COMP 7720 - Online Algorithms

Shahin Kamali

Lecture 1 - Sep. 7, 2017
University of Manitoba

Picture is from the cover of the book by Borodin and El-Yaniv. See Slide 6.
Introduction
In a Glance . . .

Online algorithms are

- Practical
- Diverse
- Fun (really!)

Let’s ‘play’ with online algorithms and enjoy
Instructor

- B.Sc. in Computer Science (2002-2006)
  - University of Tehran (Iran)
- M.Sc. in Computer Science (2006-2008)
  - Concordia University (Canada)
- PhD in Computer Science (2008-2014)
  - University of Waterloo (Canada)
  - LIAFA, Paris (France)
- Postdoctoral Fellow, Associate (2015-2017)
  - MIT (USA)
Students

Your turn to ...

- Introduce yourself.
- Your research group, research interests.
- What makes you interested in this course?
Formalities
## Logistics

- **Lecture:** Tuesdays and Thursdays, 10:00-11:15am  
  EITC E2 Rm: 360 (Sep 07, 2017 - Dec 08, 2017)
- **Webpage:** [http://www.cs.umanitoba.ca/ kamalis/comp7720.html](http://www.cs.umanitoba.ca/ kamalis/comp7720.html)
- **Piazza:** [https://piazza.com/umanitoba.ca/fall2017/comp7720](https://piazza.com/umanitoba.ca/fall2017/comp7720)
- **Office hours:** 11:30 am-12:30 pm, Tuesdays and Thursdays, in E2 586 or by appointment
Textbook

- A list of required reading will be provided on the course webpage
- No book is required to be purchased.
- The following book is suggested as a reference:
  - Borodin and El-Yaniv, Online Computation and Competitive Analysis (2005)
Grading

- There will be:
  - Four assignments
  - A final exam
  - A course project

**Theorem**

*The focus of this course is on learning, discovering, and research.*

**Corollary**

*Don’t worry about your final mark.*
Grading (cntd.)

- Four assignments:
  - 30 percent of the final mark
  - 10 percent extra for bonus questions.
  - submit only pdf files (preferably use \LaTeX)

- Final exam:
  - 30 percent of the final mark
  - 5 percent extra for bonus questions.
  - It is a closed-book exam.
  - A sample exam will be provided for practice
Projects

- Course project
  - 40 percent of the final mark
  - 20 percent extra for outstanding projects (publishable projects)
  - Work individually or in groups of two

- Projects involve:
  - Proposal
  - Presentation
  - Final report (in form of a research paper)
Projects (cntd.)

- Project topics will be suggested in the first few weeks of the class
  - You can chose your own topic based on your research
  - Come to office hours to talk about it!

- Project categories:
  - Exploring possible solutions to an open problem
  - Writing a survey paper on a current topic related to online algorithms
  - Writing code to implement and compare the performance of online algorithms for a problem
Important Dates (tentative)

September 21: assignment 1 due
September 28: project proposal due
October 12: assignment 2 due
October 26: assignment 3 due
November 16: assignment 4 due
November 17: VW deadline
November 23: Project presentation starts
December 7: last class
December 7: project final report due
December 14: final exam
Online Algorithms
Offline vs. Online Algorithms

- Traditional algorithms are ‘offline’ in the sense that they have the whole input in their hand.
- Online algorithms, in contrast, do not have/need the whole input in order to solve a problem
  - The input is a ‘sequence’ which is processed by the online algorithm piece-by-piece
  - The online algorithms often take irrevocable decisions to process the input.
Bin Packing Problem

- The input is a set/sequence of items of various sizes
  - E.g., \(<\ 9, 3, 8, 5, 1, 1, 3, 2, 4, 2, 4, 5, 5, 8, 6, 4, 5, \ldots >\).

- The goal is to pack these items into a minimum number of bins of uniform capacity.
Bin Packing Problem (cntd.)

- In the online setting:
  - an algorithm receives items one by one
  - when it receives an item, it has to place it in a bin without any knowledge about forthcoming items
  - decisions of the algorithms are irrevocable (i.e., cannot move items between bins)
First Fit (FF) Algorithm

- Find the first bin which has enough space for the item, and place the item there
- Open a new bin if such bin does not exist

< 9 3 8 5 1 1 3 2 4 2 4 5 5 8 6 4 5 >   < 9 3 8 5 1 1 3 2 4 2 4 5 5 8 6 4 >
Competitive Ratio

- We use the framework of **competitive analysis** to compare online algorithms.
- Let $\text{OPT}$ denote the best possible offline solution.
  - Given a sequence $\sigma$, $\text{OPT}$ is an algorithm which packs items in $\sigma$ in a minimum number of bins.
- Competitive ratio of an algorithm $A$ is the maximum ratio between the cost of $A$ and that of $\text{OPT}$ over all sequences

$$cr(A) \equiv \max_{\sigma} \frac{\text{cost}_A(\sigma)}{\text{cost}_{\text{OPT}}(\sigma)}$$
For First Fit, the competitive ratio is 1.7 [Johnson 1973]

- The number of bins opened by FF for any sequence is at most 1.7 times that of OPT, i.e., $c.r. \leq 1.7$ (upper bound for FF)
- There are sequences for which the number of bins opened by FF is 1.7 times that of OPT, i.e., $c.r. \geq 1.7$ (lower bound for FF)

- The best existing online algorithm has c.r. of 1.588 [Heydrich, van Stee 2017]
- No algorithm can be better than 1.54037-competitive (best general lower bound) [Balogh et al. 2015].
Ski-rental Problem

- Assume you want to go skiing for $x$ number of days
  - In the online setting, the value of $x$ is unknown!
- You can buy the equipment for a one-time cost of $b$ or rent each day for a cost of 1 per day
- If we know $x$, what is the best solution?
  - Buy at the beginning if $x \geq b$, otherwise, rent every day
Sky-rental Problem (cntd.)

- Online strategy **break-even**: rent for the first $b - 1$ days and buy in the next day.

- What is the competitive ratio of Break-even algorithm?

**Theorem**

*Competitive ratio is roughly 2, and it is the best for any deterministic online algorithm.*
Syllabus
Doubling technique

In many occasions, a ‘doubling technique’ can be used to design and analyze online algorithms

- The lost cow problem
- Online bidding
Potential function

- Potential function technique is a classic approach for analysis of online problems
  - The paging problem, Sleator-Tarjan proof, randomized paging
Data structures

- Self-adjusting data structures
  - List update problem
  - Data compression
  - Self-adjusting binary trees, and dynamic optimality conjecture
Packing problems

- Weighting technique
  - Bin packing
  - Renting servers in the cloud
  - Online scheduling
Graph problems

- Graph problems
  - $k$-server problem
  - Graph coloring
  - Bipartite matching
Computation geometry

- Robot searching
- 2-dimensional bin packing
Advice complexity of online problems

- Online algorithms with Advice: what if we have partial information about future?
  - Algorithms with advice for paging, bin packing, list update

- Many project ideas here!
Alternative analysis techniques

- Competitive ratio is a **worst-case measure**
- Alternative analysis techniques are used to compare algorithms based on their typical behaviour
  - Bijective analysis
  - Relative worst-order analysis