AMERICAN JOURNAL of PHYSICS

Teaching computational physics as a laboratory sequence

Ross L. Spencer

Citation: American Journal of Physics **73**, 151 (2005); doi: 10.1119/1.1842751
View online: http://dx.doi.org/10.1119/1.1842751
View Table of Contents: http://scitation.aip.org/content/aapt/journal/ajp/73/2?ver=pdfcov
Published by the American Association of Physics Teachers

Teaching computational physics as a laboratory sequence

Ross L. Spencer
Department of Physics and Astronomy, Brigham Young University, Provo, Utah 84602

(Received 13 November 2003; accepted 8 November 2004)

In spite of the difficulty of adding new courses to an already full undergraduate physics curriculum, the education of undergraduate physics students would be greatly enhanced by learning computational methods. The standard method of addressing this need is to offer a computational physics course. We have chosen to use the standard three credit hours allotted to computational physics by offering three separate one-credit laboratories, one for sophomores, one for juniors, and one for seniors. Students are introduced to symbolic methods using MAPLE when they are sophomores, and to numerical methods using MATLAB beginning in their junior year. This introduction helps prepare students for their upper division courses, for the research they will do for their senior projects, and spreads computational methods throughout the undergraduate curriculum.
[DOI: 10.1119/1.1842751]

I. INTRODUCTION

Physics, like other structured disciplines, has a very full curriculum because students need to learn challenging material dating back at least to the 1700's. Much of the older ideas and techniques are still important and cannot simply be replaced by the most recent advances. So even though computational methods have been important in physics for more than 30 years, the physics departments that offer them (about half in the United States, according to an Internet search of a wide sample of departments) only offer one course on numerical methods. A particularly well-developed example of such a single course using a laboratory setting is the course developed over a span of nearly 20 years by Gould and Tobochnik.[1,2] Two notable exceptions are the interdisciplinary computational physics bachelor's degree at Oregon State University, in which computational methods are employed throughout the curriculum,[3,4] and the CCLI Project at Lawrence University, which integrates computation with undergraduate coursework.[5–7]

At Brigham Young University we began teaching computational physics by offering it late in the curriculum when the students' experience allowed advanced problems to be tackled. We offered a three-credit course in numerical methods at the senior/first-year graduate level with an emphasis on grid methods for solving partial differential equations, using the excellent book by Garcia[8] which taught us, among other things, the value of MATLAB as an introductory programming language. By teaching this course we learned that it came so late for our undergraduates that little connection between these methods and their other course work and their research projects was possible. Also our lecture/homework style created much frustration, because many of our students have poor programming skills when they start our program. Students studying this subject need help from the instructor to be able to develop good programming style and debugging skills.

To reduce the second problem we started using a laboratory format so that while the students were working at computers doing their homework, the instructor and several teaching assistants were present to help them with debugging and to answer questions. (A survey of computational courses shows that other departments also use this solution.) In this format the instructor gives a short introduction to the subject for the day, after which students go to work. Occasional mini-lectures are given during the lab period to enliven the course, to anticipate difficulties, and to provide breadth. We were pleased to find that teaching this subject as a lab creates an active learning environment in which students are exposed to problems, help each other solve them much of the time, and are ready to listen when the instructor has something to say.[9]

This use of an active learning environment worked so well that we decided to abandon our three-credit lecture course and put in its place three one-credit lab courses: Physics 230, 330, and 430. (The usual semester load of a physics major is 14–16 credit hours.) Physics 230 is designed for sophomore physics majors and teaches them basic computational skills (with an emphasis on symbolic methods) using MAPLE. In the process the students review the mathematical methods they learned in their first year and apply them to physics problems from their first and second year courses. The textbook for the course is a MAPLE worksheet.[10] A similar electronic tutorial in MATHEMATICA is included in Ref. 11, and a printed MATHEMATICA tutorial with similar physics examples is in Ref. 12.

Physics 330 is designed for juniors and focuses on ordinary differential equations with applications mostly from classical mechanics, especially nonlinear dynamics. The students start by learning to solve differential equations in MAPLE, using the same handout as in Physics 230. While they are enhancing their MAPLE skills, they also are learning the basics of MATLAB, culminating in learning how to use its differential equation solvers. In the process the students review classical mechanics and are introduced to some of the basic ideas of nonlinear dynamics. Other textbooks that cover the same material as Physics 330, but not in a laboratory format, are given in Refs. 11, 13–22.

Physics 430 is designed for seniors. The emphasis is on methods for partial differential equations and on grid methods using MATLAB. These methods are illustrated with examples involving wave motion, diffusion, Schrödinger's equation, and two-dimensional electrostatics. Other textbooks which cover the same material as Physics 430 include Refs. 11, 13–17, 20, 21, 23–25.

These courses are relatively new and are still being refined, but our students give them good marks when they evaluate the courses. The students are especially pleased with Physics 230, because of the power it gives them early in their studies to solve difficult problems in their other physics courses. In fact, most of our students now use MAPLE as their standard mathematical handbook. When students become involved in research (now a graduation requirement in our department), they often use MAPLE and MATLAB. We have not as yet, however, performed any systematic study of student outcomes to assess the quality of these courses.

To compare the approach described here with what is being done elsewhere, see the extensive listings of books on computational physics given in Ref. 26.

II. BRIEF COURSE DESCRIPTIONS

The goal of the MAPLE worksheet[10] is to help students develop the ability to solve difficult problems and to review the mathematics of their first two years of study. The worksheet is awkward to use in printed form, and is designed to be read by someone sitting in front of a computer.

For example, when the students learn to integrate in MAPLE they are given problems involving integration over a known charge distribution to find the electrostatic potential and the electric field. When the students learn to use MAPLE to do sums, they review the basic ideas of wave interference, including two-slit interference and beats, using MAPLE's powerful graphic capabilities. There is also a long exercise on wave packets in which the students produce animations of packets in dispersive media, illustrating the ideas of phase and group velocity. Students also plot Bessel and Legendre functions and are introduced to the idea of orthogonal functions and Fourier analysis as they learn to make plots and perform integrations.

The environment consists of a room filled with computers and staffed by an instructor and one or more teaching assistants. The lab period lasts for three hours. Typically each student sits in front of their own computer, but we encourage them to work as partners to learn more effectively. The instructor and the teaching assistants roam the room answering questions and helping the students overcome programming and conceptual errors. When the students discuss what they are doing with a partner, read the text aloud to each other, and call the instructor or an assistant over for help, the classroom becomes an active learning environment in which doing difficult mathematics for three hours is much more fun. Students do not print their results because printing wastes paper and slows them down. Instead, when they finish an assigned problem, the instructor or a teaching assistant sits with them for a few minutes listening to them explain what they did, asking questions, and correcting errors or misconceptions. During the semester three examinations are given and the students also take a final examination, often in oral format.

In addition to helping the students handle difficult problems, Physics 230 is also an excellent preparation for our usual mathematical methods course on partial differential equations. Because our students already know how to use MAPLE, the mathematical methods course uses it extensively. However Physics 230 is not a mathematical methods course. The emphasis is on learning to write and debug the MAPLE code, and the mathematics and physics ideas in the course are mostly a review of material previously covered in other courses.

There are three handouts for Physics 330: the same handout used for Physics 230, a MATLAB tutorial and a brief reference manual, and a laboratory manual which gives the students the assigned tasks for each lab period and explains some of the physics ideas that will be encountered. The classroom environment is the same as for Physics 230, except that the instructor lectures a little more because the physical and mathematical content of the course is more advanced. The emphasis is on classical dynamics, but the goal of the course is for students to learn to write and debug the code, not to master the intricacies of dynamics. We assume that the students either have already taken or are currently enrolled in our junior-level classical mechanics course, so the physics topics in each lab are either a review of material already covered or a supplement to it.

Those who have taught this course have found that it is especially important for the instructor to carefully monitor the progress of the students during the first four labs to help students not to become discouraged. Students who have no programming background struggle at the beginning and many just give up unless they receive special attention. If it becomes clear that some students are not going to be able to finish a laboratory, the instructor often helps the students catch up by doing part of a laboratory as a class project. Help from the instructor and from teaching assistants (almost exclusively alumni of the course) is essential for all of the students in the course to have a successful experience.

The text for Physics 430 is a lab manual that contains both physical explanations and descriptions of computing techniques. The emphasis is on finite difference methods using uniform grids. All programming is done in MATLAB, and the same active learning environment is used. Because extensive programming is required in this course, several MATLAB scripts are provided to the students for them to use as templates. Key parts often are left blank so that the student can program the most important parts of the algorithms, but details like building grids and making plots are supplied. This course is a companion to electrodynamics, quantum mechanics, and thermal physics. The unifying idea in this course is the power of linear algebra to effect numerical solutions of difficult problems. This idea turns out to be a challenge to most of our students because linear algebra seems rather remote and abstract when they see it for the first time. But it comes alive for them when they use it to find the vibration frequencies of a hanging chain and the quantum bound state energies in potential wells for which no analytic solution is available. They then use linear algebra to develop implicit algorithms for the diffusion and time-dependent Schrödinger equations, culminating in an animation of a wave packet tunneling through a barrier and an animation of solitons forming and ''colliding'' with each other. Because of the more advanced nature of the problems in this course, the instructor usually requires the students to have read through the assigned laboratory before class, an activity that is encouraged by giving a short quiz on the reading at the beginning of each class period.

A drawback of this lab format is that it is difficult to fully cover the theory behind the algorithms that are presented. Students who are interested in a more complete treatment are encouraged to take a course on numerical analysis. Another concern is that Physics 430 focuses exclusively on finite difference methods using uniform grids, and ignores finite element methods, nonuniform grids, Monte Carlo methods, and particle simulation methods. There also is the limitation that programming in the MATLAB environment is different from programming in a compiled language. However, our experience is that MATLAB is an excellent preparation for learning a compiled language.

III. CONCLUSION

Computational physics is taught as a sequence of three one-credit labs instead of as a lecture course. We find that this format allows us to introduce computational methods earlier in the curriculum and to give a broader overview of computational physics. It also allows computational methods to be emphasized over a period of three semesters instead of just one, which helps the students to become more proficient at programming. The students use the symbolic computational skills they learn in these courses to do their homework in our traditional physics courses, and they use both symbolic and numerical methods in their research, gaining a set of skills that will help them succeed after they graduate.

We do not regard this lab sequence as a substitute for a rigorous course in numerical analysis. Students interested in developing a true expertise in computational methods are encouraged to take one or more traditional courses in this field. But we find that these labs allow all of our students to develop basic skills in this important area.

ACKNOWLEDGMENTS

The author thanks Branton Campbell, Grant Hart, Bryan Peterson, S. Neil Rasband, and Jean-Francois Van Huele for helpful comments and suggestions in the development of these courses.

[1]H. Gould, ‘‘Computational physics and the undergraduate curriculum,’’ Comput. Phys. Commun. **127**, 6–10 (2000).

[2]J. Tobochnik and H. Gould, ‘‘Teaching computational physics to undergraduates,’’ in *Ann. Rev. Compu. Phys. IX*, edited by D. Stauffer (World Scientific, Singapore, 2001), p. 275.

[3]Information on the computational physics program at Oregon State University can be found at ⟨http://www.physics.orst.edu/CPUG/⟩.

[4]R. H. Landau, H. Kowallik, and M. J. Perez, ‘‘Web-enhanced undergraduate course and book,’’ Comput. Phys. **12**, 240–247 (1998).

[5]D. Cook, ‘‘Computers in the Lawrence Physics curriculum. Part I,’’ Comput. Phys. **11**, 240–245 (1997).

[6]D. Cook, ‘‘Computers in the Lawrence Physics curriculum. Part II,’’ Comput. Phys. **11**, 331–335 (1997).

[7]D. Cook, ‘‘Computation and problem solving in undergraduate physics,’’ ⟨http://lawrence.edu/dept/physics/ccli/⟩.

[8]A. L. Garcia, *Numerical Methods for Physics*, 2nd ed. (Prentice–Hall, Englewood Cliffs, NJ, 2000).

[9]C. C. Bonwell and J. A. Eison, ‘‘Active learning: Creating excitement in the classroom,’’ ASHE-ERIC Higher Education Report No. 1, George Washington University, 1991.

[10]See ⟨http://webs.byu.edu/courses/⟩ for more information on the course material, which is freely available.

[11]D. Dubin, *Numerical and Analytical Methods for Scientists and Engineers Using* MATHEMATICA (Wiley-Interscience, 2003).

[12]M. L. De Jong, MATHEMATICA *for Calculus-Based Physics* (Benjamin Cummings, 1999).

[13]G. Baumann, MATHEMATICA *in Theoretical Physics* (Springer-Verlag/Telos, Santa Clara, CA, 1996).

[14]F. Cap, *Mathematical Methods in Physics and Engineering with* MATHEMATICA (Chapman and Hall/CRC, London, 2003).

[15]P. L. DeVries, *A First Course in Computational Physics* (Wiley, New York, 1994).

[16]R. N. Enns and G. C. McGuire, *Nonlinear Physics with* MATHEMATICA *for Scientists and Engineers* (Birkhauser, Boston, 2001).

[17]R. Gass, MATHEMATICA *for Scientists and Engineers* (Prentice–Hall, Englewood Cliffs, NJ, 1998).

[18]H. Gould and J. Tobochnik, *Introduction to Computer Simulation Methods* (Addison-Wesley, New York, 1996).

[19]R. L. Greene, *Classical Mechanics with Maple* (Springer-Verlag, New York, 1995).

[20]T. Pang, *An Introduction to Computational Physics* (Cambridge University Press, Cambridge, 1997).

[21]F. J. Vesely, *Computational Physics: An Introduction*, 2nd ed. (Kluwer Academic/Plenum, New York, 2002).

[22]R. L. Zimmerman and F. I. Olness, MATHEMATICA *for Physicists* (Addison-Wesley, Boston, 2002).

[23]N. J. Giordano, *Computational Physics* (Prentice–Hall, Englewood Cliffs, NJ, 1997).

[24]R. H. Landau and M. J. Paez, *Computational Physics, Problem Solving with Computers* (Wiley, New York, 1997).

[25]S. M. Wong, *Computational Methods in Physics and Engineering* (World Scientific, 1997).

[26]⟨http://sip.clarku.edu/books/⟩.