MoveIt: FROM REAL LIFE TO MODEL AND BACK AGAIN

Abstract

MoveIt contains structured dynamics and vibrations modules that are designed to imbue in students an enhanced ability to look at real life situations, derive mathematical models, animate their simulations, and then compare the results with the original systems. The goal is to strengthen the linkage between analysis and design/modeling and in so doing strengthen the students’ abilities to function as engineers.

The particular modules have all been class tested in the lead author’s classes and have been modified over several years so as to be challenging and yet not so difficult as to be off-putting. By combining visualization and analysis, students from both camps (visual and written learners) have shown success at tackling the various exercises.

The modules can be used in a variety of class levels, with goals appropriately shaped to reflect the course user groups.

Introduction

The first author has observed over time that students in his class have recently exhibited a tendency to be more focused on analytics and to have a relatively poor skill set with regard to physical intuition. This fact has been widely observed by others and is clearly due to the changing nature of our technological world. Cars that are computer controlled are not ones that lend themselves to “tinkering.” Likewise, the microelectronics that runs through most all modern technological artifacts present our nascent engineers with little of the opportunity for hands-on learning that so typified the pre-college experience of their counterparts in years past.

On the other hand, students are very comfortable with videos and, hopefully, reasonably well oriented toward simulation/animation software. Their inclinations can be used [1], in a properly designed course, to enhance learning [2], [3], [4].

What the authors have tried to do is add a new component to what have traditionally been pure analysis courses as a way of addressing this need, and MoveIt is the result of these efforts. MoveIt is made up of a collection of separate modules, and each module focuses on a particular engineering system which has been chosen to have a direct connection with the students. For example, one of the problems used in the vibrations folder is that of an electric mixer, such as would be found in a typical kitchen. The presumption is that students have some knowledge about the system and the familiarity will reduce any pre-conceived tensions about working on the project. The systems used are also ones that lend themselves to generalization. Thus the problem at hand can be tailored by the instructor to be relatively quick and easy or can be made sufficiently deep as to require some weeks of work. The entire structure is web accessible, an avenue of learning that has seen continual growth [5] that is unlikely to diminish any time soon.

The authors increase student motivation to tackle these modules by letting the students know that key concepts from the modules will find their way into the course quizzes. Thus, in addition
to fulfilling their homework requirements by working on the modules, they're assured that they're increasing their chances for examination points as well. Much as we professors would like students to embrace learning for it’s own sake, personal observation has shown that developing learning approaches with a clearly defined pay-off in the form of points, and therefore grades, has a highly salutary effect on their motivation [6] [7].

Governing equations are available to the instructors of the courses but equations and code are purposefully withheld from the website itself. The entire point of the learning exercise is to derive appropriate equations and write working MATLAB [8] code. The underlying rationale for the approach presented is gone into in depth as a guide to the student in developing his or her own engineering know-how. Personal learning then builds as they springboard from this general guidance into the particulars of the detailed analysis that is ultimately required.

A balance of collaborative and individual work is struck in the lead author’s implementation of the modules by having students work together in small teams as part of their analysis phase but the final code writing, performed after all the analysis and design work has taken place, needs to be done independently. The requirement for all modules is for an operational MATLAB code to be submitted on-line which needs to work unambiguously. Code that is only partially operational or non-operational receives a large point deduction and the final coding has to be by the person doing the submitting. Obviously, individual instructors in other courses are free to impose different work requirements.

Background

Student familiarity with and use of YouTube videos has grown enormously over the last few years and these videos have been embraced by the lead author as part of Targeted Breaks [9] in lecture. These breaks serve as a way of simultaneously amusing/relaxing the students and also allowing some pain-free engineering insights to be introduced “under the radar” as it were. Seeing a free-style skateboarder demonstrate his skills gets the students’ attention and then enables the lead author to bring out whatever salient point seemed most appropriate at the time. Dynamics, stability, friction, and so forth, all flow out of a simple video. From this beginning as a short motivational item, the idea evolved into what eventually became what is presented here as the individual elements of MoveIt.

Another supporting element of MoveIt is the inclusion of simulation and animation within the modeling and analysis part of the projects. For several years the authors’ department has tried to introduce computation into its courses and has grappled with the question of how more physically insightful experiences could be created that wouldn’t require the type of resources that dedicated labs call for. The MoveIt modules have proven to be an effective way of addressing these needs. Without computation there can be no animation, and in order to produce the computational results the student must perforce derive the appropriate analytical equations.

Instructors should note that the included modules are presented as baseline examples that can easily be scaled back (making them more akin to short term homework) or expanded so as to allow a multi-week project. Emphasis can be placed on the simulations and their interpretation or on design. For instance, the caber toss module has been used as a straightforward “model this system” and has also been used as “design a support delivery system that mimics that of the athletic competitor who is tossing the caber - a robotic caber-thrower.” The key is that MoveIt
provides a good example of a project, one which has been fully tested in class, but that certainly can be modified to fit the interests of the instructor/class.

Assessment

The first aspect of assessment that has been addressed has been a simple examination of whether students reported satisfaction (or not) with the program. This has been restricted to end-of-course written commentary and was used to give the authors feedback in terms of which aspects were working (at least as perceived by the users) and which were not. The most significant finding from this feedback has been the confirmation that some of the students have, for a variety of reasons, not gained a comfortable familiarity with programming and simulation and therefore had difficulties with this part of the modules. In response, the authors have set up a remedial set of sessions to ensure that these students receive the needed computational experience. No attempt is made in these sessions to address the full range of shortcomings that may exist due to problems with prior courses. Rather, the particular tools necessary to handle the particular range of problems presented in the modules are gone over so that the mental barrier of “I hate MATLAB” or “I hate plotting” are removed.

Students uniformly expressed an overall positive affective outcome as a result of the modules. Those coming into the course with “MATLABphobia” expressed a much higher comfort level with computation in general as a result of the assignments, and overall the students expressed an increased perceived ability to analyze and solve “real” engineering problems as a result of going through the modules.

Students also reported an increase in their appreciation of the engineering world as a result of their work. Many of the systems around them had not previously struck them as “engineering” and suddenly they realized that EVERYTHING is engineering. An amusing piece of feedback was from the student who wrote “Damn you Professor Tongue! I find that I am now unable to look at life without thinking ‘Hmmm, I wonder what the natural frequency of that is and what that implies about the support stiffness!’” Luckily for me, he/she appended a smiley face, much to my relief. But as I continually assure my students, my job is to infect them with the “understanding system dynamics is fun” virus and my hope is that it proves incurable.

A more in-depth assessment is planned in which the performance of students who experience MoveIt is directly compared to those who do not, keeping everything else in the course essentially the same. This will be enabled by having the same instructor teach different sections of the course, one with and one without MoveIt. The size of the classes that currently exist (150 for dynamics) should allow a large enough cohort in each section so as to provide statistically meaningful results. This evaluation is currently scheduled for Fall 2010.

Content of a typical module

Caber Toss  http://www.bensontongue.net/moveit/index.php/dynamics/caber_toss/

This module looks at the behavior of a rigid rod in two dimensional motion. In traditional texts one might see a line segment leaning against a wall or perhaps pivoting about a pin. In such a form, it’s not really something to inspire much interest.

What this module does is provide some needed context. It opens with a YouTube video of a
Scottish Highland Games competitor throwing a caber. This is somewhat of an arcane sport in which beefy men pick up what’s essentially a telephone pole and toss it. Contrary to one’s immediate impression, there’s actually some skill involved and, at a minimum, a successful throw must cause the caber to undergo a rotation of 270 degrees.

The chosen video provides a fun break in the middle of class. It would first be shown in about the fifth week of an introductory dynamics class and would let the students know that they would soon be expected to deal with such a system.

The following part of the module is “The Set Up,” in which the caber throw of the video is idealized. The simplification to a planar system represents the baseline system. As mentioned earlier, the instructor can further simplify it by allowing the base of the caber to remain in contact with the ground until launched by the application of an impulse. Or she can ask the students to do a more complete model that more closely tracks the actual situation (the bottom end supported by the competitor as he runs forward, a phase during which the caber rotates about the base, and a launch phase). Or it can be made into an even more open ended design case in which the student is asked to determine the forces that need be supplied by a caber launching robot in order to carry out the task.

“The Scope” breaks the basic sequence up into phases so that the students can see how complex problems can be “chunked” into manageable units. This guidance is quite necessary, in general. Without it, most students will flail around, unable to figure out quite how to proceed. After some experience with these modules, however, they quickly learn how to approach new problems in the structured manner here illustrated.

Finally we have “The Simulation.” This presents one approach to a final presentation. Students have to have created a successful mathematical code in order to construct an animation, and requiring this animation does a great job of showing them how good a set of results they’ve achieved. When students are only asked to calculate numbers, they often go far astray but never realize the fact. In fact, for problems like the caber toss that involve translational and rotational motion, it has been the authors’ experience that many students struggle to visualize the entire motion even after plotting kinematic data. Indeed, not all methods of data visualization are created equal, and when the students are tasked with creating an animation, it allows them to synthesize the simulation data into an easily understandable medium that can be used as a check to see if something went wrong with their analysis/simulation.

MoveIt has been designed as a stand-alone resource for use by different instructors in the manner desired by those instructors. It seems appropriate, though, to address the actual structure of the lead author’s course as a point of reference.

All of the lead author’s courses for which MoveIt modules have been utilized exhibit a carefully designed and unified structure. All have been lecture courses with relatively large enrollments, an educational format that has been an interest of the lead author’s for some time.

Homeworks are assigned at roughly one week intervals and the answers to each problem are always made available at the time of the assignment. A fraction of them will cover material that the students will see in the miniquizzes and in the MoveIt assignments and the students are told this fact from the start. They’re also told that the final will reflect what they’ve gone over in homeworks, in miniquizzes and in the MoveIt assignments.
Miniquizzes are single question quizzes that take (depending on the difficulty) between 15 and 25 minutes to work out. They occur, on average, every week and a half. A week before the quiz, the students are told exactly what aspect of the material will be covered. No midterms take place. It is the authors’ opinion that miniquizzes relieve some of the tension associated with testing. Each one is short and worth just a few points. Therefore, failing one isn’t a disaster. This is especially so because the lowest two scores are dropped. The idea is that the students can concentrate on exactly the relevant material and have a sense beforehand that they “get it.” The problems are always directly related to (and sometimes identical to) material that’s already been seen in class and in homework.

The overall goal is to have all aspects of the course show a tight interrelationship that serves to advance student learning. The MoveIt modules present an opportunity to delve deeper into the problem physics and tackle more complex problems than could otherwise be addressed in miniquizzes or homework assignments.

One additional aspect of the class is that every one contains a break [1] midway through the lecture. This is used as an opportunity to get to know the students better (and let them get to know their instructor as well), to show some interesting images/videos to pique their interests and to slip some engineering motivation into their lives without it being obvious “learning.”

A final comment about how the course is run. In addition to the formal structure, the atmosphere is made very student-friendly. The lead author’s goal over the last few years is to show how a very large lecture can nonetheless have the feel of an intimate seminar [2], work that has been pursued with positive effect by others as well [10], [11]. Student feedback over the years has indicated that this, perhaps all other aspects of the course, is what led them to “love it” rather than merely tolerate it.

**Learning Goals:**

As delineated in the MoveIt site itself, the learning goals are to reinforce the student’s awareness of the relationship between real world systems and abstracted models and to raise the level of their self confidence in producing useful simplified models on their own (and thus empower them to apply these engineering tools in other courses and in future employment). The approach to achieving these goals is uniform across all modules. All modules follow the same format - a real system, an abstraction phase, a modeling/simulation phase, and an animation phase. By adhering to a fixed format, the student learns that there is a logic in analysis and design and that following this logical approach allows large problems to be attacked in a systematic manner.

Students are made aware of the learning goals and are presented with a baseline example that “works” in each of the modules. Thus they can see how their animation should behave, both from the initial video and from the sample solution animation. Knowing what they’re working toward allows them to know when they’ve successfully solved the exercise. This approach parallels the one recommended by the lead author for homework exercises. In these, students are always told the final answer. Therefore they know whether or not they have mastered the material. If they can’t match the answer, then they know it’s time to visit the professor or teaching assistant for additional guidance. Nobody is in doubt as to whether they’ve achieved mastery.
Interactivity by design:

The modules are all designed to allow a greater or lesser amount of design freedom and the instructor can decide ahead of time how much of a design exercise is desired. As discussed in the previous section, by increasing the scope of the problem from the baseline one presented, the student is inescapably presented with a wider range of design choices. Narrowing down the focus, on the other hand, means there will be less opportunity for interactive learning but a better chance to emphasize a particular concept.

All choices made by the student are measured against a well defined objective. For instance, in the case of the caber toss, the problem has a well defined target for success (the caber rotates forward after impacting the ground) and any choices made by the student with regard to impulse applied, time of launch, and so forth, are ultimately judged by how well they either meet (or fail to meet) that objective.

Improving Cognition and Conceptualization Skills:

The goal is to instill a step change in the students’ competence in breaking down complex real-world phenomena into understandable abstractions and then be able to apply their knowledge of dynamics and vibrations to formulate appropriate governing equations, simulate these equations (so that they can “do something” with them rather than have them be an end unto themselves), and then animate the simulations to provide a visual confirmation of their physical reasonableness.

Students routinely enter class with a deficiency in one or more aspects of the above. Fear of computation is commonplace, strangely enough. A sense that “real” problems are too difficult to attack is another weakness. For any of these deficiencies, the solution is positive reinforcement that they can approach complex problems and can successfully solve them. By giving them a baseline problem/solution, and supporting them with office hours/discussions if they have specific needs, they find it difficult to not gain mastery and confidence. The mastery is a consequence of simply doing the work. Or, as the well worn phrase puts it: Practice makes perfect.

Structured and multimedia content:

The content of the modules has been carefully chosen to be both engaging, understandable, and relevant to the students’ everyday experiences. Because they’re familiar, the students are not overawed in any way and feel they’re more accessible due to this familiarity. Luckily for pedagogy, highly relevant examples of both dynamics and vibrations abound in the world if one takes but a moment to look for them. The hope, borne out in evaluations, is that students get an extra “kick” from analyzing the behavior of a famous YouTube video - it provides them with a sense of mastery over the world. By working with the modules, they come to a point at which they don’t just see and appreciate the video, they understand it to a level that the usual viewer does not and with that comes an attendant sense of accomplishment as an engineer.

The YouTube videos are quite delineated so that the students can see what the basic dynamics or vibrations problem may be and this delineation is further refined by the supporting text. The baseline assignments don’t say something ambiguous like “analyze this video” but rather make
explicitly clear the need for a simplified model of a particular type and demonstrate what this model would look like.

Engendering student engagement:

The topics of the modules are chosen to be inherently engaging [1]. First and foremost they’re designed to be projects that students can relate to and in which there’s a high level of inherent interest. Students uniformly react positively to the impressive (and slightly silly) sight of a competitor heaving a huge caber and thrill to an X-Games style backflip. Basically, there’s simply something compelling about seeing something moving on screen. By including an animation phase to the modules, this component is inserted into the endgame, as it were. But here it’s the students that create the animation. Experience has shown that many students find this an especially compelling part of the exercise and produce quite impressive, three dimensional, full color animations as part of their final submission. Knowing how the system should behave serves as a hook to keep them working should their animations fail to display the expected behavior.

Software interface design:

All modules present a unified appearance and procedural flow. The use of multiple nested pages was consciously avoided. Each major set of modules falls within its own folder (dynamics and vibrations) and each of these folders contains the individual modules as easily navigated clickable items on a left column.

Popups were avoided in the design of the site and clickable links that draw the user away from the particular module under consideration were also strictly limited.
Conclusion

MoveIt provides a guide to classroom learning that can be used to enhance student appreciation for the power of engineering analysis and design and which works in concert with lectures and quizzes to increase the student’s facility with regard to modeling, simulation, and animation.

It is meant as a supporting element to a class environment and provides guidance to instructors with regard to problems they can immediately utilize and a framework within which they can create their own.
References


