

Using 'Working Model' to Introduce Design into a Freshman Engineering Course

Kurt Gramoll
Georgia Institute of Technology

Introduction

Over the last several years, numerous engineering schools have instituted new design courses for Freshman and Sophomores [1-4] or modified existing courses to include a design component [5-8]. If time and curriculum allows, it is generally best to institute a new Freshman design course or a series of courses that allows the content of the course to be centered on design from the conception of the course. Colorado School of Mines [1] followed this philosophy and initiated a 11-hour design course that included both the Freshman and Sophomore. However, from a practical point of view, it is difficult to add new courses, especially 11 hours worth, since other courses must be cut from the curriculum. Furthermore, serious design projects should wait until the senior year when students have the required knowledge to fully understand and appreciate design projects.

As a compromise, most schools implement a Freshman or Sophomore design as part of an existing introduction engineering or computer courses. This method proved successful at the University of Maryland [2] where the objectives were narrowed to three topics, engineering design, modern graphics, and introduction to computer software. The inclusion of design into either the introduction to engineering course or a computer related course is appropriate since the majority of practicing Engineers do design on a daily basis that involves the use of computers. At Georgia Institute of Technology in the School of Aerospace Engineering, a similar program has been developed over the last 5 years, where engineering design, use of basic computer software, and the application of computers to engineering science is taught in the Freshman course, Introduction of Computers in Aerospace. (Modern engineering graphics is covered in a separate course.)

One of the difficulties in teaching engineering to Freshmen and Sophomores is their lack of math and physics courses. Basic engineering concepts, which are needed for most basic design projects, are difficult to introduce without losing the students. Another constraint, is the time the instructor can devote to a Freshman course. If actual design projects are undertaken in a Freshman design course, a large amount of time will be required to plan and coordinate. This

is includes 'quality' time spent with the students in the lab and in the office brainstorming.

Computer simulation programs, such as Working Model, can help overcome the lack of engineering knowledge of Freshman and the time burden associated with design courses. These program allows students to design mechanical systems their first year and to see them work without first learning all underlying theory. This paper reviews the program Working Model and its use in the Introduction to Computers in Aerospace course that all Freshman are require to take. This paper discusses how the software was integrated into the course and what educational impact (and problems) it had on the students.

Appropriate Use of Simulation Programs in a Freshman Design Course

Before presenting the actual use of the simulation program, Working Model, for a Freshman engineering course, the question of how and when computer based engineering design should be used, needs to be addressed. Graham and Wills at the University of Waterloo [9] investigated the use of various computer programs in engineering education and concluded that computers and simulation software can be used effectively. But they also noted that access to the computers and programs were a major obstacle. Most software they used was non-commercial software developed at various universities. University developed software is generally low cost or free, but is difficult to install and use, and is notorious for bugs. Also, technical support for this type of software is usually lacking. However, on the other hand, commercial software is usually to expensive to install in an undergraduate lab on multiple machines and too difficult to learn how to use for a Freshman course. The study by Graham and Willis was also geared toward Electrical and Computer Science fields. It seems that most good simulation software for engineering education has been traditionally concentrated in those two fields. This has been driven by the industry need to use computer simulation to design circuits. Mechanical Engineering has only just began to use simulation programs in real design oriented problems in industry and good simulation software for education is rare. Most programs for Mechanical

Engineering tends to be topic oriented for a single type of problem, such as animation software for dynamics [10] or beam analysis programs [11]. These program are good for that particular topic, but are not broad based and are hard to integrate into the curriculum at all universities.

Another topic of continued discussion is real labs verses simulation. Uharan and Henry [12] addressed this topic and stressed that real laboratories can be very confining since only very simplistic processes can be modeled due to time (both TA and students') and expense. In addition to those problem, Freshman students lack the required technical understanding to do more then basic physics type of experiments. With simulation programs, real complex systems can be quickly and easily modeled and simulated. The student can learn from the cause-and-effect scenario and do what-if design changes. This type of learning, trial and error, may not be the most efficient at the senior level, but it is very effective at the Freshman level. Crosbie [13] at California State University at Chico, mentioned that "...departments have been impressed with the impact which an effective simulation facility can have on engineering students." Freshman students are sharp and can quickly come to an understanding by doing complex experiments with simulation software.

However, the computers must be balanced with other teaching techniques [9,12]. Through the use of both traditional labs, when possible, and simulation projects, students can receive a broad based understanding of the material. The Kolb Learning Cycle has been widely discussed [14] as a systematic method to organize information for optimum learning and retention. It can be applied to engineering education by looking at the four basic leaning styles, Reflective Observation (watching) verses Active Experimentation (doing) and Concrete Experience (sensing/feeling) and Abstract Conceptualization (thinking). Each of these areas have been described in detail by Harb, et. al [14]. It is interesting to note, however, that paper mentions the use of simulations in three of the four learning areas. Simulation software can be adopted to many different teaching styles.

Along with simulation, Data visualization is also very important [15]. Computer based simulation programs have been available for many years, but it has only been when the graphical and visualization capabilities of the personal computers became advanced, that these programs became useful to engineering education. The main idea behind simulation is to quickly see the effects of different parameters, and this can best be done through the visual senses.

Overview of the Mechanical Simulation Program 'Working Model'

Working Model is a mechanical simulation program that allows the engineer, i.e. the student, to design and test prototype mechanical and structural systems without having to build physical test models. The program combines advanced motion simulation technology with an easy to use graphical model editor to analyze force, motion, energy and

displacement. It is best used to prototype mechanical systems and to run what-if design scenarios. The program will not produce equations or closed form solutions to problems, but will give numerical results, such as force, velocity, acceleration, position, energy, etc., in digital, bar or graph form. The program is built around a dynamic simulation engine that provides a translation of real world Newtonian mechanics to the computer. The user simply draws the mechanical system (see Fig. 1) on the screen with forces and/or other initial conditions, and then the program converts each object into a set of differential equations that describes its motion. The differential equations for all objects are solved numerically by integrating over time. The program checks for interaction between objects and boundary conditions. The results can be numerical values or a graph on the screen.

The overall operation of the program adheres to the Apple interface standards and uses basic Macintosh operations such as clicking, dragging, dialog boxes, tool palettes, and double clicking. There are some special consideration in dealing with mechanical modeling that need special attention to learn and to become familiar with, but requires no more effort than learning a good computer game. Working Model is developed and licensed by Knowledge Revolution, 15 Brush Place, San Francisco, CA 94103, 415-553-8153. A Windows version is scheduled to be released in the Spring of 1994.

When constructing a model, the mouse is used in conjunction with the tool palette (left side of main screen in Fig. 1.) Objects are drawn on the screen using the circle, rectangular, or polygon drawing tool. After drawing the basic object, it can be moved or its size modified by using the editing tools. The other two main tool groups are the joint

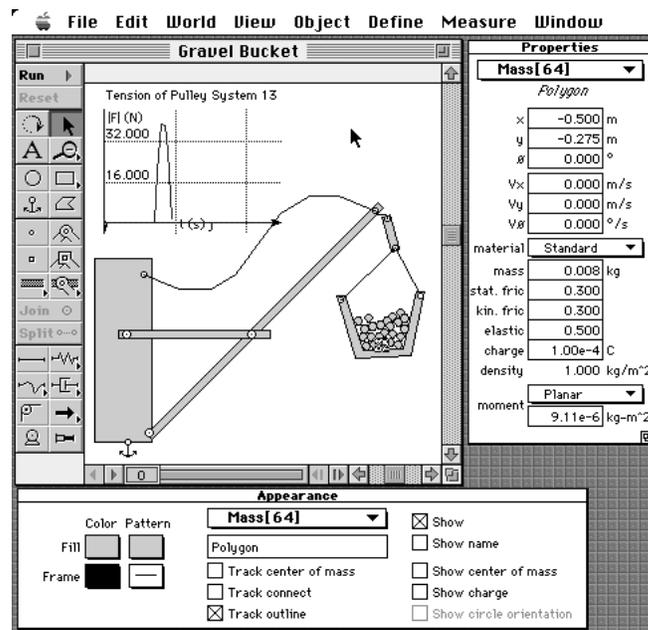


Fig. 1. Working Model Interface.

tools and the constraint tools. The objects can be connect by using pinned or fixed joints, or with slider joints. Slider joints can be placed on other objects or in the background. The constraint tools allow the user to place various types of boundary conditions objects, such as forces, pulleys, actuators, and motors on physical objects. In addition to direct constraints, objects can be connected using ropes, springs, dampers and rods. Most of the construction is done on the drawing screen. However, objects can also be modified through accessing the Properties, Appearance and Geometry windows. Of special interest is the Properties window where the objects' size, initial position and velocity, friction, charge, etc. can be modified.

After the model has been set up and all the initial conditions have been set, the simulation can be run. There are three levels of accuracy, 1st order (Euler), 3rd order (Predictor-Corrector), and 4th order (Runge-Kutta) that are available. However, the only output you will see is the motion of each object on the screen. This may be sufficient but in most cases accurate output such as velocity, forces, position, acceleration, is required and needs to be recorded. This is done by selecting the object that is to be monitored and then use the Measure menu to select which parameter is to be outputted. The output format can be in graph or digital form. An example of this is shown in Fig. 2. The data can also be sent to another program such as Excel and stored while the simulation is taking place. Multiple output graphs and values can be placed on the working screen at any one time.

There are a few limitations, such as all motion is 2-D, objects can not have holes, and there is no polar coordinate output. There are other limitations, but most can be overcome by clever use of the program and how the problem is modeled. Individual parameters, such as spring constants, mass, etc. can be modified through the use of control objects. This allows parameters to be modified while the simulation is taking place and in a sense, allows the user to control the simulation. This is an advanced feature, but a powerful one for real control

simulation. The formula capabilities of Working Model allows the program to be specialized to individual problems. Any property can be modeled by a single, one line formula. However, formulas only work with metric units.

Working Model can also produce QuickTime movies directly. QuickTime is the movie file format that is standard in the Apple operating system and available in the Windows environment. This allows instructors to make a series of example problems and play them back quickly in the classroom, or place them on the network for the students. In this way, the solution is animated and the students can visually see the problem. Working Model also allows the user to write the object position as a data file or as a Macromedia 3D file. From there, the file can be imported into Macromedia 3D and rendered.

Using 'Working Model' in a Freshman Engineering Course.

As part of the basic computers course for Aerospace Engineering, Working Model, along with Excel, Word, TK Solver, MatLab, are introduced to entering Freshmen. The main goal of this class is not only to instruct students on some of the computer tools available to do engineering, but to also teach students basic engineering concepts, both design and engineering science. Working Model has added a further dimension for this course by allowing the students to do complex designs without excessive program writing or macro writing in spreadsheets. For instance, within minutes of using Working Model, the students can visually see multi-degree of freedom vibration systems and then quickly see how vary spring constants or masses will effect the frequency or amplitude.

The course, Introduction to Computers, is only 10 weeks long with four contact hours, 2 lecture hours and 2 lab hours which limits the amount of time to be spent on any one program. Roughly, two weeks are spent with each of the main programs, Excel, Word, Working Model, TK Solver and MatLab. For the design oriented portion of the course, both TK solver and Working Model are used extensively. MatLab is primarily used for teaching engineering science which is used extensively in later control courses in the School of Aerospace Engineering.

For the Working Model program, the first lab was primarily devoted to learning how to operate the program and running verification problems. One such problem involved examining a simple spring-damper-mass vibration system. The students were told to predict the displacement of the mass over time using the three numerical integration methods in Working Model, Euler, Predictor-Corrector and Runge-Kutta and then compare them to the closed form solution. A brief summary of vibrations and the main vibration equation for the closed for solution were given to them. Since they have not yet had a differential equation course, derivation of the closed answer was not expected. The results showed clearly that the Euler method quickly deviates from the correct solution.

In the second lab, a design oriented problem is assigned. A typical design problem was to determine the

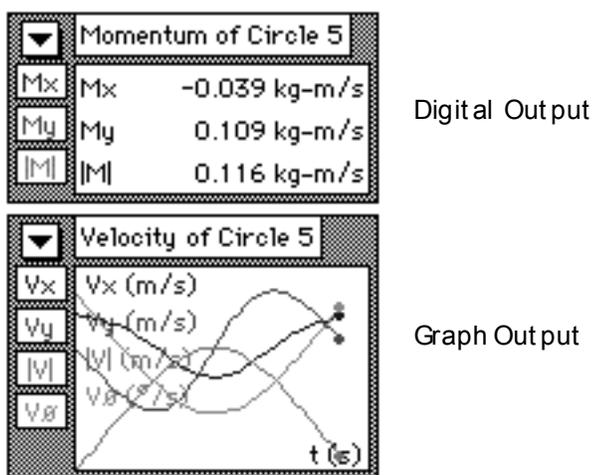


Fig. 2. Output Options

orbital speed and altitude of a satellite required for the satellite to be over a particular spot on the earth. A more detailed design project involving the design of a 'bungie' jumping rope that was assigned to the students is given below. There are four parameters that can be varied and there is more than one good solution, but there is no perfect solution. In other words, the students had to make an engineering judgment to determine which requirements needed to be relaxed to satisfy the other requirements.

Bungie-Jumping Design Problem

The Mile-High Bungie Jumping Company wants you to design a bungie jumping cord that will meet the following specifications.

1. The Jumper can't hit the ground.
2. The Jumper has a mass somewhere between 50 to 100 kg.
2. The hang time (the time a jumper is moving up and down) is greater than 25 seconds but less than 40 seconds.
3. The maximum g's on the jumper is less than 2 for both positive and negative directions.

There are four parameters that can vary, jumper's mass (between 50 and 100 kg), cord's spring constant, K , cord's damping constant, C , and the length of the cord, L . To earn the consulting fee (and your grade), you must recommend a K , C , and L value. You must also supply a working model of the cord for the

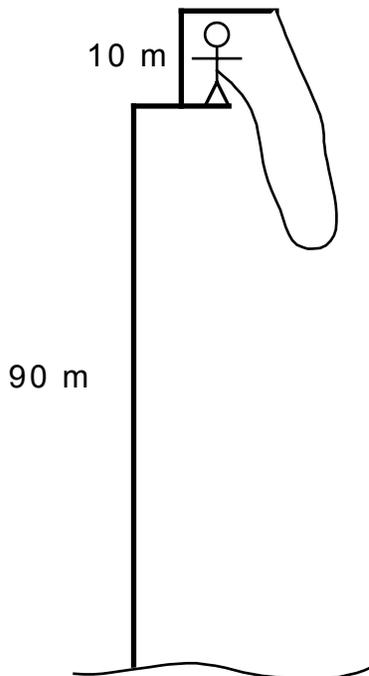


Fig. 3. Bungie-Jumping Problem.

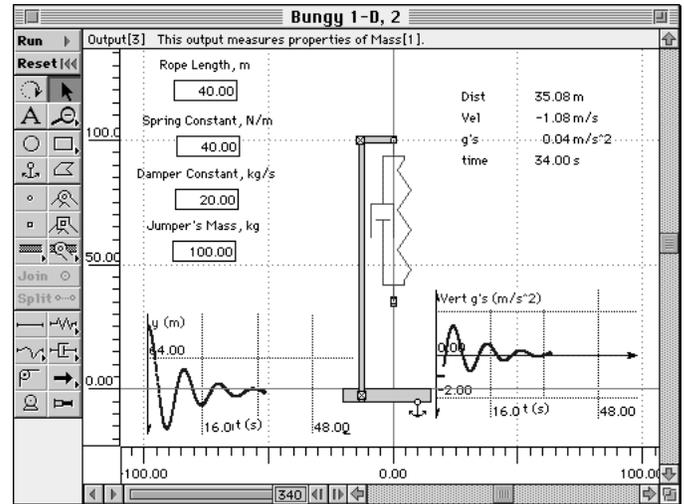


Fig. 4. Bungie Cord Problem.

client to experiment with. Your model should have both a digital and graphical meter for the vertical g's and distance from the ground. Your Working Model should also have 4 input controls, one for the rope length, spring constant, damper constant, and jumper's mass, that can be changed for each run. The overall height of the tower is 100 m, but the jumper jumps at 90 m. The height of the jumper is 2 m.

Other restrictions include, the jumper can only go in the vertical direction and the jumper has no initial acceleration other than gravity. The hang time is defined as the time the jumper is moving up and down with a vertical displacement (maximum to minimum distance for any one cycle) of greater than 4 meters. When the vertical displacement is less than 4 meters, then the ride is over.

The cord can be modeled as a spring and damper, but remember it only works if the cord is stretched. The optimum cord will allow the jumper to experience almost 2 g's (but not more), almost touch the ground (but not hit it) and let the experience last 25 seconds (but not more than 40 sec, your client still has to make a profit). Your grade will be based on maximizing the preceding values but not exceeding them. Exceeding the limits will severely tarnish your reputation as a consultant (and your grade) so be careful and be a good design engineer.

Figure 4 gives an example how the problem can be set up in Working Model. Notice the four input values that allows the user to modify for each analysis. The value can actually be modified during the simulation, but in this problem they were required to be held constant. Figure 5 shows the properties window for the spring-damper element that is used to model the rope. By using the in-line formula programming,

Fig. 5. Cord Material Properties Input.

the K value for the spring the damper can be activated only if the rope is stretched.

The results are presented in both digital and graph form (Fig. 4). Notice that the units are not correct on the graphs because the output is in g's which is not a standard unit. The program does not allow the use of non-standard units or normalized units.

As mentioned previously, there is no single correct answer but many good ones. The students had to decide which of the requirements were most important and then relax the other requirements. This particular problem forced the students to realize some requirements could not be violated for any reason (i.e. not hitting the ground) while the others were more flexible. Some students were upset because the problem did not have a single, perfect answer, but that is one of the most important engineering concepts; Engineers have to make judgment decisions. Many students think that an Engineer is told what to do and then he does it without needing to make any non-engineering decisions.

Due to the lack of time, actual bungi jumping cords were not tested. If time permitted, it would have been interesting to have the students construct their own bungi cords, made to their specifications, and then use them in a real jump, but then we can not afford to lose any of our young Engineers.

Since the course dealt with computers, the final test also included using the computers to solve design problems. The final was a combination of engineering design and science problems that tested the students engineering ability and their use of computers. The final was submitted electronically, as a word processing document, over the network. The students had 3 hours in the computer lab to complete the final. The students were not allowed to take the test home.

In conjunction with the basic Freshman computer engineering course, students are also expected to use Working Model in other basic courses, such as Statics and Dynamics. In particular, Working Model can provide a visual supplement for most homework problems in these courses. This program does not substitute for learning the theory and solving problems by hand, but gives the student an additional tool to visualize a design. It also allows the students to perform more advanced design problems that would otherwise not be possible in the introductory engineering courses.

Educational Benefits in Using Simulation Software

The two biggest advantages of Working Model, or any simulation program, in engineering education is being able to visualize engineering problems and to solve complex problems that are beyond the textbook. The first advantage, problem visualization, can be done with or without the professor using the program. About 50% of the homework problems assigned in basic mechanical/structural courses can be solved by using Working Model. This allows the student to check and 'see' the answer which enhances the students insight and understanding of the concept. The dark side of this is that the student can use the program to get the answer and not understand the problem. This brings up a very important point, *Working Model will not teach basic understanding of the underlying physical laws of engineering.* Theory must be learned from a good textbook and/or a teacher. The program does not output equations or the numerical equivalent of the equations which leaves the student wondering what the machine is doing.

The second advantage, solving advanced engineering problems, needs to be done in conjunction with a formal course. With Working Model, the professor can assign problems that could not normally be solved by undergraduate engineering students. For example, multi-body dynamics, which are normally too complex for basic Dynamics, can be easily solved. The basic theory is the same for either single or multi-body dynamics but the solution time is an order of magnitude different. Most real world engineering problems are multi-body problems, and thus by using Working Model the professor can bring actual engineering problems into the course. This requires the professor to take the time to set up the problem and to incorporate it into the curriculum, which may be the main obstacle in using Working Model in the classroom.

One problem with powerful simulation programs like Working Model is that students will not be able to check their answers. For instance, the gravel bucket problem in Fig. 1 is impossible to predict the tension in the rope due to the movement of the numerous rocks. The program will solve problems that can not be solved by hand, and thus there is no way to check if the answer is correct. It is important that the students understand the basic concepts behind numerical solution methods before using these simulation programs to solve actual engineering problems.

Summary

Working Model, a mechanical simulation program, was successfully implemented into the Freshman engineering computer course as a time and cost effective method to bring design to the Freshmen. Only two weeks were devoted to teaching and using the program but in that time, the students participated in basic design of two mechanical systems. The program was easy to learn and use for design, which allowed its use in the classroom. The two main benefits from using the simulation software was the ability of students to model real engineering problems and to perform advanced mechanical simulation that are not possible with pen and paper. One restriction is of course the cost which ranges from \$100 to \$300 (depends on the number of copies) for educational institution. On the other hand, since it is a commercial program, it is supported and is of high quality. The program is now being integrated into other courses in Aerospace Engineering.

Acknowledgments

The financial support of The Edutech Institute at Georgia Tech and the School of Aerospace Engineering in purchasing the Working Model software are gratefully acknowledged. The assistance of Dr. Jim Craig and Ms. Sandra Pierotti in teaching the course is also acknowledged.

References

1. Olds, B.M., M.J. Pavelich and F.R. Yeatts, "Teaching the Design Process to Freshman and Sophomores," *Engineering Education*, July/August 1990, pp. 554-559.
2. Dally, J.W. and G.M. Zhang, "A Freshman Engineering Design Course," *J. Eng. Ed.*, Vol. 82, No. 2, Apr., 1993, pp. 83-90.
3. Morris, D., "An Experiment in Freshman Design," 1990 ASEE Conf. Poc., Toronto, Canada, June 1990, pp. 839-841.
4. Riffe, W.J. and B.P. Henderson, "A Second Year Mechanical Engineering Design Course," 1990 ASEE Conf. Poc., Toronto, Canada, June 1990, pp. 980-985.
5. Millar, S.B. and T.M. Regan, "A Process Model of Teaching/Learning for Designing First-Year Engineering Design Courses," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 1376-1382.
6. Sathianathan, D., R.S. Engel and R.L. Foster, "A Freshman Engineering Design Course," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 883-887.
7. Biezad, D.J. and R.M. Cummings, "Introducing Aircraft Design During the Freshman Year," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 30-33.
8. McNeill, B.W., D.L. Evans, D.H. Bowers, L. Bellamy and G.C. Beakley, "Beginning Design Education with Freshman," *Engineering Education*, July/August 1990, pp. 548-553.
9. Graham, E.D. and B.L. Wills, "Appropriate Uses of Computers in the Undergraduate Engineering: New Applications and Their Evaluation," 1990 ASEE Conf. Poc., Toronto, Canada, June 1990, pp. 281-285.
10. Jacquot, R.G., D.A. Smith and D.L. Whitman, "Animation Software for Enhancement of Lectures in Engineering Dynamics," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 1529-1531.
11. Gramoll, K.C., "Interactive Beam Analysis Program for Engineering Education," *Computer Application in Engineering Education*, to appear in 1994.
12. Uhran, J.J. and E.W. Henry, "Real Labs vs. Simulation," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 1438-1441.
13. Crosbie, R.E., "Use of Computer Simulation in Engineering Design Courses," 1991 ASEE Conf. Poc., New Orleans, June 1991, pp. 1865-1868
14. Harb, J.N., S.O. Durrant and R.E. Terry, "Use of the Kolb Learning Cycle and the 4MAT System in Engineering Education," *J. Eng. Ed.*, Vol. 82, No. 2, Apr., 1993, pp. 70-77.
15. Higuchi, H. and E.F. Spina, "Improving Undergraduate Engineering Education Through Scientific Visualization," 1993 ASEE Conf. Proc., Urbana-Champaign, June 1993, pp. 1581-1583.

KURT GRAMOLL

Dr. Kurt Gramoll is currently an Assistant Professor in the School of Aerospace Engineering at Georgia Institute of Technology. He graduated from the University of Utah with a BS in Civil and a MS in Mechanical Engineering. Later, after working at Thiokol, Inc., he received his Ph.D. in Engineering Science at Virginia Tech in 1988. Since then he has taught at Memphis State University and worked at the Japanese National Institute for Space and Astronautical Science. His technical research work involves filament winding of composite structures and fluid-structure interaction. He is also heavily involved educational research, particularly in the development and use of multimedia programs in engineering.