AC 2010-1084: LEARNING DYNAMIC SYSTEMS THROUGH THE HELP OF COMPUTER PROGRAMMING

Tanja Magoc, University of Texas at El Paso

Eric Freudenthal, University of Texas, El Paso

Francois Modave, Central Washington University

Learning Dynamic Systems through the Help of Computer Programming

Abstract

Dynamic systems are not easily understood by students entering college due to complexity of underlying concepts, which are frequently stated but not understood in early mathematics and science courses. Moreover, students majoring in disciplines other than computer science, such as biology or finance, are usually resistant to taking computing courses and view them as irrelevant to their field of study. On the other hand, it is nowadays practically impossible to study, understand, and analyze dynamic systems without help of computers. Computation for Science and Engineering (CompSE) is an introductory computer programming course that will be offered in the Spring 2010 with the new curriculum that is anticipated to attract STEM students to computation fields. The CompSE introduces programming concepts through study of dynamic systems taken from various fields such as finance, molecular biology, and environmental science. Moreover, CompSE introduces students to using computer programming and simulation to analyze systems that are hard or almost impossible to understand and model using only analytical methods. This course is expected to increase students' interest in multi-disciplinary studies, which include computation and programming.

Introduction

Dynamic systems, such as ballistics and resonance, are not easily understood by students entering college. Even harder to understand are unstable natural systems, such as market fluctuation and development of cancer. Nowadays, it is practically almost impossible to study, understand, and analyze dynamic systems without help of computers. Even though numerous computer simulations are widely available on web pages to visually present dynamics of some systems, the reasoning behind these simulations is not readily available and understandable to college students. Moreover, the students are limited to visualizing only samples that are provided on web pages rather than being able to experiment on their own in order to understand causes of certain behaviors.

On the other hand, if students posses basic computer programming skills, they would be able to create their own simulations, compare their solutions to the existing ones, and perform additional experiments. Moreover, in order to produce a computer program that correctly simulates a natural phenomenon of interest, students would have to thoroughly understand the dynamics of the system.

However, many college students, with the exception of those majoring in computer science, are resistant to taking computer programming classes since the curriculum of computer science classes principally focuses on the syntax of a particular programming language. Furthermore, programming projects are typically structured to provide practice of programming concepts rather than examining the application of programming to STEM studies. Computer Programming for Scientists and Engineers (CPSE) is a course offered at the University of Texas at El Paso intended to teach basic computer programming skills to undergraduate students majoring in STEM disciplines other than computer science. This course, which previously focused on the syntax and semantics of the C language, attracted too few students and was largely viewed as irrelevant by faculty in attendees' home departments.

This paper describes a new curriculum for CPSE titled Computation for Science and Engineering (CompSE), which is being introduced in Spring 2010[i]. CompSE uses real-life examples, such as investment, production-consumer markets, predator-prey models, agricultural ecosystem, muscle contraction, and development of cancer, to introduce students to representing natural phenomena as dynamic systems. In order to effectively engage students with little prior experience in system modeling and simulation, initial projects will examine simplified systems. Subsequent projects will progressively expose additional complexity until a realistic system is understood and simulated. Through this approach the course exposes the utility of simulation in examining the evolution of realistic dynamic systems whose complexity limits the accessibility of analytical evaluation. CompSE thus motivates the use of computation to the study of compelling realistic situations rather than hypothetical or simplified examples frequently found in coursework.

Moreover, programming skills are introduced throughout a semester. The presentation of programming concepts is suited to the dynamic systems at hand, rather than following a more traditional approach in which programming concepts to be taught are determined first, and the examples to demonstrate these notions are chosen based on the concepts that are to be explained.

One of the main goals of CompSE is to attract students not majoring in computer science to computational sciences courses and stimulate their interest in continuing study of computation. To achieve these goals, besides teaching students the basics of programming, CompSE aims at attaining two other learning objectives:

1. Students will examine and understand the basis of analytical techniques that they have probably seen and memorized in mathematics or science courses but have frequently not been comprehended deeply enough to be able to apply them to real-life problems, thus enabling students to use analytical techniques in real-life problems that they have not seen before.
2. Students will learn how computation can be used to analyze problems that are difficult or impossible to understand using only analytical techniques, thus enabling students to approach real-life problems that they previously could not tackle.

Our planned evaluation will examine both the effectiveness of the course in achieving student learning objectives, and student interest in continuing multi-disciplinary studies that include computation and programming.

Computation for Science and Engineering

The new curriculum of the CompSE will be offered for the first time in the Spring of 2010. By this curriculum, rather than focusing on syntax, CompSE will immerse students in problem solving and incrementally introduce language features in an as-needed manner. Python is selected as the initial programming language due to its simplicity to convey major programming skills, and Java is introduced later in the semester to demonstrate to students the ease of learning a new programming language once they are comfortable using Python.

The new curriculum still examines foundational programming concepts such as input and output, branching and loop statements, objects and functions. However, the presentation of these concepts is driven by the needs of the projects being examined rather than following a more traditional approach in which programming concepts to be taught are determined first, and the examples to demonstrate these notions are chosen based on the concepts that are to be explained.

Families of dynamic systems are examined as modules from various disciplines including finance, molecular biology, and environmental science. Each dynamic system is originally presented under simplifying assumptions to make it easier for students to understand the basic principles of the system at hand. The layers of complexity are progressively added to the model of the system to increase the realism of the computational model.

Below, we describe several modules of CompSE. Each module examines a sequence of evolving challenges and refinements to develop realistic systems from initial simplified easy-to-understand models.

A. Investment Module

We begin with a financial investment model that first implements simple interest offered by a bank. Even though it is clear that, using this approach, the amount of money in a bank account is represented by a linear function, the students are asked to simulate the balance of the account in consequent years, for a given initial investment and interest rate, without finding the exact equation of the line. Only after experimenting with different initial amounts and interest rates, students become familiar with expected behavior of the investment system in the case of a simple interest rate. Next, students are asked to relate the behavior of the system to known mathematical functions, and therefore deepen their understanding of simple mathematical functions, such as linear functions.

The next layer of complexity is added by considering compound interest, which could be represented by an exponential function. Students perform steps similar to those they followed in the case of a simple interest, which leads them to understanding a more realistic investment system and, at the same time, understanding the behavior of exponential functions.

The following labs involve multiple investments at predefined time intervals, for example, monthly or annual investments. This extension adds much flavor to the investment problem with minimal additional work required by students. Even though this problem seems to be much more complex than the originally attempted problem of just one-time investment, it could still be represented by an exponential function, and thus leads students to much deeper understanding of the investments in bank accounts.

Subsequent labs examine the selection of alternative investment instruments such as stocks. The selection of stocks in which to invest money is usually based on two criteria - return and risk of the stock. For the return and the risk, we can define a stable (possibly linear) relationship: if the return of a stock increases, the risk of investing in this stock increases as well[1]. However, in reality, when the risk reaches a certain high point, no one will invest money in the stock even if it offers a really high return, and therefore the established (linear) relationship between return and risk becomes invalid. The CompSE course explores how this new phenomenon could be modeled. Naïve models are demonstrated to be inadequate when the resulting system reaches an absurd state. Various approaches to refining the models are examined such as piece-wise corrections, higher-order approximations, or alternate models.

One option is to determine the boundaries of the regular behavior (i.e., until which point the relationship between risk and return is linear), where the branching condition is placed. The already built model is used until the boundary condition is reached, and a new model is built for the remaining part of the problem.

Another option is to throw away the existing model and build a new model to address the new situation. Instead of using a linear relationship between return and risk, possibility of using mathematical functions that tend towards some horizontal asymptote are examined, such as exponential and logarithmic functions.

Furthermore, the course explores models of how a financial system can continue to evolve including the risks of oscillatory behavior, market crash, and various approaches that can lead to stable recovery.

Market crashes could result in losing money, thus the amount of money in investment instruments decreases rapidly in this situation. If appropriate attention is not taken during the simulation, this amount might become negative, which is not an adequate representation of the real-life situation. In order to repair the simulation, students need to determine when exactly the value becomes zero. However, it might not always be possible to find the exact value due to granularity of the simulation. Thus, students are forced to accept an approximate value as the solution, therefore facing the round-off error in this particular problem.

The finance module of CompSE is accessible to students with a variety of academic backgrounds. The course assumes that students have only a basic understanding of investments.

B. Environmental Science Module

CompSE continues by studying dynamic systems from other disciplines including environmental science by examining the impact of environment on living organisms.

Through the simulation of a simple agricultural ecosystem, the lab sequence examines the nature of random processes, their effect on dependent stateful dynamic systems, and the Monte-Carlo method. At each time step, the evolved health (state) of the plant population is modeled as a function of current health and current climate conditions. For example, a simplistic model of a healthy plant will transition through several states of diminished health when subjected to drought, and its recovery to a healthy state will similarly require a prolonged period of adequate irrigation.

Lab exercises will examine the nature of randomness in climate models. For example, in an early lab, the simulation of irrigation water availability is naively simulated as an independent random daily event. Students can run multiple Monte-Carlo experiments and determine the relationship between expected availability of irrigation water and probability that the crop will fail.

Subsequent exercises challenge students to develop more realistic climate models by assigning probabilities to random variables representing water irrigation and to examine their effect on plant resilience. In addition, projects can model or simulate more complex models of ecosystems such as ones that include reproduction or competition.

C. Molecular Biology Module

In contrast to the familiar finance and environmental science examples described above, it is unusual for students not studying biology to be familiar with the processes of molecular biology. This section offers most students the opportunity to understand unfamiliar biological processes and approaches to simulating uncertainties and constraints.

One of the problems that we model in CompSE is development of cancer. For example, research has shown that four types of cells' mutations contribute to development of cancer[2]. These four types of mutations are explained to students with enough details to offer understanding of the problem and the opportunity to develop a starting point in building a meaningful model for prediction of cancer given a current state of a patient.

Since the prediction of cancer is an ongoing long term project in medicine, the students are not expected to build the entire model. Rather the students brainstorm about possible approaches to model development of cancer. Moreover, they face limitations of particular techniques for simulation of the system on a smaller scale.

Since many cell mutations remain unexplained nowadays, students are faced with a somewhat random generation of mutations. Based on the existing literature on cell mutations, it is not known when a mutation will occur. However, the rate at which each type of mutation occurs is well-studied and documented. Thus, even though the mutations are random, the students need to simulate these mutations taking into consideration the estimated rates of each mutation.

While students' models of cancer development will be simplistic, students will learn the importance of simulation and the importance of multi-disciplinary studies.

Anticipated Outcomes

Anticipated outcomes of CompSE include:

1. increase in students' interest in continued multi-disciplinary study, which include computation and programming;
2. students' deeper understanding of the dynamics of real-life systems from various disciplines including finance, molecular biology and environmental science;
3. students' mastery of basic programming concepts;
4. students' understanding of common simulation techniques.

With a huge amount of data available, it is currently almost impossible to study, understand, and analyze dynamic systems without the aid of computers. Thus, one of the main goals of CompSE is to involve in computational sciences a larger number of students from various disciplines (e.g., biology, finance) by providing a more problem-oriented approach to teaching introductory programming course for students not majoring in computer science. Since the CompSE investigates examples from various disciplines, some of the problems examined will be close to the attendees' field of study. For this reason, it is expected that the new curriculum will not only attract a larger number of students to study the first course in programming, but also increase the interest of these students to continue enrolling in computing classes and apply the material learned in these classes to multi-disciplinary problems that rely on quick analysis of large datasets.

Moreover, CompSE aims at improving students' understanding of real-life dynamic systems by the use of computer simulations. Since simulations allow students to quickly evaluate the impact of different parameters in a model, the behavior of dynamic systems is better understood by experiments. Furthermore, students are challenged to develop these simulations on their own, thus a thorough understanding of the dynamic system at hand is needed.

Even though the focus of CompSE is on modeling dynamic systems, basic computer programming skills are covered in this course. Simple concepts such as input, output, and control statements, as well as more complex concepts such as objects and functions, are studied in the CompSE. The programming concepts are introduced gradually and examined through addition of layers of complexity to dynamic systems.

Students are also expected to gain a good understanding of the advantages and disadvantages of different modeling techniques with respect to their accuracy, simplicity, and speed. It is anticipated that students will also understand the benefits and drawbacks of small scale simulations of large dynamic systems as well as limitations faced during simulation phase.

Evaluation

Our planned evaluation will examine both the effectiveness of the course in achieving student learning objectives, and student interest in continuing multi-disciplinary studies that include computation and programming.

The main student learning objectives include understanding underlying concepts and complexity of dynamic systems, mastery of basic programming skills, learning simulation techniques along with their limitations, and ability to use simulations to solve problems that are hard or even impossible to solve using only analytical techniques. The achievement of these objectives will be measured by two methods: applying in-class examinations and designing multiple lab assignments to challenge students to apply learned concepts to model dynamic systems that they have not seen yet.

Students' interest in continued study of computation will be measured by pre- and post- class online surveys. These surveys will measure:

- students' beliefs that computation is necessary in their field (e.g., finance, biology);
- students' confidence about understanding dynamic systems;
- students' confidence about their programming skills;
- students' intention to take more computing courses;

- students' interest to engage in a multi-disciplinary career.

The comparison of students' answers at the beginning and the end of the semester will determine the change of students' attitudes towards CompSE, other computation courses and multi-disciplinary fields of study. These surveys and analysis of surveys will be prepared and applied in collaboration with an external evaluator.

Synopsis

CompSE is an introductory computer science class for students not majoring in computer science. This course aims at attracting more students to engage in multi-disciplinary study, research, and career by providing a problem-oriented approach to learning programming and understanding dynamic systems.

Acknowledgement

This report is based on work supported by the National Science Foundation through the grant IIS-0829683. Any opinions, findings, and conclusions or recommendations expressed in the paper are those of the authors and do not necessarily reflect the views of the NSF.

Bibliography

1. M. Joshi, The Concepts and Practice of Mathematical Finance. Cambridge, 2003.
2. S. L. Spencer, M. J. Berryman, J. A. Garcia, and D. Abbott, "An ordinary differential equation model for the multistep transformation to cancer," Journal of Theoretical Biology, no. 231, pp. 515-524, 2004.