“CPSC 420 — Modeling and Simulation”

Professor: Stephen Davies
Spring semester 2018
Class: MWF 2pm in Trinkle B7
Final exam: Monday, April 30th, 3:30–6pm
Office Hours (in Trinkle B22):
  Tue 3:30–4:30pm
  Wed 1–2pm and 3–4pm
  Thu 12–2pm

http://cs.umw.edu/~stephen/cpsc420

Computers are a tool that can help us understand our world.

It’s true, of course, that computers do many other things as well: they store and retrieve information, carry out financial transactions, and display 3-D graphics of spaceships. For this reason, we’re sometimes likely to see computer programs principally in terms of what we can build with them, rather than what we can learn from them.

This class, by contrast, is about seeking understanding, not about building devices.

For a long time, there have essentially been two ways to gain knowledge. The first one — theory — goes back to before Aristotle. We analyze our world, using our reason, and come up with simplifications, abstractions, and categorizations that help us make sense of it. Around the 17th century, a second method came on the scene: experimentation. Instead of just speculating about how things behave, we subjected Nature (and People) to principled testing, seeing if we could reliably and repeatably predict how they will respond.

The 20th-21st centuries introduced a third way: modeling and simulation. This new paradigm, made possible by the digital computer, helps us answer some questions that were unanswerable previously. Estimating the energy in a nuclear chain reaction? Plotting the trajectory of a hurricane? Predicting the outcome of a political revolution? Anticipating how large a city can get before its crime rate increases? None of these problems or thousands of others are solvable without adding this powerful third weapon to our arsenal.

At the center of this new approach are abstract computational models that extrapolate outcomes from innumerable tiny elements and rules. And the complexity that we see arising from lots of simple things interacting in seemingly simple ways is breathtaking.

We technologists have a powerful skill set which can further our knowledge of society and of the universe. The amazing thing is that without even getting our hands dirty in a petri dish, the programs we write shed real illumination on real problems. Our models and their simulations become the science in a very real way, as we leverage the computer to attack the grand challenges of our age.
Course Objectives

- To give you a high-level overview of the techniques, promises, and limits of computational modeling as a way to understand complex phenomena.

- To tie together the two halves of Data Science. Whereas a course like 419 (Data Mining) is about analyzing data to infer properties about the process that generated it, 420 is about creating models of the process itself and seeing whether the data it generates matches up. It’s kind of like 419 in reverse.

- To introduce you to three of the important modeling and simulation paradigms: System Dynamics, Cellular Automata, and Discrete-Event Simulation. Different real-world problems lend themselves to one (or more) of these different types of approaches. You will learn which ones are most appropriate to a particular situation, and how to recommend the best way or ways to go about implementing them.

- To give you some experience in implementing and evaluating models, so that you understand the issues involved in designing a simulation that is likely to give helpful and reliable results.

- Finally (and seriously) to improve your understanding of life in general and make you a better thinker.

Rules of the game

1. There are absolutely, positively, NO stupid questions!! Your job is not to already know everything before you start the course. Your job is to try hard to learn, and part of that involves asking questions. I’m a nice guy, and I will not ever belittle you, snub you, or make fun of you; and if anyone else does so I will personally break both of their arms.

2. This class will be interactive. When I point at you in class, say your first name, and be prepared to try and answer questions. (Don’t worry if you don’t know all the answers.)

3. This class involves a good bit of reading. This is because the class involves a good bit of learning, and with apologies to this video-centric generation, most deep learning still comes through reading. Part of what you should get out of this class is an improvement in your reading habits. I’ll work hard on my end to try and make that happen, since I think becoming a more efficient and effective reader is actually the most important thing you can get out of college, period. What you need to do is have a good attitude about it and dedicate yourself to rolling up your sleeves and digging in.
4. Don’t skip class. Just don’t. It’s bad form. I work hard to prepare for class, to make it compelling and relevant. It hurts my feelings when you don’t come. Plus you miss out on important stuff, and you’ll end up falling behind if you skip lecture. So come every time. Come happy, fresh, excited, ready to think and to participate.

5. Don’t cheat. Cheating is heinous, rude, and bad karma. It really makes me mad, and it will also eat away your character like hydrochloric acid if you’re not careful. If you ever feel tempted to cheat, in this class or any other, come and talk to me about it. It’s not wrong to feel tempted, and we can find other ways out of whatever dilemma you’re facing without compromising your moral character.

6. Absolutely no laptops, cell phones, or other devices during class. I’ve had students claim that they take notes on their laptop during lecture, but even if it’s true, those things are way too big a distraction to you and your fellow students to make it worth it. Just stay tuned in, because I move fast.

The Honor Code and this course

For this course, all the work that you turn in for a grade must be solely your own work, period. Specifically, this means:

- The quizzes must be taken alone, in a quiet place, without any form of contact with anyone.

- You must write all your own programs in their entirety. I don’t mind if you chat informally about the programming assignments with your fellow students, but you must not show anyone else your code nor look at anyone else’s code. This includes people who are not in the class nor even at UMW. I am happy to help you over email or in office hours about whatever questions and problems you have.

Books


Dana Meadows was, in my opinion, one of the true geniuses of the 20th century, and her influence has been vastly underrated. Studying under Jay Forrester and with a group of highly talented thinkers at MIT, she developed many of the notions of what we today
call “system dynamics,” and applied them in mind-boggling ways to the environmental and economic systems of the world. Her most famous book, Limits to Growth, was a worldwide best-seller and very controversial: it predicted, essentially, the End Of The World As We Know It unless things changed, and fast. Things changed, but slow, leading to a continued grim view of the future that most of today’s world tends to ignore. The book we’re reading for this course was published posthumously, and is a little-known gem. It contains, in distilled form, much of Meadows’ philosophy about system dynamics, and is so well-written that its tremendous depth can get lost in the ease with which it is read. It has been very influential in my own thinking, and I’m excited to share it with you.

Among the greatest still-living geniuses of the 21\textsuperscript{st} century, in my opinion, is the hugely influential Nate Silver of \texttt{fivethirtyeight.com} fame. He got his start forecasting statistical outcomes for major league baseball players, of all things, but has since done incredibly broad and ground-breaking work analyzing statistical models of all kinds. His book The Signal and the Noise is hands-down one of the best books I have ever read, and like Meadows’ book has literally changed the way I look at life. It is particularly relevant to this class, as Silver has years of sober-minded expertise to share about simulations and their predictions.

**Grading**

- 10% – One-minute, \texttt{closed}-book, \texttt{open}-note “Reading checks” at the start of most classes. These are intended just to hold you accountable to the reading for the day, and should be easy as long as you have done so.
- 25% – Eight \texttt{open}-book, \texttt{open}-note Canvas quizzes at various times throughout the semester. I’ll drop your lowest two scores.
- 15% – Six lab-ish homeworks, designed to give you some hands-on practice with the class concepts.
- 25% – Six substantial programming assignments, each creating and analyzing a computational model.
- 25% – Final exam, comprehensive, open-book, open-notes, Monday, April 30th, 3:30–6pm.

**Late policy**

No late work will be accepted this semester. Get your stuff in on time!
Submitting programs

Rules for submitting programs will be given when the program is assigned. Most of the time, you’ll be emailing me your program code as an attachment, and using a specific subject line to distinguish it from my hordes of other email. Meeting the deadline is a matter of sending your email before time expires.

Also, most of my homeworks are due at “midnight.” Here’s what “midnight” means: if a homework is due “at midnight on Thursday,” then it is due after all of Thursday has elapsed, and the clock strikes twelve. (In other words, this is good news: you have all Thursday to work on it.)

Basis for determining mid-semester reports

For midterm progress reports, I look mostly at (a) whether you’ve been turning assignments in (and preferably on time), and (b) quiz scores. If either or both of these categories are lacking, it’s a sign of danger, and I will give you a “U” for your mid-semester grade. Please don’t hesitate at all to come talk to me about this so we can figure out how you can do better in the course.

Guidelines for class participation

I believe that students learn best when they participate wholeheartedly in all aspects of the learning process. Hence while your grade will not be partially determined by any “class participation score” *per se*, it is very much to your advantage, and very much recommended, that you join in during class discussions, ask questions, and make comments.

Disabilities

If you have a documented disability, please present me your letter from the Office of Disability Resources and I’ll be happy to accommodate you.

How to reach me

Come to office hours, see me after class, or e-mail me (stephen@umw.edu).
How to reach you

I will be communicating with you outside of class time via e-mail, so make sure to check your UMW e-mail every day! I will also post announcements to the course website, so be sure to subscribe to its RSS feed and check it in your feed reader at least once a day!

Calendar

The official calendar for the course, complete with assignment due dates, tests, etc., will be maintained on the course website at http://cs.umw.edu/~stephen/cpsc420. In any way that the website conflicts with the tentative calendar below, the website is to be considered correct, and the tentative calendar below out of date.

<table>
<thead>
<tr>
<th>Week</th>
<th>Topics</th>
<th>Readings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Modeling and Simulation principles</td>
<td>Meadows: Intro &amp; ch. 1a</td>
</tr>
<tr>
<td></td>
<td>The Python ecosystem</td>
<td>Silver: Preface &amp; Intro</td>
</tr>
<tr>
<td>2</td>
<td>System Dynamics models: stock &amp; flow diagrams</td>
<td>Meadows: ch. 1b</td>
</tr>
<tr>
<td></td>
<td>Numerical calculus</td>
<td>Num. calc. tutorial</td>
</tr>
<tr>
<td>3</td>
<td>Exponential and logistic growth patterns</td>
<td>Meadows: ch. 2</td>
</tr>
<tr>
<td>4</td>
<td>System patterns and analysis</td>
<td>Meadows: ch. 4</td>
</tr>
<tr>
<td>5</td>
<td>SD models: Lotka-Volterra, SEIR</td>
<td>Meadows: ch. 5</td>
</tr>
<tr>
<td>6</td>
<td>SD models: Competitive exclusion, delays</td>
<td>Silver: ch. 1</td>
</tr>
<tr>
<td>7</td>
<td>Cellular Automata</td>
<td>Silver: ch. 2</td>
</tr>
<tr>
<td>8</td>
<td>Implementing stochastic simulations</td>
<td>Silver: ch. 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Prob/stat tutorial</td>
</tr>
<tr>
<td>9</td>
<td>Analyzing stochastic simulations</td>
<td>Silver: ch. 5</td>
</tr>
<tr>
<td>10</td>
<td>CA models: Wolfram, Schelling</td>
<td>Silver: ch. 6</td>
</tr>
<tr>
<td>11</td>
<td>Diffusion models</td>
<td>Silver: ch. 7</td>
</tr>
<tr>
<td>12</td>
<td>Discrete Event Simulation</td>
<td>Silver: ch. 8</td>
</tr>
<tr>
<td>13</td>
<td>DES models: Queueing systems</td>
<td>Silver: ch. 11</td>
</tr>
<tr>
<td>14</td>
<td>Survey of Agent-Based Modeling</td>
<td>Silver: ch. 12</td>
</tr>
</tbody>
</table>