs were designed to build the student's skills in a vertical manner (at least early on) and include the following:

- MATLAB® Basics
- More MATLAB® Basics
- Simulink® Basics
- Simulink® Applications: Vibration Absorber and Elevator Dynamics
- Inverted Pendulum
- Unicycle on a Sphere
- Semi Tractor-Trailer
- Animation Basics
- Accumulator Simulation and Animation
- Temperature Control in a Die

Collectively, these simulation labs cover a wide range of topics, including some interesting applications that contain subsystems from each of the energy domains covered in lecture plus some classical feedback control. Detailed handouts were also prepared for each lab and 2 textbooks were required (Hanselman & Littlefield, 2005; Dabney & Harman, 2004).

MATLAB® Basics and More MATLAB® Basics

Because MATLAB® is so essential to the course, the first 2 simulation labs were designed to endow the student with some facility in using MATLAB®, especially as it is applied to the simulation of fairly generic dynamic systems. Some critical topics in the MATLAB® Basics simulation lab include 1D arrays, 2D arrays, script style M-files, 2D graphics and plotting basics, functions, function style M-files, solution of simply structured ordinary differential equations, and user interface basics. A "cheat-sheet" was also provided that summarizes some of the most commonly used MATLAB® commands. More MATLAB® Basics continues on with the following topics: polynomial functions, anonymous functions, detailed 2D and 3D plotting, miscellaneous functions, some basic programming concepts and additional commands (global vs. local variables, for loop, while loop, and if-else-end commands). Upon completion of these 2 labs, the student is capable of using MATLAB® quite effectively to solve a variety of basic dynamic systems problems and is prepared to take the next logical step – learning Simulink®.

Simulink® Basics

After a brief tutorial on the concept of a simulation diagram (sometimes loosely referred to as a block diagram), the basics of Simulink® were covered, including block libraries, commonly used blocks, constructing and editing simulation diagrams (or "models"), interfacing with MATLAB®, and running simulations using the solver. Several problems were assigned as in-class exercise (an "mbk" mechanical translational system and a nonlinear pendulum).

Simulink® Applications: Vibration Absorber and Elevator Dynamics (see Figures 1 and 2)

Several more significant applications for Simulink® included a vibration absorber attached to a vibrating machine and an elevator supported by a compliant cable that is driven by a gearbox and in turn a motor (former MIT "system dynamics" problems). In each case, the equations of motion were derived and presented, along with the simulation diagram, which formed the basis of a good class discussion. Students entered the corresponding simulation diagrams into Simulink® and performed a variety of specific simulations, in addition to documenting them.

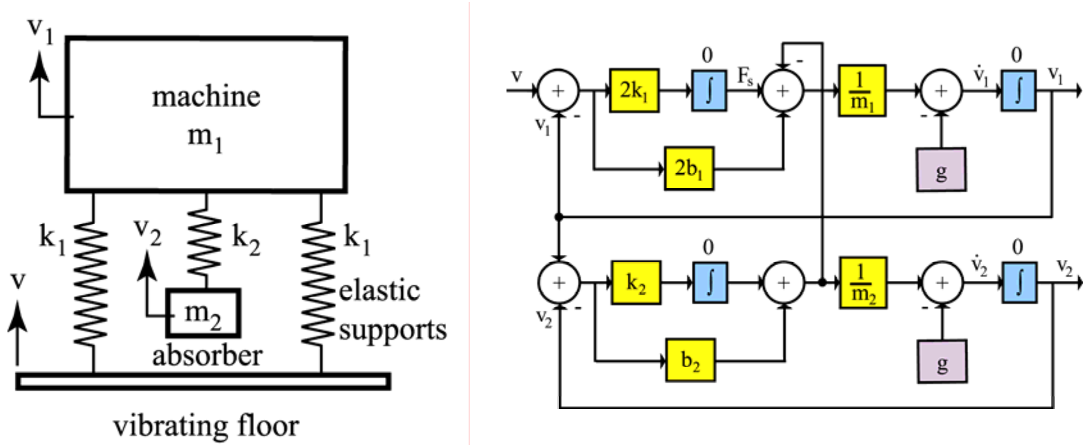


Figure 1 Conceptual sketch of 1D vibration absorber attached to a vibrating machine and associated simulation diagram (with damping present as well).

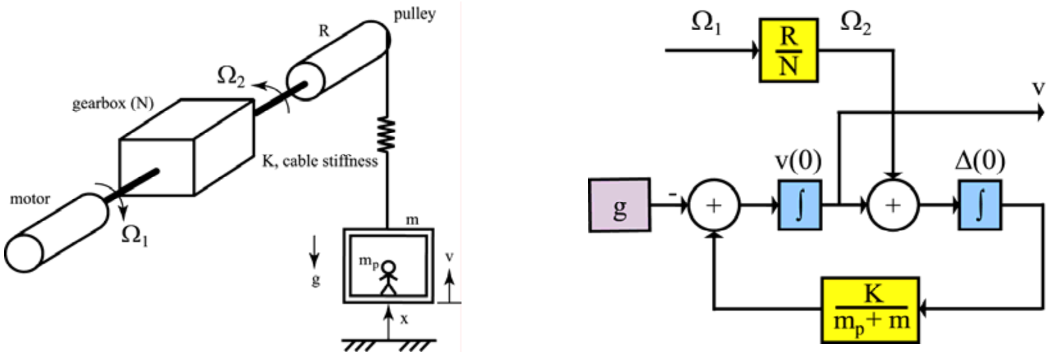


Figure 2 Conceptual sketch of an elevator system and corresponding simulation diagram.

Inverted Pendulum (see Figure 3)

A simulation diagram was created for the classic inverted pendulum stabilization problem augmented to include a dc motor (Ogata, 1997). This problem offered a natural opportunity to introduce control aspects, namely proportional and derivative (PD) style classical control. Another perspective is that elements from several energy domains (i.e. mechanical translational, mechanical rotational, and electrical) are integrated together in the system. Students created the simulation diagram within Simulink® and performed specific simulations based on use of previously established parameters.

Unicycle on a Sphere (see Figure 4)

An intriguing nonholonomic mechanical system is that of a single-wheeled vehicle (i.e. a unicycle) moving without slip over a curved surface, in this case a sphere (Hennessey, 2006a). It represents a generalization of the well-known kinematic car driving on a planar surface. Inputs to the system are the steering yaw rate and wheel speed. Equations in control input form are:

$$\begin{bmatrix}\dot{\theta}\\ \dot{\psi}\\ \dot{\gamma}\\ \dot{\phi}\end{bmatrix}=\begin{bmatrix}0\\0\\1\\0\end{bmatrix}\omega+\frac{r}{R}\begin{bmatrix}c_\gamma / c_\psi\\ s_\gamma\\ -c_\gamma t_\psi\\ 1/(r/R)\end{bmatrix}\dot{\phi}$$

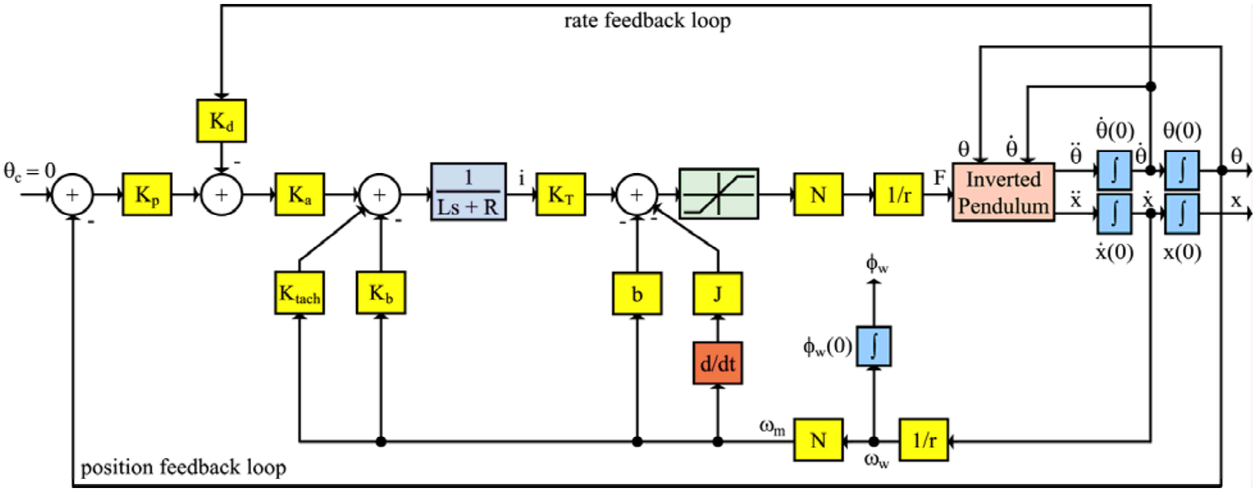


Figure 3 Simulation diagram of an inverted pendulum stabilized with proportional and derivative feedback control action.

Students modeled the system in Simulink® and generated portions of 3D contact point trajectory data previously determined. One of the interesting simulation diagrams used possessed an unavoidable algebraic loop embedded within it. This offered a convenient time to discuss more generally the topic of algebraic loops, as it is non uncommon for practicing engineering analysts to encounter them when modeling various electro-mechanical systems. Fortunately, these days many of the integration algorithms used within the solver are better able to deal with algebraic loops than in the past.

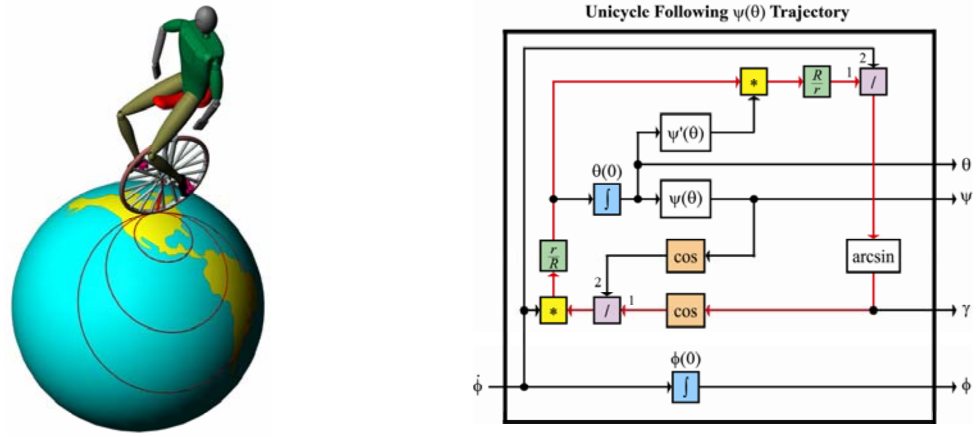


Figure 4 Unicycle on a sphere with corresponding contact point trajectory driven simulation diagram that contains an embedded algebraic loop shown in red (Hennessey, 2006a).

Semi Tractor-Trailer (see Figures 5 and 6)

Anyone who has towed a trailer behind a vehicle is aware that significant dynamic effects are often present and yaw instability can ensue, caused by various parameters, such as vehicle speed. To better understand this system, especially during transient events, students studied the El-Gindy model (1989) using Simulink®. Various scenarios of interest, such as observing the effect of a constant, or step input in steered wheel angle on the rest of the system were studied through simulation. Students also learned about a useful concept referred to as "hierarchical modeling," (implemented as "subsystems" within Simulink®) as it was convenient to use hierarchical modeling when modeling this system, as well as others.

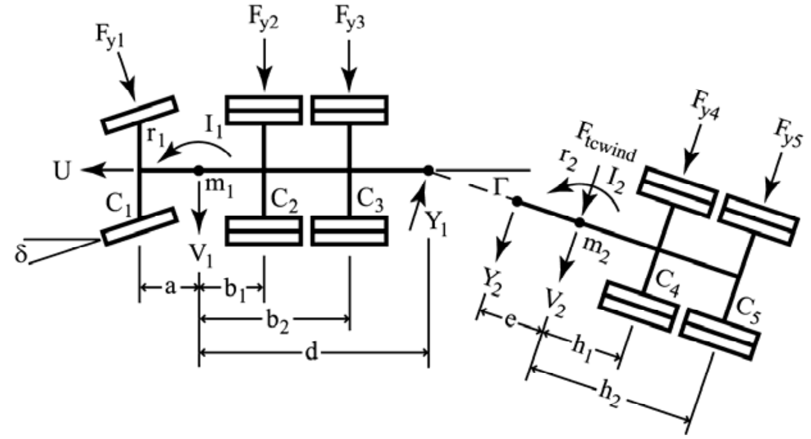


Figure 5 Conceptual sketch of the El-Gindy semi tractor-trailer.

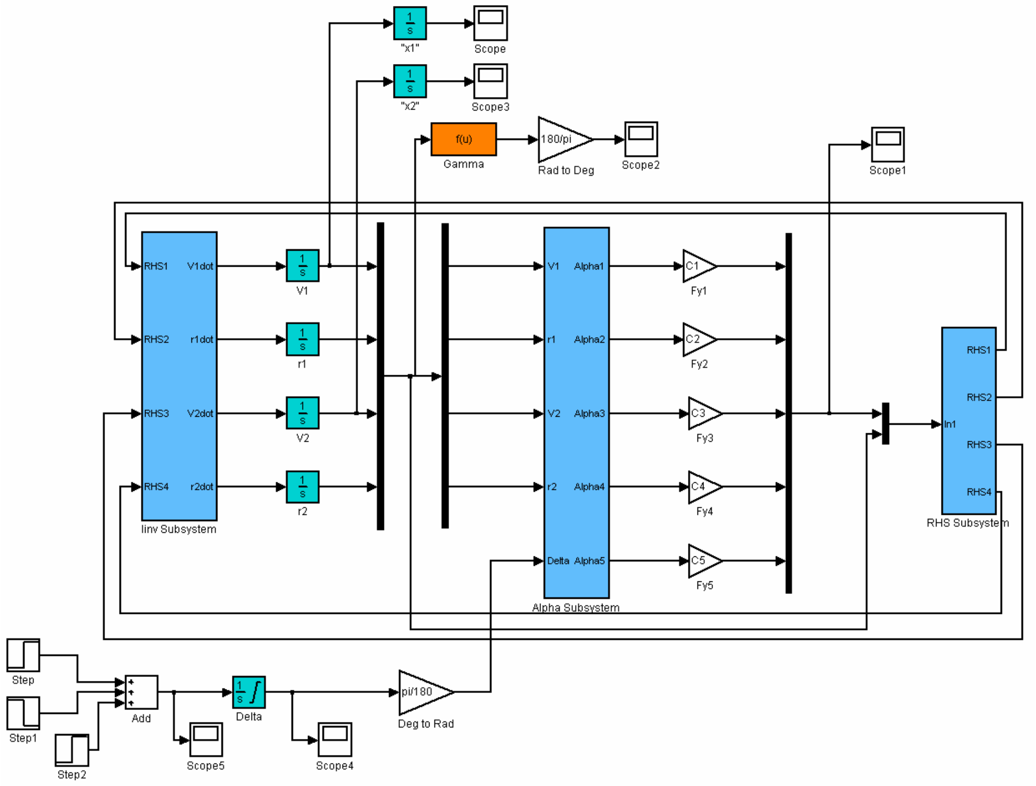


Figure 6 Top-level Simulink® model of the El-Gindy semi tractor-trailer.

Animation Basics

How can one realize the benefits of combining CAD with dynamic simulations to create animations? If possible, such a capability represents a very generic and powerful computer-aided-engineering (CAE) analysis tool since dynamics plays a significant role in many mechanical systems, all of which possess CAD representations. The approach presented here is to use SolidWorks driven by MATLAB® data. Several other valid, although non-CAD intensive approaches are described by Mazzei and Scott (2006) who use Maple® and Nordenholz (2006) who uses MATLAB® exclusively. Fortunately, SolidWorks allows one to create different "configurations" of assemblies based on use of an external "design table" from Excel that has been created from simply copying and pasting MATLAB® data from the array editor environment, with of course proper formatting, including row and column header definitions. In essence then, each row in the design table represents for a given instant of time, a set of position states for the system under consideration (i.e. "time-sliced" data). Generating this type of data is easy within MATLAB® and these position states (vs. velocity, or other states) uniquely define the location of each movable part or subassembly within the top-level assembly. Typically, several hundred configurations would be created. Given the design table along with the SolidWorks assembly file, Animator (a SolidWorks add-on) can then be used to create a reasonably high quality animation (.avi file type) lasting for a few seconds that can be played on common media players such as Windows Media Player. A detailed description of the entire process outlined here was given to the students along with a short demonstration and during lab they were encouraged to create an assembly and an animation of their own choosing.

Accumulator Simulation and Animation (see Figure 7)

An accumulator (Karnopp et al., 2006) is a mechanical translational and fluid mechanics device analogous to an electrical capacitor and is commonly used in hydraulic systems. Within the simulation diagram differential causality is present in 3 blocks. This offered an opportunity to discuss the topic of differential causality with students in more detail (without getting into bond graph theory) and its often practical resolution using a Pade approximation in the s-domain (i.e. $s / (\tau s + 1)$ with $0 < \tau$ and τ sufficiently small but not so small as to introduce numerical problems). For the students, this was the first time that they could participate in all steps in the process, including: (1) dynamic modeling of a mechanical system and obtaining the differential equations, (2) creating a CAD model in SolidWorks, (3) performing simulations in Simulink® and porting data into MATLAB®, (4) creating a design table in Excel with MATLAB® data, and (5) creating time-sliced configurations and an animation using Animator.

Temperature Control of a Die (see Figure 8)

This problem was worked on in the early-nineties by the author in response to a friend (and his colleague) who was working on a die temperature control problem in a factory. The die that was used to manufacture parts was not working properly because the temperature in the die would not easily stabilize in the 500 deg F range. Rather, it oscillated widely, requiring many hours to finally stabilize and during this time period many poor quality parts were being made, with money being lost. The purpose of this lab was to expose students to a practical temperature control problem using MATLAB®/Simulink® and to develop an intuitive sense of what classical control is all about without going through a formal design or tuning process. Using a simulation diagram with the non-control gain related parameters provided, students investigated the effects of varying the controller gains (i.e. K_p, K_i, and K_d) and attempted to replicate the qualitative temperature oscillation that was observed in the die. By fixing K_p and increasing K_i and K_d uniformly, the transient response became progressively more oscillatory and typical of what was observed in the factory die (it appeared as though those gains were set too high).

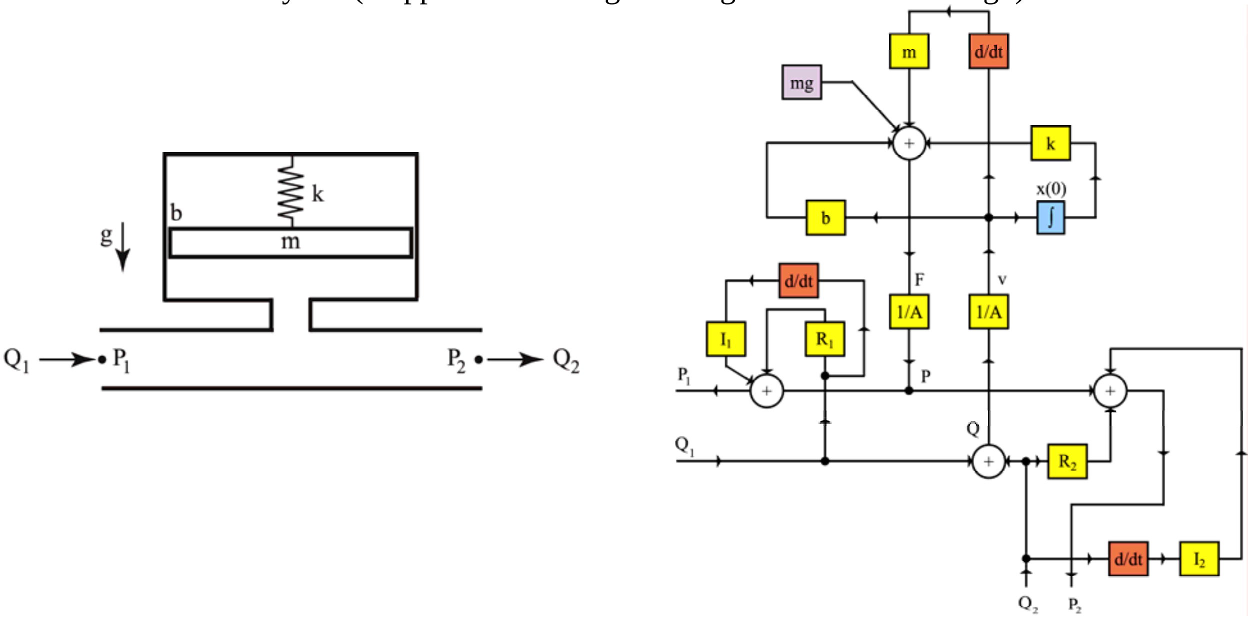


Figure 7 Accumulator and associated simulation diagram.

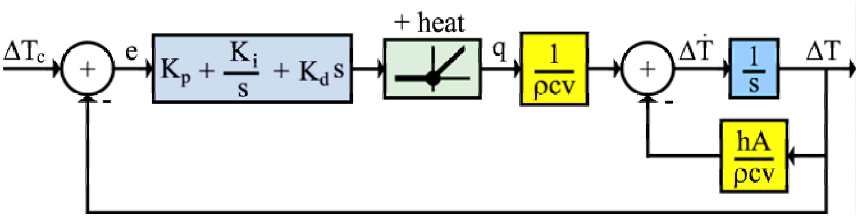


Figure 8 Temperature control of a die using classical control.

Project Descriptions and Sample Student Work

The objective of the project is to gain expertise and experience in dynamic modeling, simulation, and visualization through study of a specific mechanical system. In the process the student:

- Understands and/or develops system equations of motion for a specific open-loop or closed-loop mechanical system
- Creates appropriate and mathematically correct simulation models using MATLAB® and Simulink®
- Makes simulation runs and creates plots using MATLAB® for specific scenarios
- Creates CAD models of the system using SolidWorks
- Builds several animations of interesting scenarios that can be played on Windows Media Player
- Prepares a technical report (with other team members) and makes a short presentation to the class

This past semester (Spring 2007) a total of 7 students completed this technical elective course and worked on a total of 4 projects, each of which is described briefly below and presented with sample student work (Figures 9-12). While not practical to make available in this paper, interesting animations that bring the static images to life exist for all of the projects and are available for viewing (please contact author). Lastly, due to space limitations and at the risk of not being totally complete, only snippets of the work done are presented, to capture a glimpse of what these projects are all about.

Zero-Friction Skiing Brachistochrone Problem (see Figure 9): A practical interpretation of this problem is how to perform a "traverse" down a constant slope mountain in minimum time? (By the way, it is not along a straight line!) Equations of motion are available (4th order state-space model) and the minimum-time optimal control problem is solved numerically using a so-called "shooting method" whereby an ordinary differential equation obtained from the calculus of variations is solved assuming a particular initial condition (skier orientation in this case). Through appropriate iteration of the initial condition, convergence is achieved and the resulting trajectory (or route) is run through the desired final destination. It can also be shown that the optimal control input, namely the yaw rate, is constant. Using MATLAB®, solve this problem for interesting scenarios and create associated animations using SolidWorks and the add-on Animator.

$$\dot{x} = \frac{-V}{\sqrt{1+b^2}}\cos\phi,\ \dot{y} = -V\sin\phi, \dot{V} = \frac{bg\cos\phi}{\sqrt{1+b^2}},\ \dot{\phi} = \frac{bg\sin\phi}{V\sqrt{1+b^2}}$$

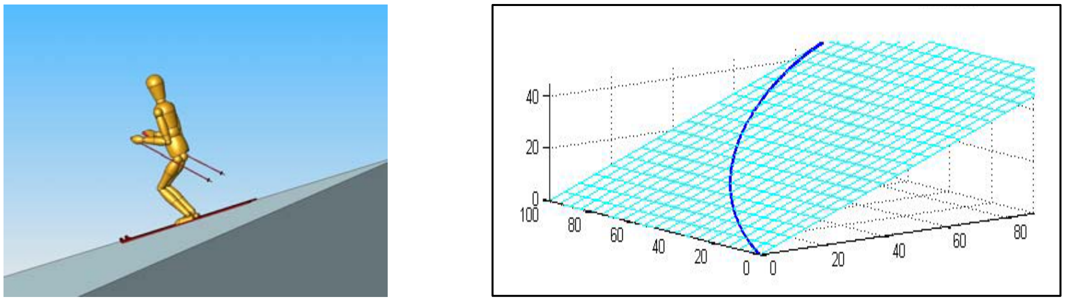
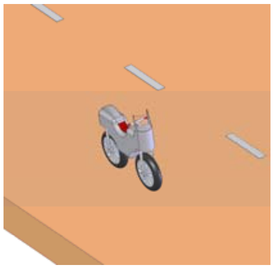


Figure 9 "IKEA-boy" <http://solidworks.cad.de/lib_mensch01.htm> skier is traversing the minimum-time route when starting from rest, with accompanying closed-loop state equations (Lindsey Hines and Ann Majewicz).

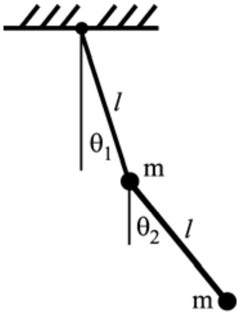
Kinematic Car with both Front and Rear Wheel Steering (see Figure 10): Consider a 4-wheeled vehicle such as a car, lawn mower, golf cart, etc. performing low speed maneuvers without wheel slip such as parallel parking, driving on city streets, passing, or perhaps maneuvering in an open parking lot, all in a controlled manner. Kinematic models of such vehicles exist and your job will be to follow the derivation (Murray et al., 1994) and extend it to the case with both front and rear wheel steering (assume symmetry). That is, as the vehicle turns to the left (or right) both the front wheels and the rear wheels turn the same amount. Note: here you may assume use of the short axle "bicycle model," vs. incorporating Ackerman steering effects (Wong, 1993). Given such an analytical model, using MATLAB®, Simulink®, and SolidWorks (with Animator) create some interesting simulations and animations that are realistic.

N-Pendulum Chain Approximation of a Weighted String (see Figure 11): The single pendulum dynamics problem has been solved years ago. What about adding pendulum (or link) elements creating an "N-chain" so as to more closely approximate a weighted string? To facilitate this, Lagrangian dynamics will be used successively, starting out with N = 1 (solution provided). Can you solve analytically the N = 2, 3, 4, or even 5 problem correctly by yourself? If you are interested, a symbolic manipulation package can be used to determine the equations of motion for larger N problems. Once the equations are determined and the simulation model is created in MATLAB® or Simulink®, try out different initial conditions for the pendulum angles (θ's) and angular velocities (ω's) and animate the intriguing and seemingly chaotic motion that can occur using either SolidWorks with Animator or MATLAB® only, since a simple conceptual model is adequate in this case. Lastly, be aware of distant mate limit effects in SolidWorks.



$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\theta-\phi) \\ \sin(\theta-\phi) \\ \dfrac{2\sin\phi}{l} \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} u_2$$

Figure 10 The kinematic car model depicted in CAD with both front and rear wheel steering in preparation for a performing a passing maneuver on a highway and described analytically in control input form (Lina Salah and Michael Zimmerman).

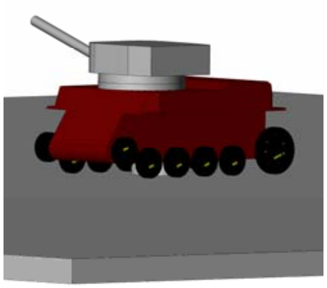


$$\ddot{\theta}_1 = -\frac{g}{l}\sin\theta_1 - \frac{1}{2}\begin{bmatrix} \ddot{\theta}_2\cos(\theta_1-\theta_2) + \\ \dot{\theta}_2^{\,2}\sin(\theta_1-\theta_2) \end{bmatrix}$$

$$\ddot{\theta}_2 = -\frac{g}{l}\sin\theta_2 - \ddot{\theta}_1\cos(\theta_1-\theta_2) + \\ \dot{\theta}_1^{\,2}\sin(\theta_1-\theta_2)$$

Figure 11 The N = 2 pendulum chain approximation of a weighted string: conceptual figure and corresponding coupled ordinary differential equations (Robert Ertel).

Tracked Vehicle (see Figure 12): Imagine a tracked vehicle, such as a tank or bulldozer, moving at low speeds and performing various maneuvers on a flat, horizontal surface. Develop a kinematic model, complete with appropriate differential equations. As a valid approach, you may assume use of an invisible, virtual, superimposed "no-slip 2-wheeled cart" model. As each virtual wheel rotates without slip, the motion of the corresponding track can be inferred accordingly. Can your tracked vehicle "spin-on-a-dime," go around a circular track, drive straight forward and in reverse, and perform a figure 8 maneuver? Create your own tracked vehicle model (perhaps based on a physical scale model) in SolidWorks and animate different scenarios of interest using Animator driven with data from MATLAB®.



$$\dot{x} = \frac{R_w}{2}(\omega_L + \omega_R)\cos\theta$$
$$\dot{y} = \frac{R_w}{2}(\omega_L + \omega_R)\sin\theta$$
$$\dot{\theta} = \frac{R_w}{d}(\omega_R - \omega_L)$$

Figure 12 Tracked vehicle modeled in SolidWorks with associated 3rd-order state equations with 2 angular rate inputs, one for each virtual wheel (Matthew Gartmann and Michael Moran).

ABET Compliance and Student Feedback

Consistent with other courses within the ABET (EAC division) accredited BSME program (ASEE, 1998), learning outcomes were identified based on the well-known "(a)-(k)" ABET 2000 outcomes. For example, lecture homework and simulation lab problems assigned require application of lower division undergraduate mathematics (ABET outcome (a)). In total, the course had outcomes that comply with 5 of the traditional (a)-(k) learning outcomes. Student feedback from the course evaluations was slightly above "very good," with one student commenting that it should be a required course for all mechanical engineering students.

Conclusions and Recommendations

The major conclusion is that integration of traditional CAD and animation with system dynamics based simulation into a new course (in this case entitled *Simulation and Visualization of Dynamic Systems*) is a very powerful idea, makes tremendous sense from a pedagogy point of view, and lastly, the technology is here today and very accessible to undergraduate engineering students.

Acknowledgements

The author acknowledges the decision by the School of Engineering to offer this technical elective course (with 7 students) and the engaging students in the course, most of whom are planning on pursuing graduate degrees in engineering: Robert Ertel, Matthew Gartmann, Lindsey Hines, Ann Majewicz, Michael Moran, Lina Salah, and Michael Zimmerman.

References

ASEE (1998). How do you measure success? Washington DC: ASEE.

Close, C. M., Frederick, D. K., & Newell, J. C. (2002). Modeling and Analysis of Dynamic Systems (3rd ed.). New York, NY: John Wiley & Sons.

Dabney, J. B., & Harman, T. L. (2004). Mastering SIMULINK®. Upper Saddle River, NJ: Pearson Prentice Hall.

El-Gindy, M. (1989). Directional Response of a Tractor Towing a Semitrailer. International Journal of Vehicle Design, 10(2), 210-216.

Hanselman, D., & Littlefield B. (2005). Mastering MATLAB® 7. Upper Saddle River, NJ: Pearson Prentice Hall.

Hennessey, M. P. (2001). A Structured Activity Based Approach to Teaching Machine Design. In Proceedings of the ASEE North Midwest Section Annual Conference (Session II-B). Grand Forks, ND: ASEE.

Hennessey, M. P., Jaedike, A. C., & Rhode, P. S. (2002). Galleries of CAD Generated Imagery. In Proceedings of the ASEE North Midwest Section Annual Conference. Madison, WI: ASEE.

Hennessey, M. P. (2003). The Kinematic Car: Teaching Undergraduates Nonholonomic Mechanical System Basics. In Proceedings of the ASEE North Midwest Section Annual Conference. Ames, IA: ASEE.

Hennessey, M. P. (2006a). Visualization of the Motion of a Unicycle on a Sphere. The International Journal of Modelling and Simulation, 26(1), 69-79.

Hennessey, M. P., & Hacker, L. A. (2006b). Dynamic 3D Visualization of Stress Tensors. In Proceedings of the Annual ASEE Conference (Session 1568). Chicago, IL: ASEE.

Hennessey, M. P., & Kumar, S. (2006c). Integrated Graphical Game and Simulation-Type Problem-Based Learning in Kinematics. The International Journal of Mechanical Engineering Education, 34(3), 220-232.

Karnopp, D. C., Margolis, D. L., & Rosenberg, R. C. (2006). System Dynamics: Modeling and Simulation of Mechatronic Systems (4th ed.). Hoboken, NJ: John Wiley & Sons.

Mazzei, A., & Scott R. (2006). Enhancing Student Understanding of Mechanics using Simulation Software. In Proceedings of the Annual ASEE Conference (Session 1466). Chicago, IL: ASEE.

Murray, R. M., Li, Z., & Sastry, S. S. (1994). A Mathematical Introduction to Robotic Manipulation. Boca Raton, FL: CRC Press.

Nordenholz, T. (2006). Animation as the Final Step in the Dynamics Experience. In Proceedings of the Annual ASEE Conference (Session 2468). Chicago, IL: ASEE.

Ogata, K. (1997). Modern Control Engineering (3rd ed.). Upper Saddle River, NJ: Prentice-Hall.

SolidWorks Corporation (2005). SolidWorks® 2006: SolidWorks Essentials; Parts and Assemblies. Document Number: PMT0070-ENG, Concord, MA.

Wong, J. Y. (1993). Theory of Ground Vehicles (2nd ed.). New York, NY: John Wiley & Sons.

MICHAEL P. HENNESSEY is an Associate Professor of ME at the University of St. Thomas, has published 32 technical papers in journals (6), conference proceedings (25), or in magazines (1). He is a member of ASME, SIAM, ASEE and earned a BMATH from the U of MN, an MSME from MIT, and a PHD from the U of MN in ME.