Operations Research for CS

CSCI 4XX

Instructor Info

Mete Saka

Office Hrs:

CTLM 114

http://metesaka.com

saka@mines.edu

Course Info

Prereq: Basic Linear Algebra, Data Structures, Python

MWF

11a-12p

Lecture Room

TA Info

Alan Turing

Office Hrs: Tues & Thurs 10-11a

CTLM 114

Overview

What is the best course of actions to do with limited resources?

This is a question that we ask ourselves many times a day intentionally or unintentionally. When we shop, schedule our day, choose what to study, or basically anything that involves a decision, we encounter that one or many resources involved in our decision process is limited. In this context, Operations Research provides us numerous methods to model a real-life problem and then find the optimum solution of the model.

At this point, you might notice what we describe is leading to something you might heard of: Optimization (which is one of the key objectives of operations research).

Operations Research (OR) is the broad collection of techniques, used for understanding and modelling complex systems, determining its problems, purpose, and effectiveness, and optimizing the processes to gain maximum efficiency. These techniques are including but not limited to, mathematical optimization, linear-integer-nonlinear programming, simulation, queuing, graph and network analysis.

As a computer scientist, many algorithms, models, and techniques that you've learned and applied are also used in OR with different perspectives. For instance, foundations of Machine Learning rely on optimizing many functions to find the best explanation for a phenomenon.

This course is meant to show you the core concepts of OR, together with numerous modelling techniques and optimization algorithms. Within the course, several graph problems (i.e. shortest path), algorithms (i.e. binary search) and Machine Learning techniques (i.e. Clustering, ANN), you encounter in other CS courses will be revisited.

The main benefit of the course would be enhancing your understanding of Algorithms, Simulation, Statistics and highlight your Mathematical Modelling skills. For each module of the course, we will take a real-life problem, model it, implement the model, and use one (or more) of the techniques we learn to find the optimal solution. Additionally, throughout the semester, you'll work on a project for each module, model several real-life problems, implement your solution. Finally, we will have an end of the year study. You can either propose a research project or find a case study within your interests, and implement your solution for the problem using the methods we learn. Instead of a final exam, we will gather up and you'll present your study to the class in the end of the semester.

Material

Required Texts

TBA

We will have reading assignments for each week.

Recommended Text

Any required journal articles and book chapters will be provided on Canvas.

Course Learning Objectives

1. Develop formulations for practical problems into appropriate mathematical models
2. Compare and contrast which algorithmic method is the best to choose, based on your model
3. Implement the models and algorithmic methods using software, solve the model and analyze its outcomes in reasonable amount of time.
4. Integrate the extensive interpretation of model outcomes and propose model-driven solutions the problem.
5. Exemplify possible integration of models and methods learned in class with a software engineering approach.
6. Differentiate the nature of a problem such as, non-deterministic, deterministic, stochastic, optimization, decision, ...

Assessment Plan

This course uses mastery grading. You will be expected to get a mastery grade in each major course area. The assignment evaluation is solely aimed to provide feedback.

Associated CLO	Assessment:
1,2,6	Weekly Readings, Reading Cards
1,2,4,6	Weekly Homework
1,2,3,4	Module Projects
1,2,3,4,5	Case Study / Research Project (End of the Term)

Grading Scheme

This course uses mastery grading. You will be expected to get a mastery grade in each major course area. The assignment evaluation is solely aimed to provide feedback.

Definitions:

E	Exemplary
S	Satisfactory
N	Needs Support
U	Unsatisfactory

To get an B in the class, you're expected to obtain a minimum of S in each module of the class. Additionally, if x-y of them are E, then you will be evaluated for an A in the class. To pass the class (with C), you have to get S for at least y-z of the modules. This course does not grade D.

Late Work Policy

Throughout the course, treat your grades as our feedback to your progress. They will not be strictly correlated with your final letter grade. Similarly, treat the due dates as recommended completion time not a deadline. The only hard due date of this course is last day of classes, after which the course team has to evaluate your final progress.

Inclusive Teaching Statement

In this course I assume you are familiar with the basics of linear algebra, and core CS skills such as Algorithmic Thinking, Data Structures and Programming. Other than these, all concepts in the class will be provided with the relevant and required background. It is highly appreciated to communicate us whenever you notice a contradiction with the this.

In this course, we encourage all of you to contribute to the course with your own experience, collaborate between your peers and communicate your thoughts with the classroom. Furthermore, we strongly recommend you contact the course team for any question or concern. This includes but not limited to, in class questions, emails, texts, discussion board etc. Our course has a "no questions / no judgements" policy, which means no one in the course team will ask you questions like "why did you ask this to me so late?" or hear statements like "99.999999% of the class did it so it's your fault" or "Oh I see, you missed abc, xyz, already; you took 0 on this assignment; you won't achieve in this course".

To enhance your understanding, majority of the assignments will be derived from real life industry or academia problems. Additionally, depending on your interests, you will be able to choose between different assignments at certain stages of the course. If time permits, we will invite guest speakers to the class to show you how you can utilize the course content in the real-world applications.

Class Schedule

MODULE 1: Optimization

Week 1	Linear Programming	Introduction and Fundamentals
Week 2	Linear Programming	Modeling
Week 3	Integer Programming	Fundamentals and Modeling
Week 4	Non-Linear Programming	Fundamentals and Modeling
Week 5	Mixed Models and Implementation	Mathematical Modeling and Solvers
Week 6	Project Week and Review	Module 1
	Project 1 DUE: End of Week 6	

MODULE 2: Stochastic Processes and Simulation

Week 7	Stochastic Models	Introduction and Fundamentals
Week 8	Stochastic Models	Monte Carlo
Week 9	Stochastic Models	Discrete Event Simulation
Week 10	Project Week and Review	Module 2
		Project 2 DUE: End of Week 10
		Case Study / Research Proposal DUE: End of Week 10

MODULE 3: Network Models, Graphs

Week 11	Network Models	Introduction and Fundamentals
Week 12	Network Models	Optimization Problems
	Mini Project	Due: End of Week 13

MODULE 4: Algorithms (Search, ML,ANN)

Week 13	Search Algorithms in Optimization	Fundamentals
Week 14	Machine Learning, Neural Networks	Introduction to optimization in ML / NN
Week 15	Machine Learning, Neural Networks	Model Reviews
Week 16	Comprehensive Review	Presentations

Notes/Reflection

The idea of this course comes to my mind due to two reasons. My background is in Industrial Engineering and OR is what we learn generally. After I start my masters in CS, I noticed that many topics in CS share similar concepts and ideas with OR, so it might be interesting for CS students to learn OR that will broaden their perspective. Second reason is, I observed that most of the struggle of CS curriculum comes from mathematical, algorithmic and conceptual parts of CS. So, a course that approaches these parts from a different perspective would be interesting for the students and contribute to their learning. However, the course might create confusion and for sure, it will be a hard course. To maintain the student engagement, I would focus on choosing similar topics that they already familiar with to reduce frustration. Additionally, all projects and homework will be chosen from real-life CS problems.

After the first feedback I noticed that my syllabus was a little offensive (the military background part of OR). Therefore, I wrote nearly all my syllabus again from scratch. I tried to focus on how to communicate how this course will be relevant to a CS student.

To write the inclusive teaching statement, I was influenced by the seminar and the discussion we had at CTC, and additionally my personal experience as a student and instructor affect several policies I provided above.