Teaching Rigid Body Mechanics Using Student-Created Virtual Environments

Todd D. Murphey

Electrical and Computer Engineering

University of Colorado

Boulder, CO 80309

`murphey@colorado.edu`

Abstract— **This paper describes a recent effort to use student-created virtual environments to teach rigid body dynamics to electrical and computer engineering senior-level undergraduates and first-year graduate students. Many of these students have no background in dynamics except for their freshman year physics course. The approach described here relies on students creating their own virtual environments, takes roughly four to five weeks of instruction, and therefore fits within the scope of other courses that may require electrical and computer engineering students to be familiar with dynamics. The background required only consists of basic linear algebra and ordinary differential equations. By the end of the pilot class, the majority of students were able to expertly model and animate high degree-of-freedom systems, such as slip-steered vehicles, helicopters, submarines, balancing unicycles, motorcycles, and puppets. A core component of the learning philosophy is that by providing students with a venue for creating their own virtual labs, they are able to promote their own learning. This paper summarizes the basic material presented to students, some of the virtual environments they chose to implement with only minimal guidance, and three instruments used to assess the effectiveness of the class architecture.**

I. INTRODUCTION

The course "ECEN 4028/5028: Robot Dynamics and Motion Planning" was introduced into the curriculum in Spring term, 2006, at the University of Colorado in order to give senior-level undergraduate and first-year graduate students with a predominantly electrical and computer engineering background a solid background in dynamics. Most of these students have previously only had dynamics in their freshman physics sequence, yet many of them had an interest in animation, mechatronics, control, and other applications of multi-body dynamics. The goal was to give these students a sufficient background (in a relatively short amount of time) to correctly compute the dynamics of complex, realistic multi-body systems and to then use these in some area of application. Every student was required to develop, over the course of the term, a virtual environment for animation, dynamics, and motion planning.

The use of virtual laboratories in the classroom has become much more prevalent recently, including teaching physics [8], [19], circuits [6], photonics [7], and instrumentation [12], to name just a few. The advantages of virtual laboratories are great–in particular, they allow for laboratories that can be scaled to many students without requiring many more resources and can be used for remote-learning students [7], [9]. However, these courses are typically aimed at pre-designed laboratory experiences in a manner similar to physical, in-class laboratories. Such approaches largely ignore one of the intrinsic benefits of virtual laboratories–that students can create their own virtual laboratory in a manner that reflects their talents and interests. Moreover, one of the difficulties with virtual laboratories is that they rely on students already knowing enough about the real, physical system that they can successfully transfer that knowledge to their understanding of the virtual system. In such cases, students may or may not already be familiar with the physical counterpart of the virtual laboratory. The basic strategy employed in the class architecture presented here is to create a simple, uniform virtual laboratory that students program themselves, and then allow them to create a more sophisticated virtual laboratory of their own choosing. By allowing them to choose their own virtual laboratory, students are guaranteed to be able to work with a virtual system that they already know something about, or at the very least find compelling. Developing a course that allows students to do this is the focus of the present paper.

One of the difficulties with a senior-level course is that assessment in a quasi-experimental setting is difficult due to a lack of control group. The class described here had only twenty nine students, and therefore called for using multiple modalities of assessment. Assessment was addressed using a combination of three modalities (relevant in a quasi-experimental scenario [15]). First, an in-class survey was used to assess student perception of their learning. Second, they were asked a question that specifically targeted their ability to generalize the concepts used in class to a more complicated situation. Third, students' final projects were assessed along a novice/expert continuum. Notably, the majority of final projects reflect a level of expert knowledge that is well beyond what was directly addressed in the classroom. The author gave relatively little input into projects, and students productively engaged in projects involving helicopters (with remarkably well-constructed system dynamics), submarines, unicycles with moment wheels for stabilization, swinging crane problems, and marionettes. These last two modalities of assessment reflect a recent focus on generalizing from concrete problems to more abstract problems, as detailed by the National Research Council [3], [13].

Lastly, along a more anecdotal line, it is worth noting that

nearly half the class asked about graduate level coursework they could take as electives, such as probability theory, differential geometry, abstract algebra, and advanced linear systems. The fact that this course would motivate students who do not need to take these courses to take them as electives is also a good, if informal, sign of the motivational effectiveness of the class architecture.

This paper is organized as follows. Section II discusses the course content, the canonical example for which the class created a virtual environment, and the students' creation of their own virtual environments. Section III discusses the various modalities of assessment of the course. We end with conclusions and future work (including how we plan to extend this course structure to teaching linear algebra and differential equations) in Section IV.

II. COURSE ORGANIZATION AND CONTENT

Electrical and computer engineering students typically have little to no background in dynamics aside from the standard Newtonian mechanics (e.g., $F = ma$) they learn in an introductory physics course. To address this, the class is organized in such a way that mechanics are presented in a manner that is directly parallel to the creation of a virtual environment in order to make the material more concrete. In particular, the class is organized so that students follow the procedure below:

1) Learn how to define the configuration for a system;
2) Learn how to create a virtual environment for it using rigid body transformations;
3) Use these same transformations to compute the kinetic and potential energies;
4) Compute dynamic equations of motion using the Euler-Lagrange equations;
5) Use these models in motion planning strategies.

This procedure is first implemented using the rolling disk example (discussed shortly), for which all students must implement a virtual environment. After this, students choose a system of interest to them and create a virtual environment for it. (All assignments were turned in using the web-based class management system, Moodle [1], where they also engaged in online discussion forums as in [2].) A difficulty with allowing students to choose the systems for which they will create virtual environments is that the resulting equations of motion are difficult to control because of underactuation, nonlinearities, and the multi-input multi-output nature of most systems. Therefore, rather than focus on control directly, the rest of the course focuses on motion planning. This gives the ability to create global results using only locally valid feedback controllers or even with open-loop controllers.

The course consisted of 25 bi-weekly lectures at an hour and fifteen minutes each. The number of lectures is divided as follows:

[3] Configuration Spaces and Topology;
[4] Vector Fields and Nonlinear Controllability;
[8] Dynamics;

[8] Motion Planning.

Note that only four weeks of the course are dedicated specifically to teaching dynamics, though the discussion of both configuration spaces and vector fields lay the appropriate foundation for both dynamics and motion planning. Hence, roughly half of the course is dedicated to dynamics in one way or another. This indicates that the method presented in this course could easily be transferred to another course where something other than motion planning is the co-emphasis with dynamics.

One of the main challenges in running a course of this sort is to make sure that it does not become a mathematics course. The author took the approach of introducing the minimum amount of material required to understand, at a computational level, the body velocity and its use in computing system dynamics, describing constraints, etcetera. Hence, this class does not address the fundamental principles of mechanics, and instead focuses on getting dynamics right for relatively complex systems. The course is therefore not a replacement for more advanced dynamics courses, and should be thought of as a precursor for such courses.[1] In this course, the author merely comments that an advantage of Lagrangian mechanics is its ability to use generalized coordinates of the student's choosing, and that it is therefore not as limiting as the Euclidean nature of Newtonian mechanics.

A. Computational Tools Required For Both Dynamics and Virtual Environment Development

To learn dynamics in the way described here, one needs only a few very simple tools to describe the mechanics of a system [4], [11]. Moreover, these tools can be immediately used for purposes of creating a virtual environment for the system of interest. Hence, it is worthwhile to emphasize that both animation and rigid-body mechanics are quite *simple* and tightly related to each other. To this end, only the following computational tools are introduced during the course:

1) The configuration q, which uniquely specifies all points in the mechanical system;
2) $g(q)$ the homogeneous representation of a rigid body motion, as it depends on q. It is of the form

$$g(R,p) = \left[\begin{array}{cc} R & p \\ 0 & 1 \end{array} \right]$$

where R is a $n \times n$ rotation matrix and p is a point in $\mathbb{R}^n$ (where $n = 2, 3$);
3) Three particularly useful rotation matrices are the Euler angle matrices R_X, R_Y, R_Z defined as rotation about

---

[1]It is worth noting that the students are somewhat suspicious of the way they have been taught mechanics in any case, so it is not clear that this is a loss. In particular, several students noted that their introductory physics *concepts* did not cover much of what they experience. For instance, they all knew that a bouncing ball does not fit within the scope of $F = ma$, and that the velocity reflection rule they were taught is effectively ad hoc (this is partially because they were not told that reflection rules arise naturally from variational principles rather than Newtonian mechanics).

the coordinate axes, e.g.,

$$R_X(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix};$$

4) V_b, the body velocity, where $\widehat{V_b} = g^{-1}\dot{g}$ and the "hat" operator is defined by

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 & v_x \\ \omega_3 & 0 & -\omega_1 & v_y \\ -\omega_2 & \omega_1 & 0 & v_z \\ 0 & 0 & 0 & 0 \end{bmatrix} = \widehat{\begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}} = \widehat{V_b};$$

5) $KE(q, \dot{q})$, the kinetic energy (which for any component of a multi-body mechanical system can be represented as $\frac{1}{2}V_b^T \mathcal{I} V_b$ with $\mathcal{I}$ constant and diagonal when the rigid body transformation is mapped to the center of mass and the coordinate axes of the body frame are aligned with the principle axes of inertia);

6) $V(q)$, the potential energy, which for most problems is either related to the height of each center of mass (e.g., $V = mgh$, where h can be calculated directly from the associated g) or the distance between two points (e.g., a spring potential);

7) $\omega(q)$, the constraints on the system where $\omega(q)\dot{q} = 0$, which are typically calculated as $v \cdot V_b = 0$, with v a constant co-vector.

8) External forces F_i (which are typically physical inputs to the system).

With this data, one can correctly (to the degree that d' Alhembert's principle is correct) compute the equations of motion for multiple degree of freedom systems. Given a Lagrangian $L = KE(q, \dot{q}) - V(q)$, constraints $\omega(q)$, external forces F_i, the Euler-Lagrange equations are:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = \sum_i F_i + \omega(q)^T \lambda$$

$$\frac{d}{dt}(\omega(q)\dot{q}) = 0$$

The kinetic energy, potential energy, constraints, and forces can be thought of as data types that must be present to solve a given dynamics problem. Dynamics are therefore presented in terms of how these equations represent a strongly typed syntax, much like other computer languages such as C and C++. This helps students see mechanics in a more systematic and general manner (compared to, say, force balance techniques arising from $F = ma$ which typically require the space to be Euclidean and can lead to students confusing Coriolis terms with forces). Moreover, students learn to use the homogeneous representation g (and the associated body velocity V_b) as the fundamental building block of describing any mechanical system. Moreover, the homogeneous representation g of rigid body motion is precisely what students use to create their virtual environments.

B. Creation of Virtual Environments

As just mentioned, the course is organized so that students first learn how to choose configurations for mechanical systems and then create virtual environments for them. The students are told how to make primitive graphics using vertices of polygons. Hence, for each part of a robot, they create a graphic primitive that represents that part as a collection of vertices. With homogeneous representations g of rigid body motions (described in Section II), they simply multiply every vertex by g to compute the set of vertices as they move along a trajectory in configuration space. That is, if one has a set of vertices describing a part of a mechanical system $\{p_i\}$, then $\{g(q(t)) \cdot p_i\}$ represents how the vertices move as the configuration changes with time. This is closely aligned with how three dimensional graphics are created (e.g., using OpenGL), and is how the students create their virtual environments.[2]

It is important to note how crucial virtual environments are to this style of teaching. Students are able to quickly build, from scratch, models of complex mechanical systems and represent them in a virtual environment that they will have intuition about. For high degree-of-freedom systems, this is essential to convincing them that they have indeed mastered the requisite modeling techniques.

C. Canonical Example: The Rolling Disk Virtual Environment
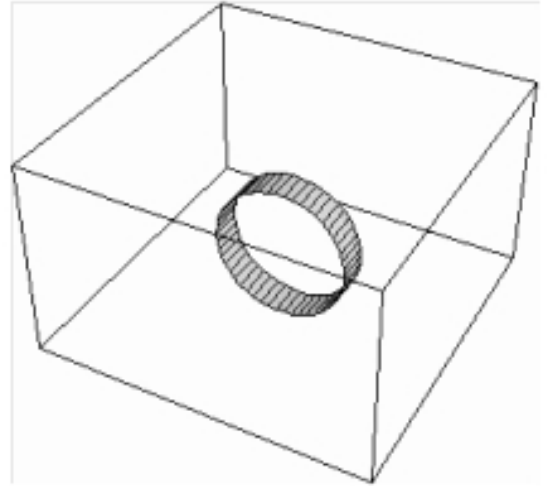

Fig. 1.   The Rolling Disk as a canonical example.

The use of a canonical example to unify the course was essential for continuity. The Rolling Disk, seen in Fig.1, was the canonical example used throughout the course (as

[2]The students did all the programming in *Mathematica* because of its symbolic programming capabilities. In particular, the representation of dynamics in terms of data types is compatible with the style of programming in *Mathematica*, which helps connect the theory to the implementation. It is certainly possible that other symbolic computational tools (such as Maple) could be used with a course such as this, but the author feels it would be nearly impossible to use a package without significant symbolic capability. Hence, MATLAB would be an inappropriate choice for a class such as this.

it has been in many texts, for instance [4], [11]). It has sufficiently interesting dynamics and motion planning issues that it makes an excellent example to develop in class. (Plus, it is simple to draw on a chalkboard.)

Figure 1 shows the Rolling Disk in a *Mathematica* virtual environment. All students in the class create an environment very similar to this roughly midway through the course. With equations of motion and a choice of input forces corresponding to a rolling torque and a steering torque, one can numerically integrate the equations of motion to see the effect of the input forces. In this sense, the Rolling Disk virtual environment is a virtual laboratory where students can investigate dynamics and motion planning.

One can relatively easily compute the forces, constraints, and kinetic energy for this system (see the Appendix for these calculations). With calculations in hand, and the fact that there is no potential energy, we have all the data needed to use the Euler-Lagrange equations, which can be evaluated using a symbolic software package. The question, then, is whether or not, given this introduction to rigid body dynamics, the students are able to successfully generalize the techniques applied to this example to a more broad array of applications. This is the subject of Section III-B and Section III-C.

D. Final Projects: Students Create Their Own Virtual Environments

In this course, students do a final project in which they create a virtual environment for a mechanical system of their choosing. It is acceptable for the students to decide to use the rolling disk in their final project, though only a small number of students actually chose to do so in the pilot course. The general guidelines for this project were that:

1) it must be animated;
2) it must include kinematic constraints;
3) it must include dynamics;
4) it must include motion planning.

Students generalized the rolling disk to quite difficult examples. Of the 29 students, all but five chose to do a project that was nontrivial in some significant way. An overview of the projects they chose is listed below.

1) A swinging crane with a package moving through a factory floor;
2) A dynamic inverted pendulum on a unicycle moving through obstacles;
3) A submarine that has to negotiate narrow corridors;
4) A space shuttle inspection robot that uses a 200,000 node CAD model as the basis for its motion planning scheme;
5) A slip-steered vehicle with correct overconstrained frictional dynamics and feedback control;
6) Motion planning for the snakeboard–an extremely difficult problem [5];
7) One student dynamically modeled and built (as part of another class) a hardware vehicle on which he implemented motion planners;

8) Computational geometry of efficiently computing the effect of polygonal obstacles on a polygonal robot's configuration space;
9) Robot that could solve a maze;
10) Dynamics and motion planning for marionettes;
11) A biped walking through a field of obstacles;
12) Marching band choreography;
13) Motorcycles;
14) Helicopters flying through obstacles;
15) Coordinated control problems, including dynamic reallocation of the desired formation and decentralized dynamic replanning;
16) Exploration robots that need to cover an entire space in limited time;
17) Pursuit/evasion tasks;
18) Only five students chose "standard" projects that were direct applications of the rolling disk example considered in class.

III. ASSESSMENT

Assessment was addressed using three separate modalities. The traditional course survey was used to find out how the students perceived their learning. In this course survey, students were additionally asked to describe modeling a spacecraft in orbit, a task requiring substantial generalization from what they had learned in class. The third modality of assessment was to assess their final projects based on the expert/novice scale (see [3], [13]) and the degree to which they substantially generalized what they had learned in class. Experts and knowledgeable students have extensive, well-structured, declarative knowledge; this is not the case with novices. Furthermore, experts and knowledgeable students possess nuanced understanding of how to approach complex tasks, both familiar and novel; again, novices have no such knowledge (again, see [3], [13]). These last two modalities reflect the recent trend in encouraging learning in terms of going from concrete problems to more abstract problems and going from specific problems to more general ones, as detailed by the National Research Council [3], [13], [15].

A. Student Response

Students were asked to fill out a traditional survey, where they ranked items related to how much they perceived they had learned, as well as other aspects of the class. These questions, along with student response, are in Table I. The main thing that is worth noting about their responses is that students almost uniformly believed they had learned more in this class than they had in an "average" course, and similarly almost uniformly believed that the course use of a concrete, unifying example and the homeworks were helpful. In particular, the majority of students felt that they can use the material in a practical setting.

B. Generalization to Satellites

Students were also asked the following question:

In a few sentences, how would you use the techniques developed in this class to model a satellite in orbit and make it point at a particular target.
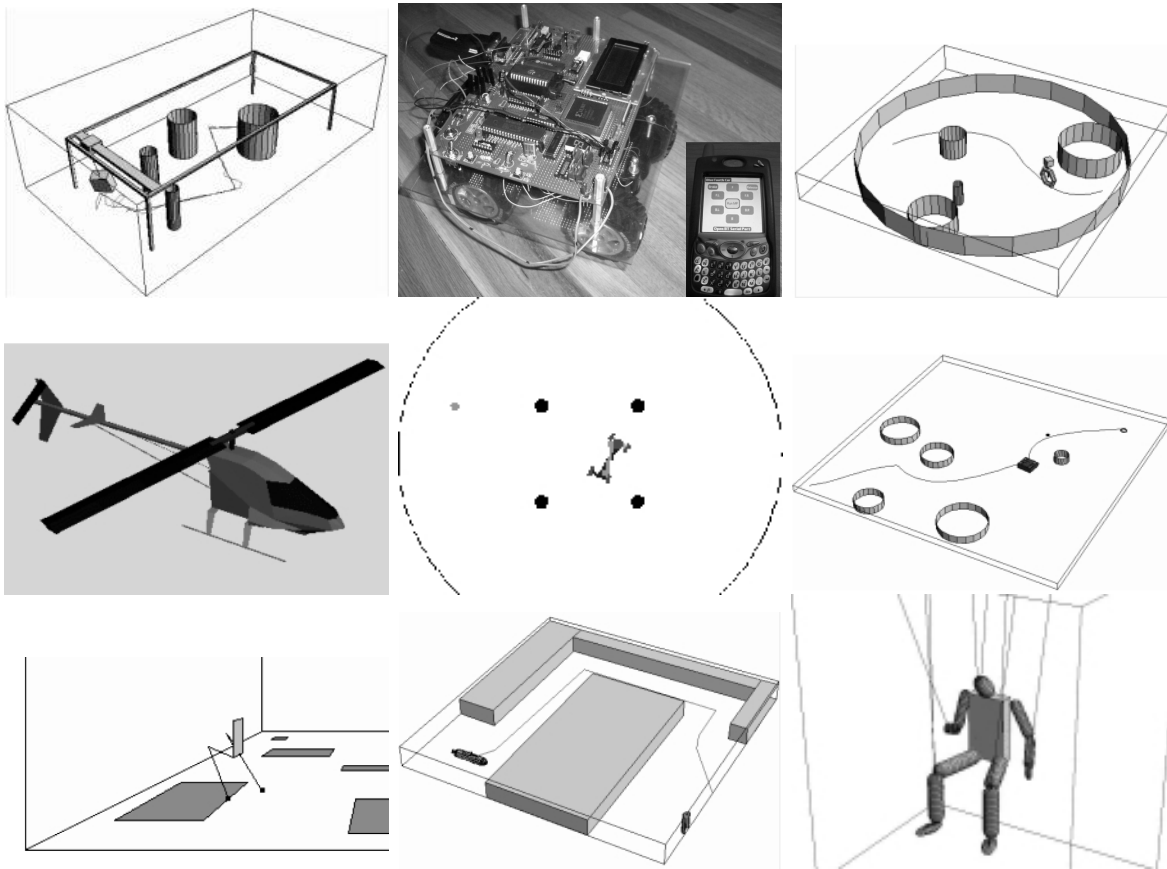
Fig. 2. Final projects initiated by students included (from left to right, top to bottom) a) solving a swinging crane problem (a standard canonical problem in controls), b) creating a hardware vehicle along with a simulation of the vehicle for motion planning, c) an unstable unicycle with moment wheel for stabilization, d) a dynamic model of a helicopter navigating through obstacles, e) modeling of and motion planning for the snakeboard, f) a slip-steered vehicle with all of its constraints properly taken into account, g) a kinematic model of a biped walking through a field of obstacles, h) a submarine, and i) a marionette. These and other projects can be found at *http://robotics.colorado.edu/~murphey/ecen4028/finalprojects/*.

	Question	much less	a little less	about the same	more	much more
1	I have learned _____ in this class than the average class I have taken	0	2	9	10	5
		Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
2	I like the fact that we used just a few concrete examples throughout the course	0	2	2	9	13
3	I would recommend this course to other students	0	0	3	4	19
4	I feel I can use the material we are learning in class in practical situations	0	1	3	13	10
5	Homeworks were helpful in understanding the material	1	0	5	12	8

TABLE I

TABLE OF STUDENT RESPONSES. STUDENTS WERE ASKED HOW MUCH THEY PERCEIVED THEY HAD LEARNED AND ASKED HOW MUCH THEY AGREED WITH SOME STATEMENTS ABOUT THE EFFECTIVENESS OF THE COURSE.

This is a standard, difficult problem that generally requires significant expertise to solve. A correct answer would be that the students would model the rigid body components of the satellite using the Euler-Lagrange equations (which consists of the techniques already discussed), would generate a motion plan to get close to the desired pointing angle, and then would use a linear feedback controller to stabilize to the desired pointing angle. Of the 26 respondents to the survey, only four did not correctly identify the basic approach to solving this problem. Ten of the students were able to give significant detail, including specifics of modeling and what motion planner they would use. The fact that 22 out of 26 students were able to so effectively generalize what they had learned in class is one of the best indications of the success of this course architecture. Section III-C discusses the final component of assessment, which is also along the lines of addressing generalization.

As an aside, it is interesting to note that two of the students who were not able to address this problem were aerospace students who had already taken classes on satellite dynamics. They both commented that they had seen this before and immediately attempted to write down equations they had seen in those prior courses. This, of course, is exactly the opposite of what is desired–it would seem that in this instance their prior coursework was actually acting as an obstacle to generalization.

C. Novice/Expert Assessment of Final Projects

As mentioned in Section II-D, students were asked to do a final project instead of taking a written final. This was designed into the curriculum for purposes of assessment. In particular, the author was interested in assessing the degree to which the students could *generalize* what they had learned and the extent to which they had obtained a more *expert* level of understanding of mechanics and motion planning. Both of these types of assessment have become more common recently, as documented in [3], [13] as well in [14], [16]–[18].

The breadth and depth of these projects in Section II-D is the most convincing evidence that allowing students to create their own virtual environments provides a venue for students to drive their own learning to a great extent. Indeed, 24/29 students engaged in significant, non-trivial projects that required a knowledge of dynamics obviously much closer to "expert" than to "novice."

This type of assessment is more difficult than the assessment in Section III-B because the students choose the sample system instead of the assessor choosing the sample system. Indeed, in the previous mode of assessment, it is already known that a satellite represents a significant generalization from the rolling disk–many universities offer entire courses solely dedicated to satellite dynamics. In assessing the degree to which students generalized the rolling disk in their final projects, six discrete modes of generalization were singled out. These modes are:

1) Addition of internal dynamics or potential energy;
2) More than two inputs;
3) Higher degree of freedom;
4) Feedback control;
5) Multiple, separated, mechanical systems;
6) Mechanics with state transitions.

These six modes of generalization are certainly not unique, but represent reasonable directions of generalization that may, among other things, be required of students when they work in industry and generally require an expert's ability to model. Each project was then (subjectively) evaluated as to whether or not it had significantly generalized what had been discussed in class. Table II summarizes these findings.

Number of Modes Generalized	Number of Students
0	5
1	5
2	3
3	7
4	6
5	3
6	0

TABLE II

Summary of student project generalization by number of modes generalized

On average, students generalized along 2.45 modes of generalization. Moreover, 24 out of 29 students generalized along at least one mode of generalization. Lastly, some of the projects clearly generalized class material (such as the hardware robot), though not along the lines of these specific modes. Hence, this assessment of generalization is somewhat conservative.

It is also worth noting that, as part of their projects, various students learned (on their own or by bringing it up with the author) about differential flatness [4], overconstrained systems [10], fluid mechanics with potential flow, implicit integration of differential equations, as well as many other nominally off-topic areas.

D. Sample Student Comments

Students were encouraged to write comments regarding the class. A sample of these comments follows.

1) "I liked the emphasis on the end results (animations) rather than the approach. It allowed for many different correct answers and allows the student to come closer to fully understanding the solution. The programming afforded me a level of understanding that pure[ly] mathematical homework could not have."
2) "This class stressed me out a lot, but the way [the professor] ran the class alleviated a lot of it. I struggled a lot with the homeworks, but sincerely felt like I understood a lot after completing them."
3) "I think this was a great class and should be offered again in the fall. Although I don't expect to do much work in controls, this class has provided me with a lot of insight into how to practically model a system and design reasonable animations for that system. I haven't had any experience with Euler [angle] matrices

or a true understanding of how to apply Euler-Lagrange equations to describe dynamics."

4) "Moodle was a very useful tool to see other people's ideas. I really enjoyed it. ... This project really made me realize how much I know and how much I lack. It is actually quite scary yet very motivating to think about future classes to come. I had a great semester."

5) "This class is the best class I have taken since starting at the university."

IV. CONCLUSION

This paper presented a basic framework for introducing dynamics to students with little to no background in mechanics using student-created virtual environments. The framework requires students to develop their own virtual environment in both homeworks (using the Rolling Disk and some other canonical examples) and in a final project. The fact that the students create their own environments helps them become more invested in the work as well as helps to make the class feasible for many students in the absence of hardware. All of the virtual environments were made in *Mathematica*, which is widely available and used, and is relatively inexpensive for students ($45 for six months).

A weakness of the approach is that it does not address the underlying theoretical aspects of mechanics in this setting. If the course were purely dedicated to mechanics, there would be more than enough time to cover this. Moreover, as our assessment indicates, this course was successful in helping students become competent at rigid body dynamics.

The eventual goal for this approach is to use this framework of encouraging students to create their own virtual laboratories for helping students learn and retain linear algebra and differential equations. The mechanical systems described here are particularly concrete, and none of the techniques employed are beyond a freshman student with a background in calculus. The hope is that we can create a freshman-level course that has as its other half an introduction to differential equations and linear algebra that would then motivate the courses students typically take later (such as linear systems).

REFERENCES

[1] http://www.moodle.org.
[2] S. Bermejo. Cooperative electronic learning in virtual laboratories through forums. *IEEE Transactions on Education*, 48(1):140–149, Feb 2005.
[3] J.D. Bransford, A.L. Brown, and R.R. Cocking, editors. *How People Learn: Brain, Mind, Experience, and School*. National Academy Press, Washington, D.C., 2000.
[4] F. Bullo and A.D. Lewis. *Geometric Control of Mechanical Systems*. Number 49 in Texts in Applied Mathematics. Springer-Verlag, 2004.
[5] F. Bullo, A.D. Lewis, and K.M. Lynch. Controllable kinematic reductions for mechanical systems: Concepts, computational tools, and examples. *Int Symp on Math. Theory of Networks and Systems (MTNS)*, August 2002.
[6] B.P. Butz, M. Duarte, and S.M. Miller. An intelligent tutoring system for circuit analysis. *IEEE Transactions on Education*, 49(2):216–223, May 2006.
[7] G.W. Chang, Z.M. Yeh, H.M. Chang, and S.Y. Pan. Teaching photonics laboratory using remote-control web technologies. *IEEE Transactions on Education*, 48(4):642–651, Nov 2005.
[8] N.D. Finkelstein, W.K. Adams, C.J. Keller, P. B. Kohl, K. K Perkins, N.S Podolefsky, S. Reid, and R. LeMaster. When learning about the real world is better done virtually: a study of substituting computer simulations for laboratory equipment. *Physical Review - PER*, 2005.
[9] T. Kikuchi, S. Fukuda, A. Fukuzaki, K. Nagaoka, K. Tanaka, T. Kenjo, and D.A. Harris. DVTS-based remote laboratory across the pacific over the gigabit network. *IEEE Transactions on Education*, 47(1):26–32, Feb 2004.
[10] T.D. Murphey. Motion planning for kinematically overconstrained vehicles using feedback primitives. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 2006.
[11] R.M. Murray, Z. Li, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
[12] J.M.G. Palop and J.M.A. Teruel. Virtual work bench for electronic instrumentation teaching. *IEEE Transactions on Education*, 43(1):15–18, Feb 2000.
[13] J.W. Pellegrino, N. Chudowsky, and R. Glaser, editors. *Knowing What Students Know: The Science and Design of Educational Assessment*. National Academy Press, Washington, D.C., 2001.
[14] E. Seymour. Student Assessment of Learning Gains (SALG) http://www.wcer.wisc.edu/salgains/instructor/SALGains.asp.
[15] W.R. Shadish, T.D. Cook, and D.T. Campbell. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Houghton-Mifflin, Boston, 2002.
[16] R. J. Shavelson, M. A. Ruiz-Primo, and E. W. Wiley. Windows into the mind. *Higher Education*, 49:413–430, 2005.
[17] R.J. Shavelson, G.P. Baxter, and J. Pine. Performance assessments: Political rhetoric and measurement reality. *Educational Researcher*, 21(4):22–27, 1992.
[18] R.J. Shavelson and M.A. Ruiz-Primo. *Assessing science understanding (J. Mintzes, J. H. Wandersee, & J. D. Novak Eds.)*, chapter On the psychometrics of assessing science understanding, pages 304–341. Educational Psychology Series. Academic Press, San Diego, 1999.
[19] C. E. Wieman and K. K. Perkins. Transforming physics education. *Physics Today*, page 36, Nov 2005.

APPENDIX

It is worth describing some of the calculations mentioned in Section II-A as they apply to the rolling disk in Fig. 1. A standard choice of configuration is $q = (x, y, \theta, \phi)$, where x and y are the planar coordinates of the point of the center of mass, θ is the orientation of the disk relative to the planar x-axis, and ϕ is the rolling angle of the disk. Let $R_Z(\theta)$ and $R_Y(\phi)$ be the Z and Y Euler angle rotation matrices, respectively. With these, the homogeneous representation of the disk's configuration is

$$g(x, y, \theta, \phi) = g(I, (x, y, 0)).g(R_Z(\theta), 0).g(R_Y(\phi)).$$

The animation used to represent this system is shown in Fig.1, where 40 vertices are used to approximate the rolling disk with polygonal sides. With mass m, width w, and radius r, a homogeneous cylinder has a body-fixed inertia tensor of

$$\mathcal{I} = diag(m, m, m, \frac{1}{12}mw^2 + \frac{1}{4}mr^2, mr^2, \frac{1}{12}mw^2 + \frac{1}{4}mr^2).$$

Moreover, its body velocity (again, where $\widehat{V_b} = g^{-1}\dot{g}$) is

$$V_b = \begin{bmatrix} \cos(\phi)\,(\cos(\theta)\dot{x} + \sin(\theta)\dot{y}) \\ \cos(\theta)\dot{y} - \sin(\theta)\dot{x} \\ \sin(\phi)\,(\cos(\theta)\dot{x} + \sin(\theta)\dot{y}) \\ -\sin(\phi)\dot{\theta} \\ \dot{\phi} \\ \cos(\phi)\dot{\theta} \end{bmatrix}.$$

Its kinetic energy is therefore

$$\frac{1}{2}V_b^T \mathcal{I} V_b = \frac{1}{24}m(12\dot{x}^2 + 12\dot{y}^2 + (w^2 + 3r^2)\dot{\theta}^2 + 6R^2\dot{\phi}^2).$$

Its nonslip constraints can be expressed as the body velocity of the point of contact being zero, which means we must only include the transformation to the center of mass of the wheel. The body velocity for this frame is

$$V_b = \begin{bmatrix} \cos(\theta)\dot{x} + \sin(\theta)\dot{y} \\ -\sin(\theta)\dot{x} + \cos(\theta)\dot{y} \\ 0 \\ 0 \\ 0 \\ \dot{\theta} \end{bmatrix}.$$

This represents the velocity of the frame attached to the axis of ϕ rotation. The constraints are that the forward velocity (the first component–$[1\ 0\ 0\ 0\ 0\ 0] \cdot V_b$) must be equal to the rolling velocity and that the sideways velocity (the second component–$[0\ 1\ 0\ 0\ 0\ 0] \cdot V_b$) must be equal to zero. That is:

$$\begin{aligned} \cos(\theta)\dot{x} + \sin(\theta)\dot{y} &= R\dot{\phi} \\ -\sin(\theta)\dot{x} + \cos(\theta)\dot{y} &= 0. \end{aligned}$$

Therefore, the constraints are of the form

$$\omega(q)\dot{q} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 & -R \\ -\sin(\theta) & \cos(\theta) & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = 0$$

Lastly the input forces are the torque about ϕ (the drive torque) and the torque about θ (the steering torque). Hence, $F_1 = \{0, 0, \tau_\theta, 0\}^T$ and $F_2 = \{0, 0, 0, \tau_\phi\}^T$.