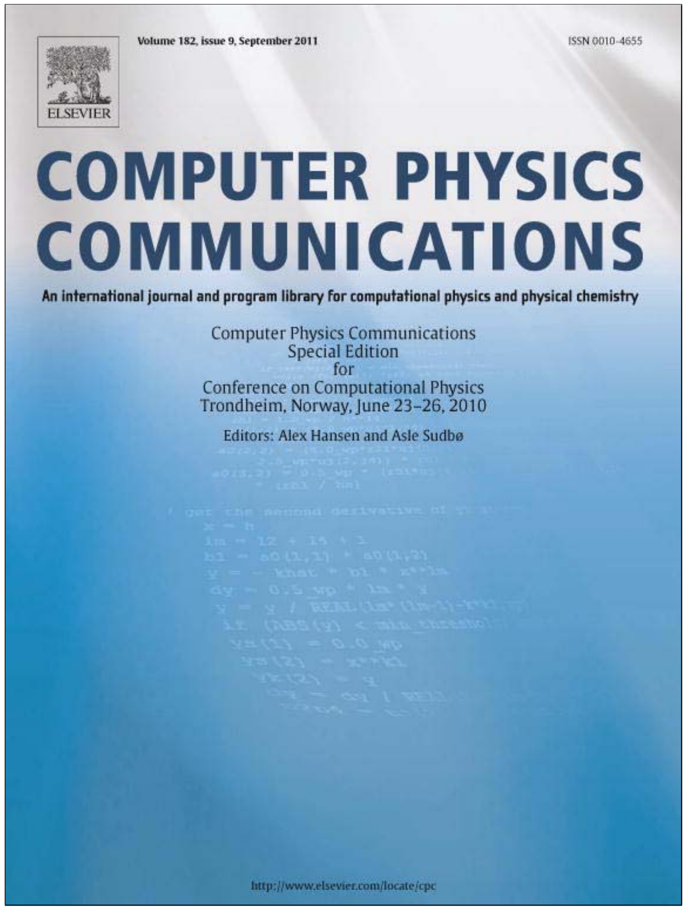

Volume 182, issue 9, September 2011
ISSN 0010-4655
ELSEVIER
COMPUTER PHYSICS COMMUNICATIONS
An international journal and program library for computational physics and physical chemistry
Computer Physics Communications
Special Edition
for
Conference on Computational Physics
Trondheim, Norway, June 23–26, 2010
Editors: Alex Hansen and Asle Sudbø
http://www.elsevier.com/locate/cpc

Computer Physics Communications 182 (2011) 2071–2075

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

# Making physics education more relevant and accessible via computation and eTextBooks

Rubin H. Landau [a,*], Manuel J. Paez [b], Cristian Bordeianu [c], Sally Haerer [a]

[a] *Department of Physics, Oregon State University, United States*
[b] *Department of Physics, University of Antioquia, Colombia*
[c] *Department of Physics, University of Bucharest, Romania*

ARTICLE INFO

*Article history:*
Received 5 July 2010
Accepted 4 November 2010
Available online 13 November 2010

*Keywords:*
Computation
Education
Computational physics
Problem solving
Content
Digital book
eBook
Video lectures

ABSTRACT

Various aspects of computational physics education are discussed, including the need for it, its content and various efforts at providing it. Also described is a new eTextBook that incorporates video lecture modules, source and executeable codes, multimedia enhancements and extensive linkages. The first draft is in pdf and can be "read" with a variety of devices.



## 1. Introduction, what is CP education?

This paper examines five aspects of computational physics (CP) education:

1. What is CP education?
2. What is the need for CP education?
3. Elements of a CP education.
4. How best to provide it?
5. A new CP eTextBook.

I view CP as a multidisciplinary subject that combines physics, computer science and applied mathematics for the purpose of scientific problem solving:

$$\text{CP} = \text{Physics} + \text{Math} + \text{Computation}.$$

We prefer teaching all three subjects in one or several integrated, projects-based course. In contrast, although there is nothing wrong with using computers to help teach physics, it is *not* the same as using computers to help teach computational physics. Furthermore, while the theory of CP may be appropriate for graduate education, for undergraduates we recommend a "hands-on" project approach that leads to active engagement and stimulation of the students. Because teaching the three subjects in context is an effective and efficient approach to education, time not spent teaching physics is thereby compensated for by the increased interests of the students who want to learn the materials in order to solve "their" problem.

## 2. The need to change physics education

The rapid increase in the power and pervasiveness of computers in this last decade have led to a historically rapid change in how science is done and in what types of science are being done. This is the basis of our premise that physics education should change in order to reflect the fact that computation is an integral part of physics, and more than just a delivery tool to assist education. Just as physics departments throughout the world teach their own "Math Methods of Physics" courses because math is too important to leave just the math department, so too *computation has become too important to leave its teaching to just CS Departments*!

As we see it, physics students will benefit greatly from having the common computational toolset and the ability to think about science computationally. The physics education community needs to decide if they want physics to be taught as a static subject, like Classical Greek (fascinating but over with), or a living one that changes as times change. In fact, a recent AAPT–AAPS joint task

* Corresponding author.
*E-mail address:* rubin@science.oregonstate.edu (R.H. Landau).
*URL:* http://physics.oregonstate.edu/~rubin (R.H. Landau).

0010-4655/$ – see front matter 
doi:10.1016/j.cpc.2010.11.006

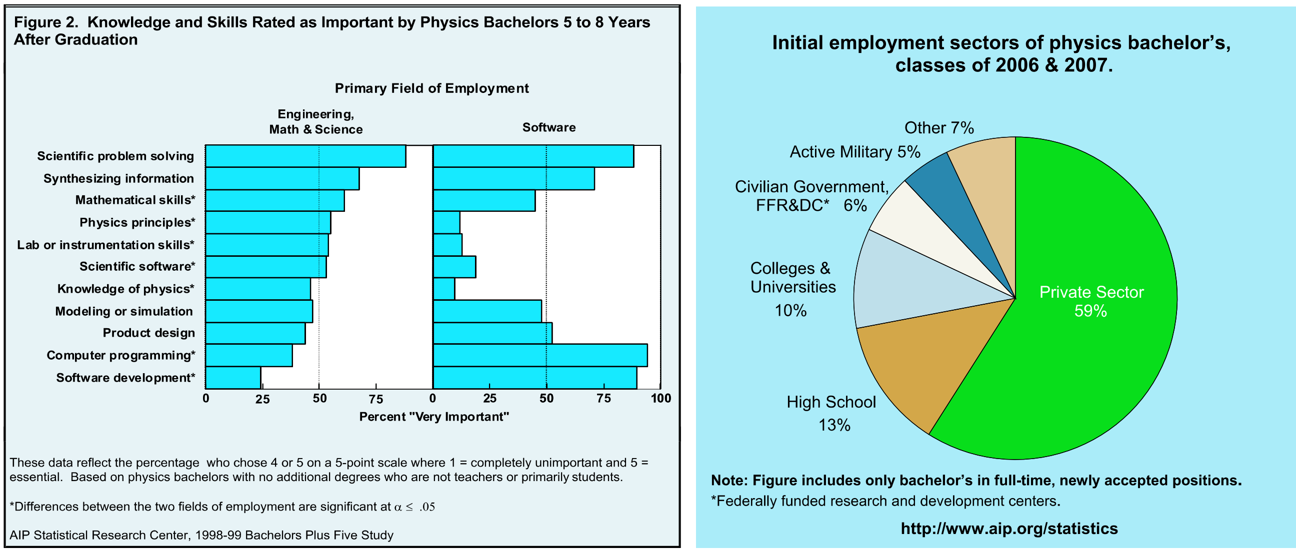



**Fig. 1.** *Left*: Results of an AIP survey of physics majors five years after graduation that asks what skills they use in their present employment. *Right*: Results of an AIP survey that charts where physics graduates get employed.

force on graduate education [1] has concluded that "we are teaching the same things we taught 50 years ago".

My own personal observation is that physics educators, in their quest to make physics learnable to a larger number of students, have been narrowing the physics being taught, but explaining it more. Simultaneously, I observe computational science exploring subjects such as fluid dynamics, molecular dynamics, nonlinear dynamics, and stochastic processes, all subjects too often absent from the undergraduate education. In contrast, modern research in subjects such as astrophysics, complex material and quantum chromodynamics often require multiscale and multiphysics teams, which is quite different from a narrow physics education that aims to have first-time students master mathematical and physical concepts as do experienced teachers. I see a continuing value in a physics education that enhances the ability to attack a broad range of problems using basic principles, and now with computation.

Even though the preceding views do not seem to be finding much traction in the physics education community, there is ample evidence for these views, of which we will give just a bit here. In Fig. 1 left we see some of the results of a survey done by the AIP that asked physics graduates five years after graduation "what aspects of your education is important in your careers?" [4]. The two colors correspond to those graduates with careers involving science, math or engineering (lighter bars) and those who have moved into careers involving software (darker bars). We note that for both groups the most important skills are basic problem solving and synthesizing information. Yet for both groups, detailed knowledge of physics is essentially the least important part of their education! Further data from the AIP [3] (Fig. 1 right) indicate that most physics graduates end up in the private sector and that very few of those end up doing physics per se. So while we professor types like to imagine that we are educating our replacements, only ∼13% of our students will end up doing physics.

I believe the conclusion to be drawn from the AIP data such as Fig. 1 is not that an education in physics fails the students, but rather that it prepares them well for a variety of careers that employ some of the basic skills they have learned in that education. However, trying to tech too much physics in that education, or trying to have the students master physics and math concepts as have their professors, actually leads to a weaker education for what they will face in the real world.

## 3. The contents of a CP education

In Fig. 2 we present a concept diagram of the skills needed for basic scientific computing, and in Fig. 3 we present concept diagram containing the topics that we suggest as constituting a CP education [2]. These figures were taken from our eTextBook [6], where the different elements in the figures are links to actual contents.

## 4. How to teach CP

I dare not be so bold as to suggest that there is one best way to teach CP. However, I would suggest that while using computers as a tool to help teach physics classes is probably a good thing, it is does not substitute for teaching CP. Likewise, while having students run computational simulations in classes is probably better than having just the instructor use them for demos, this also is not teaching computational physics. We need to give students a solid foundation in computation and then build CP and computational thinking upon that. While it is probably too much to expect this to be accomplished in one stand-alone course, it is at least a good start.

Beyond a CP course, there are 5–6 undergraduate degree programs in CP in the US, and about 25 minors, concentrations, tracks, options *et cetera* that try to provide a balanced education using some new and existing courses. I believe that the best general approach for physics would be a new physics curriculum in which computational education and practise is present throughout the curriculum. To this end, Steve Gottlieb and I are editors of a new series of textbooks that we hope will replace some of the standard texts with ones incorporating modern computational thinking into the subject.

## 5. eTextBook

Sitting at a computer in a trial-and-error mode is an excellent locale for learning CP. Likewise, multimedia web technologies permit enhancements to paper texts that enrich the learning experience. For the last fifteen years our CP group has been developing

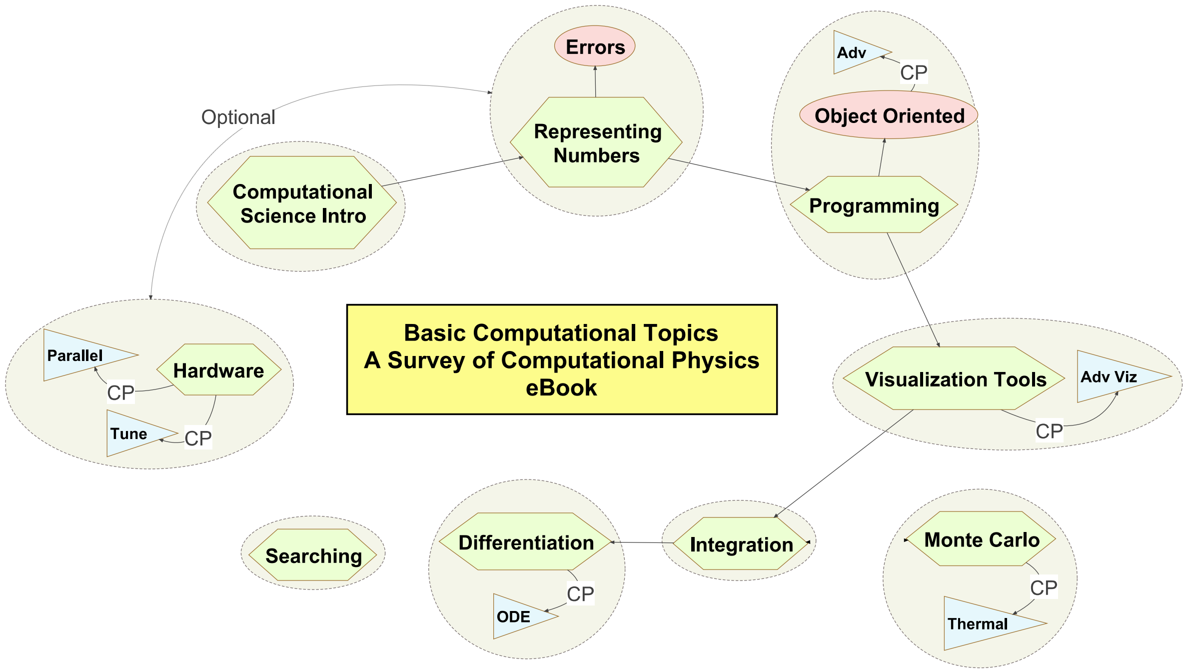



Fig. 2. The basic computational topics needed as background to education in CP.

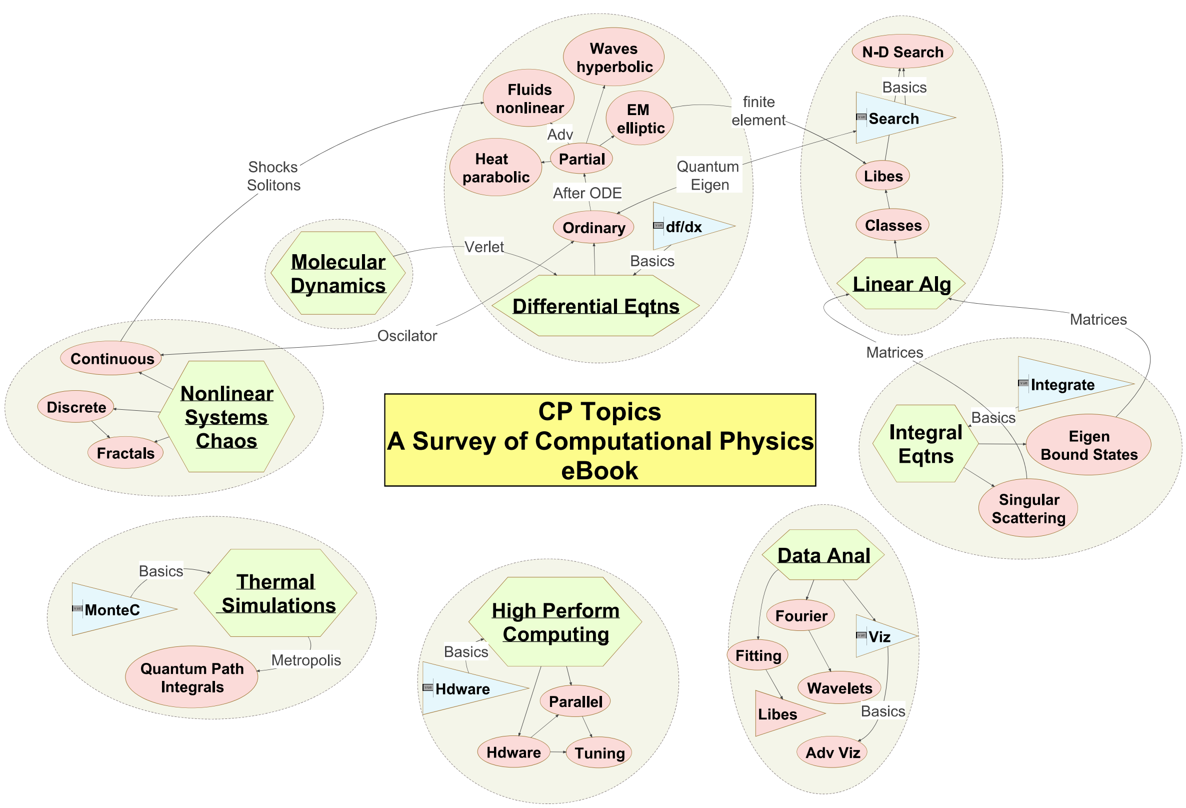
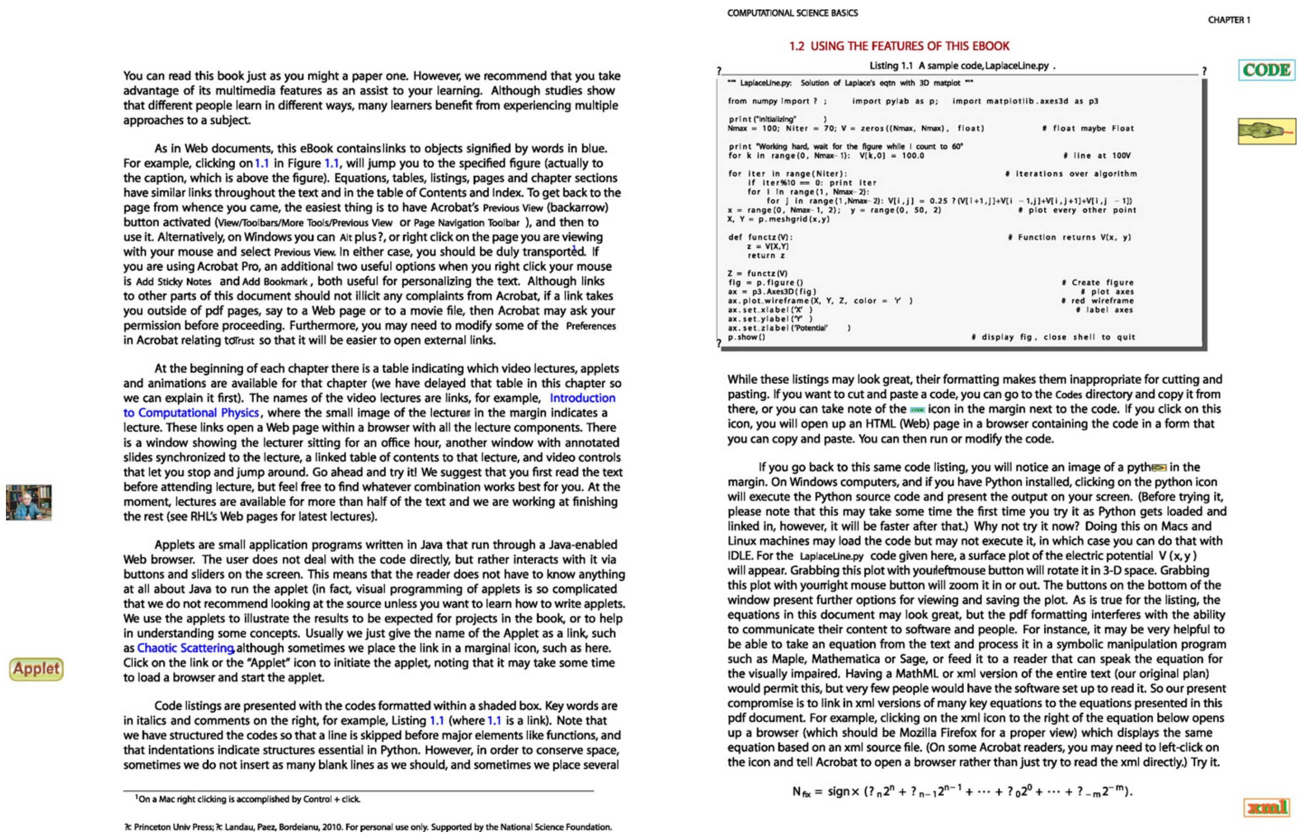



Fig. 3. A display of the major topics to be covered in a CP education.

web enhancements for the courses we teach and the books we have written [2], while for the last few years we have been converting our lectures and texts into electronic forms with the goal of encouraging wider adoption of computation into sciences. The lectures are organized into slide-plus-talking-head modules on individual topics that can be blended into a variety of courses (we use them as lectures for an online course).

In addition, we have created the Python version of our Survey text [2] as an eTextBook [5,6] that will be available on Compadre [7] as well as a commercial product. Our eBook integrates the video-slide modules, text materials, interactive programs, multimedia enhancements into a pdf file (Fig. 4) that can be read and interacted with on a wide range of computers, tablet PCs and eReaders. (Although we initially envisioned the eBook as being written in MathML, the absence of devices able to read MathML, led us to change to linked pdf. However, we have linked various equations in the text to MathML files so that these equations can be ported into programs such as Maple and Mathematica that can manipulate them dynamically.)

In Fig. 4 we show a snapshot of two pages from our eBook that explain some of its unusual features. As on a web page, text objects in blue are hyperlinks whose clicking opens up a page somewhere else in the text. (Not to worry, the Adobe Reader, and especially the Pro version, has multiple navigation tools, as well as highlighting,



You can read this book just as you might a paper one. However, we recommend that you take advantage of its multimedia features as an assist to your learning. Although studies show that different people learn in different ways, many learners benefit from experiencing multiple approaches to a subject.

As in Web documents, this eBook contains links to objects signified by words in blue. For example, clicking on 1.1 in Figure 1.1, will jump you to the specified figure (actually to the caption, which is above the figure). Equations, tables, listings, pages and chapter sections have similar links throughout the text and in the table of Contents and Index. To get back to the page from whence you came, the easiest thing is to have Acrobat's Previous View (backarrow) button activated (View/Toolbars/More Tools/Previous View or Page Navigation Toolbar ), and then to use it. Alternatively, on Windows you can Alt plus ?, or right click on the page you are viewing with your mouse and select Previous View. In either case, you should be duly transported.[1] If you are using Acrobat Pro, an additional two useful options when you right click your mouse is Add Sticky Notes and Add Bookmark, both useful for personalizing the text. Although links to other parts of this document should not illicit any complaints from Acrobat, if a link takes you outside of pdf pages, say to a Web page or to a movie file, then Acrobat may ask your permission before proceeding. Furthermore, you may need to modify some of the Preferences in Acrobat relating to Trust so that it will be easier to open external links.

At the beginning of each chapter there is a table indicating which video lectures, applets and animations are available for that chapter (we have delayed that table in this chapter so we can explain it first). The names of the video lectures are links, for example, Introduction to Computational Physics, where the small image of the lecturer in the margin indicates a lecture. These links open a Web page within a browser with all the lecture components. There is a window showing the lecturer sitting for an office hour, another window with annotated slides synchronized to the lecture, a linked table of contents to that lecture, and video controls that let you stop and jump around. Go ahead and try it! We suggest that you first read the text before attending lecture, but feel free to find whatever combination works best for you. At the moment, lectures are available for more than half of the text and we are working at finishing the rest (see RHL's Web pages for latest lectures).

Applets are small application programs written in Java that run through a Java-enabled Web browser. The user does not deal with the code directly, but rather interacts with it via buttons and sliders on the screen. This means that the reader does not have to know anything at all about Java to run the applet (in fact, visual programming of applets is so complicated that we do not recommend looking at the source unless you want to learn how to write applets. We use the applets to illustrate the results to be expected for projects in the book, or to help in understanding some concepts. Usually we just give the name of the Applet as a link, such as Chaotic Scattering, although sometimes we place the link in a marginal icon, such as here. Click on the link or the "Applet" icon to initiate the applet, noting that it may take some time to load a browser and start the applet.

Applet

Code listings are presented with the codes formatted within a shaded box. Key words are in italics and comments on the right, for example, Listing 1.1 (where 1.1 is a link). Note that we have structured the codes so that a line is skipped before major elements like functions, and that indentations indicate structures essential in Python. However, in order to conserve space, sometimes we do not insert as many blank lines as we should, and sometimes we place several

[1] On a Mac right clicking is accomplished by Control + click.

?c Princeton Univ Press; ?c Landau, Paez, Bordeianu, 2010. For personal use only. Supported by the National Science Foundation.

COMPUTATIONAL SCIENCE BASICS CHAPTER 1

1.2 USING THE FEATURES OF THIS EBOOK

Listing 1.1 A sample code, LaplaceLine.py .

CODE

```
""" LaplaceLine.py: Solution of Laplace's eqtn with 3D matplot """

from numpy import ? ;      import pylab as p;   import matplotlib.axes3d as p3

print ("Initializing"      )
Nmax = 100; Niter = 70; V = zeros((Nmax, Nmax), float)         # float maybe Float

print "Working hard, wait for the figure while I count to 60"
for k in range(0, Nmax-1):  V[k,0] = 100.0                     # line at 100V

for iter in range(Niter):                                      # iterations over algorithm
    if iter%10 == 0: print iter
    for i in range(1, Nmax-2):
        for j in range(1,Nmax-2): V[i,j] = 0.25 ?(V[i+1,j]+V[i -1,j]+V[i,j+1]+V[i,j -1])
x = range(0, Nmax-1, 2);  y = range(0, 50, 2)                 # plot every other point
X, Y = p.meshgrid(x,y)

def functz(V):                                                 # Function returns V(x, y)
    z = V[X,Y]
    return z

Z = functz(V)
fig = p.figure()                                               # Create figure
ax = p3.Axes3D(fig)                                            # plot axes
ax.plot_wireframe(X, Y, Z, color = 'r' )                       # red wireframe
ax.set_xlabel('X' )                                            # label axes
ax.set_ylabel('Y' )
ax.set_zlabel('Potential'      )
p.show()                                       # display fig, close shell to quit
```

While these listings may look great, their formatting makes them inappropriate for cutting and pasting. If you want to cut and paste a code, you can go to the Codes directory and copy it from there, or you can take note of the icon in the margin next to the code. If you click on this icon, you will open up an HTML (Web) page in a browser containing the code in a form that you can copy and paste. You can then run or modify the code.

If you go back to this same code listing, you will notice an image of a python in the margin. On Windows computers, and if you have Python installed, clicking on the python icon will execute the Python source code and present the output on your screen. (Before trying it, please note that this may take some time the first time you try it as Python gets loaded and linked in, however, it will be faster after that.) Why not try it now? Doing this on Macs and Linux machines may load the code but may not execute it, in which case you can do that with IDLE. For the LaplaceLine.py code given here, a surface plot of the electric potential $V(x, y)$ will appear. Grabbing this plot with your left mouse button will rotate it in 3-D space. Grabbing this plot with your right mouse button will zoom it in or out. The buttons on the bottom of the window present further options for viewing and saving the plot. As is true for the listing, the equations in this document may look great, but the pdf formatting interferes with the ability to communicate their content to software and people. For instance, it may be very helpful to be able to take an equation from the text and process it in a symbolic manipulation program such as Maple, Mathematica or Sage, or feed it to a reader that can speak the equation for the visually impaired. Having a MathML or xml version of the entire text (our original plan) would permit this, but very few people would have the software set up to read it. So our present compromise is to link in xml versions of many key equations to the equations presented in this pdf document. For example, clicking on the xml icon to the right of the equation below opens up a browser (which should be Mozilla Firefox for a proper view) which displays the same equation based on an xml source file. (On some Acrobat readers, you may need to left-click on the icon and tell Acrobat to open a browser rather than just try to read the xml directly.) Try it.

$$N_{fix} = \text{sign} \times (?_n 2^n + ?_{n-1} 2^{n-1} + \cdots + ?_0 2^0 + \cdots + ?_{-m} 2^{-m}).$$

xml

**Fig. 4.** Two pages from the CP eTextBook illustrating some of its hypermedia features.

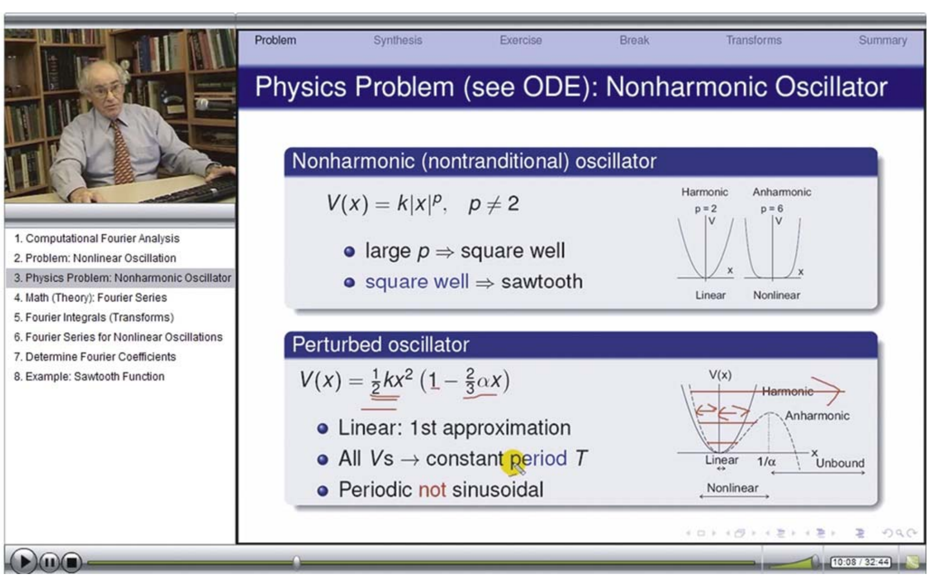



**Fig. 5.**

writing and editing tools that let students or teachers write in and customize the text.)

The icons in the margin denote some of the other features of the text. The speaking head ikon indicates a link to a lecture module. Clicking on it opens a video module, a snapshot of which is shown in Fig. 5, where you should note that in addition to the talking head, there synchronous slides, a dynamic table of contents, and sometimes links to programs and other resources. (However, since these web-based video modules employ Abode Flash, they do not yet run on ipods/ipads, and so we have a version using mpv/mp4 files that will run on ipads.) The snake icon is meant to remind you of a python, and clicking on it will execute the Python code listed in the box. If you want to modify the code, clicking on the CODE icon opens a Web page in a browser from which the user can copy and paste the code (pasting the pdf version of the code may not preserve the spacing, which confuses the Python compiler). The Applet icon in the margin indicates a related applet, and clicking on it will open a browser within which the applet runs. You will also notice on the second page that there is an xml icon. Clicking on this opens a web page containing an xml version of the related equation that can be ported over into programs.

## 6. Take home lessons

Computing has become an essential ingredient in most forefront areas of research in physics and other sciences. It is important that we rejuvenate Physics Education with Research (PER) by bringing real-world research problems into the classroom, and by having students become familiar with multiscale and multiphysics problems that are solved by multidisciplinary teams. Learning physics plus computation plus math in a problem-solving, research context is both stimulating and efficient (which means that the extra topics do not short change the physics education). If you are interested in writing books that do this, or in describing your educational program, please contact the author.

## References

[1] Report of the joint AAPT–APS task force of graduate education in physics, http://www.aps.org/programs/education/graduate/upload/Graduate-Education-in-Physics-Joint-APS-AAPT-Task-Force-Report.pdf, June 2006.
[2] R.H. Landau, M.J. Paez, C.C. Bordeianu, A Survey of Computational Physics, Princeton University Press, Princeton, 2008, http://press.princeton.edu/titles/8704.html, http://www.physics.oregonstate.edu/~rubin/Books/Survey/.
[3] P. Mulvey, B. Shindel, Physics bachelor's initial employment data from the degree recipient follow-up survey for the classes of 2006 and 2007, http://www.aip.org/statistics/trends/reports/emp2010.pdf.
[4] Skills used frequently by physics bachelors in selected employment sectors, Tech. report, Amer. Inst. of Physics Education and Employments Statistics Division, 1995;
R. Ivie, K. Stowe, The early careers of physics bachelors, Amer. Inst. of Phys., College Park, MD, 2002, AIP Pub. R-433.
[5] Video lectures derived from [2], http://www.physics.oregonstate.edu/~rubin/COURSES/VideoLecs/.
[6] eTextBook, http://www.physics.oregonstate.edu/~rubin/BOOKS/eBook.
[7] comPADRE, Digital resources for physics & astronomy, http://www.compadre.org/portal/.