

CSC2421: Online and Other Myopic Algorithms

Fall 2026

Allan Borodin

September 23, 2026

Lecture 4

Today's announcements

- Something got messed up in terms of the permissions for the course web page. I am trying to see how to fix the issue. In the meantime, you can access the course web page by accessing my home page, then click on teaching and then click on the course 2421f26. However, the permissions seem to have been fixed.
- If you have any questions about the assignment please let me know. For the research diary, try as best you can to recall what led you to think of any questions you ask of the LLMs. Did the LLM(s) lead you to a new approach to the open problems you are trying to solve? What approaches seemed to fail?
If you quickly get a solution to the open problem you should try another one.

Today's agenda

Today's agenda

- I will repeat a list of greedy algorithms we often study in CSC373. They all fit into the fixed priority template I gave (and will repeat today) with the exception of Prim's MST algorithm and the greedy algorithm for set cover which are adaptive greedy algorithms.
- The basic definitions of the competitive ratio for deterministic and randomized algorithms. ‘
- I will present some open problems listed in the text. I should note here that we can already mention (although it is part of Chapter 8 in Volume 2) that we do not know if LPT is the optimal fixed or adaptive priority algorithm for the makespan problem for uniform machines.

Greedy algorithms in CSC373

Some of the greedy algorithms we study in different offerings of CSC 373

- The optimal algorithm for the fractional knapsack problem and the approximate algorithm for the proportional profit knapsack problem.
- The optimal unit profit interval scheduling algorithm and 3-approximation algorithm for proportional profit interval scheduling.
- The 2-approximate algorithm for the unweighted job interval scheduling problem and similar approximation for unweighted throughput maximization.
- The Kruskal and Prim optimal algorithms for minimum spanning tree.
- Huffman's algorithm for optimal prefix codes.
- Graham's online and LPT approximation algorithms for makespan minimization on identical machines.
- The 2-approximation for unweighted vertex cover via maximal matching.
- The “natural greedy” $\ln(m)$ approximation algorithm for set cover.

The fixed (order) priority algorithm template

$\mathcal{I}$ is the set of all possible input items
Decide on a total ordering π of $\mathcal{I}$
Let $\mathcal{I} \subset \mathcal{I}$ be the input instance
 $S := \emptyset$
 % S is the set of items already seen
 $i := 0$ % $i = |S|$
while $\mathcal{I} \setminus S \neq \emptyset$ **do**
 $i := i + 1$
 $\mathcal{I} := \mathcal{I} \setminus S$
 $I_i := \min_{\pi} \{I \in \mathcal{I}\}$
 make an irrevocable decision D_i concerning I_i
 $S := S \cup \{I_i\}$
end

Figure: The template for a fixed priority algorithm

Some comments on the fixed order priority model

- A special (but usual) case is that π is determined by a function $f : \mathcal{J} \rightarrow \mathbb{R}$ and then ordering the set of actual input items by increasing (or decreasing) values $f()$. (We can break ties by say using the input identifier of the item to provide a total ordering of the input set.) **N.B. We make no assumption on the complexity or even the computability of the ordering π or function f .**
- **NOTE:** Online algorithms are fixed order priority algorithms where the ordering is given *adversarially*; that is, the items are ordered by the input identifier of the item.
- As stated we do not give the algorithm any additional information other than what it can learn as it gradually sees the input sequence.
- However, we can allow priority algorithms to be given some (hopefully easily computed) global information such as the number of input items, or say in the case of the makespan problem the minimum and/or maximum processing time (load) of any input item. (Some inapproximation results can be easily modified to allow such global information.)

The adaptive priority model template

$\mathcal{J}$ is the set of all possible input items

$\mathcal{I}$ is the input instance

$S := \emptyset$ % S is the set of items already considered

$i := 0$ % $i = |S|$

while $\mathcal{I} \setminus S \neq \emptyset$ **do**

$i := i + 1$

 decide on a total ordering π_i of $\mathcal{J}$

$\mathcal{I} := \mathcal{I} \setminus S$

$l_i := \min_{\leq \pi_i} \{I \in \mathcal{I}\}$

 make an irrevocable decision D_i concerning l_i

$S := S \cup \{l_i\}$

$\mathcal{J} := \mathcal{J} \setminus \{I : I \leq_{\pi_i} l_i\}$

 % some items cannot be in input set

end

Figure: The template for an adaptive priority algorithm

Defining the competitive ratio

Lets start with deterministic algorithms for a minimziation problem O (e.g., thje makespan problem).

Let ALG be an online algorithm for a minimization problem and let OPT denote an optimal solution to the problem. The (*asymptotic*) *competitive ratio* of ALG , denoted by $\rho(ALG)$, is defined as follows:

$$\rho(ALG) = \limsup_{OPT(\mathbf{x}) \rightarrow \infty} \frac{ALG(\mathbf{x})}{OPT(\mathbf{x})}.$$

Equivalently, we can say that the competitive ratio of ALG is at most ρ if $ALG(\mathbf{x}) \leq \rho \cdot \mathbf{OPT}(\mathbf{x}) + o(\mathbf{OPT}(\mathbf{x}))$.

This then is just a renaming of the asymptotic approximation ratio as is widely used in the study of offline optimization algorithms.

The definition implies that the competitive ratio is always at least 1. The closer to 1, the better.

The deterministic ratio continued

Some people define the asymptotic ratio as

$ALG(\mathbf{x}) \leq \rho \cdot \mathbf{OPT}(\mathbf{x}) + \mathbf{O}(1)$; that is sufficient but it is not an equivalent definition.

When $ALG(\mathbf{x}) \leq \rho \cdot \mathbf{OPT}(\mathbf{x})$ for all $\mathbf{x}$ we say that the ratio is strict.

The $\liminf_{ALG} \rho(ALG)$ is then the competitive ratio for the problem when considering all algorithms ALG for the given problem.

When we have matching upper (i.e., positive) and lower (i.e., negative) results, we have we have an (strict or asymptotically) optimal; result for a given algorithm or for a given problem.

The competitive ratio of for maximization problem

For maximization problems, there are two ways to state competitive and approximation ratios for a maximization algorithm ALG .

- 1 $\rho(ALG) = \liminf_{OPT(x) \rightarrow \infty} \frac{ALG(x)}{OPT(x)}.$
- 2 $\rho(ALG) = \limsup_{OPT(x) \rightarrow \infty} \frac{OPT(x)}{ALG(x)}.$

There is no clear consensus as to which convention to use. In the first definition we always have $\rho \leq 1$. It is becoming more standard to express competitive and approximation ratios as the fraction of the optimum value that the algorithm achieves especially if the ratio is a non-parameterized constant.

With this convention, we have to be careful in stating results as now an “upper bound” is a negative result and a “lower bound” is a positive result. Using the second definition we would be following the convention for minimization problems where again $\rho \geq 1$ and upper and lower bounds have the standard interpretation for being (respectively) positive and negative results.

The randomized competitive ratio

A randomized algorithm can make random decisions for each input item. (We do not usually consider allowing random choices for the next input item.) We will just define the ratio for a minimization problem and ignore the lim sup. There are analogous definitions for maximization problems. .

The definition for the competitive ratio depends on the power of the adversary. The most common (and the weakest) adversary is called an oblivious adversary that creates a nemesis sequence based on the algorithm but not on the outcomes of the random choices the algorithm is making. In this case the competitive ratio is

$$\frac{\mathbb{E}_{r \sim R} [ALG_r(\mathbf{x})]}{OPT(\mathbf{x})}$$

Note that $OPT(\mathbf{x})$ is not a random variable so that the denominator is not an expectation.

Adaptive adversaries

An adaptive adversary gets to see the outcomes of the random variables (and hence each decision) before choosing the next input item.

There are two types of adaptive adversaries, namely adaptive online and adaptive offline adversaries. In the former case, the adversary also has to decide on a decision for the i^{th} input before choosing the $(i + 1)^{st}$ input item. In the latter case, OPT can make all of its decisions at the end of the game.

For both adaptive adversaries, we define the competitive ratio as :

$$\frac{\mathbb{E}_{\mathbf{r} \sim \mathbf{R}} [ALG_{\mathbf{r}}(\mathbf{x})]}{\mathbb{E}_{\mathbf{r} \sim \mathbf{R}} [OPT(\mathbf{x})]}.$$

Now $OPT(\mathbf{x})$ is a random variable and hence the denominator in the ratio is an expectation.

The relation between the adversaries

Again, let's just state the relations for a minimization problem. We have the following results in the text.

$$\rho_{\text{OBL}}(\text{ALG}) \leq \rho_{\text{ADON}}(\text{ALG}) \leq \rho_{\text{ADOFF}}(\text{ALG}).$$

This follows immediately from the definitions.

$$\rho_{\text{det}}(\text{ALG}') \leq \rho_{\text{ADOFF}}(\text{ALG}).$$

I added “det” for emphasize the deterministic ratio.

NOTE: This is a non-constructive existence result.

$$\rho_{\text{ADOFF}}(\text{ALG}) \leq \rho_{\text{OBL}}(\text{ALG}) \cdot \rho_{\text{ADON}}(\text{ALG}) \leq [\rho_{\text{ADON}}(\text{ALG})]^2.$$

Open problems in Volume 1

I will first mention open problems for the chapters in Volume 1. The topics and results we present represent our view of what we find interesting. The Open Problems also reflect our taste in what would be nice to prove.

Warning: It may be that in a few months (all of these open problems will have been solved. Personally, I think some or maybe even many open problems will remain if for no other reason as these LLMs may not want to spend time on our text. But I do think some will remain open. To be seen.

I might sometimes distinguish between what I think is a more substantial problem but again this is taste. Personally, I don't get too excited about "relatively minor" improvements in a competitive ratio or impossibility result. Of course, even a relatively small improvement can be considered very important if the problem is a long standing open problem.

End of the class

Unfortunately, we lost some time due to technical issues with the connection to the projector. I am including the rest of the slides I had for today. I will be commenting on various open problems and will include the remaining slides in the slides for the next class.

Some specific open problems in Volume 1

Chapter 2:

We only have listed three open problems for this chapter. I don't consider narrowing the gap between the known negative and positive results here to be that important. I consider open problems 2 and 3 to be examples where the interest would be mainly if one could obtain an optimal or near-optimal result for these classic problems.

I just want to point out the relation between the time series problem and the secretary problem. Namely both are trying to pick a single winner. In the time series problem, the input sequence is adversarial whereas in the secretary problem the sequence is random order.

In later chapters (in both volumes), there are some open problems where we generalize a problem from adversarial order to random order. Similarly, we can generalize an online problem to the same problem with priority orderings or best orders.

Open problems: Chapter 3

Open problem 1 seems like a problem where we should know the answer but it isn't clear to me what the answer is. It might be in the literature somewhere. However, be careful since some authors define the unweighted knapsack problem to what we believe should be called the proportional weight knapsack problem.

We can also consider a problem related to Open problem 2; namely extend the proportional weighted result to C-Benevolent weights as defined in chapter 5 of volume 2.

Open question 3 can be more generally considered for all randomized competitive ratios which only provide expected ratios and could perform very poorly on some sets of inputs.

We like Open question 4. It might be false in the generality we are asking. So maybe the question could be restricted to covering and packing problems.

Open problems: Chapter 4

The first four open problems are all of the form "close the gap between the best positive and negative results. The interest in these problems depends on how much you like the problem and whether or not you can identify the optimal ratio. I personally am less interested in Open problem 2 since I am less interested in the paid model. Just a matter of taste.

Open problem 5 is a significant problem given the gap and the interest in the Steiner tree problem.

Open problems: Chapter 5

Open problems 1 and 2 both concern the randomized k -server conjecture which we might state as “Does there exist a $\log^2 k$ (or even $\text{polylog}(k)$) randomized competitive ratio for every metric space. The paper stating that the “ k server conjecture is false” was for the conjecture that there exists a $O(\log k)$ randomized competitive algorithm for all metric spaces.

In particular, we do not seem to know if there is a $o(k)$ randomized competitive algorithm for the infinite line.

Open question 3 has been solved as mentioned last week. This is a big deal given the important role that the k -server conjecture has played in the development of online algorithms.

I am personally not that interested in Open question 4. Again, this is a matter of taste. And whenever I say taste, I should emphasize that if the method of proof is novel then I would be very interested especially if the method might be applicable more generally. .

Open problems: Chapter 6

Online learning is an area of research I am not that familiar with and here I would first suggest trying to find out what might be the most interesting questions for expert learning and bandit problems. You can ask an LLM what it thinks are the most interesting open questions.

Again since I am not as certain about the status of problem in this area, it could be that Open questions 1 and 2 are already solved,

Open question 3 extends the bandits framework to the random order model discussed in Volume 2. But it is a natural extension that again I wonder if it has been looked at and, if not, why not?

Open problems: Chapter 7

This chapter contains a number of “open problems” but of course I mean were open as far as we knew before ChapGPT and other LLMs changed the world of mathematics and TCS.

Open problem 1 was solved in August 2026. “A Tight Bound on Online Vertex Cover under Edge Arrivals” by Tang and Zhang

I think the other open problems in Chapter 7 are still open. I like all these problems as they all involve reasonably substantial gaps in our knowledge with the exception of Open problem 6 where the gap is between 5 and 8 for the competitive ratio of the First Fit colouring algorithm for interval graphs. When we don't understand First Fit algorithms, it is already interesting.

Open problems: Chapter 7 continued

Open question 2 is an instance of the general question as to the tradeoff between the number of random bits (or more generally, the amount of entropy in the random source) vs performance (e.g., the competitive ratio). Here the specific question is the number of random bits vs the competitive ratio for the unweighted one-sided bipartite max matching (BMM) problem. In class, I will define the problem and give some of the history and importance of the unweighted and weighted BMM problem.

In particular, the randomized Karp, Vazirani and Vazirani [Ranking algorithm](#) is a seminal paper in competitive analysis.

Open problems: Chapter 8

Open problem 1 probably has not received that much attention. I haven't followed the literature on File Migration and File Replication problems but I would think the gaps in the competitive ratios are still open.

Multidimensional bin packing and vector bin packing are well studied problems which still have open problems and would be of interest.

The hard k -taxi problem remains one of the more prominent questions in competitive analysis.

The Weighted k -server problem is perhaps less studied but still an important problem in competitive analysis.

Open problems: Chapter 9

Online algorithms in social choice (eg fairness and voting) and mechanism design are relatively new areas of research. So I suspect all the problems here and many others in these areas are open problems.

I think Open problem 4 concerning the Parking problem is an interesting question. More generally, how can we better reflect real parking problems. In particular, people usually only park cars for a short duration. The k -taxi problem can also be extended to reflect the time duration involved in responding to a request.

The Parking problem is an instance of studying online algorithms where the decisions are being made by self-interested agents. Many online problems can be formulated so that self-interested agents are making decisions and the online mechanism must incentivize agents so that the overall “social welfare” is approximately optimized. Uber requests are also being decided by self-interested drivers.

Open problems: Chapter 10

Open problem 1 concerns what can be achieved when the total weight of clauses is not known,

Ali got ChapGPT to work on Open questions 2 and 3. I have to look at the ChapGPT paper. I have to see exactly what is being claimed in the ChapGPT paper which I will do. .

I believe Open problem 4 is still open if for no other reason that it is not precisely formulated. What do we mean by “simple online”?