CSC2421: Online and other myopic algorithms Spring 2021

Allan Borodin

February 4, 2021

Week 4: agenda

Our goal today is to try to converge as much as possible on initial projects. I was hoping that we could even hear some preliminary discussions next week. But that will require having one or two people doing some reading.

I will walk through the chapters of the text (many not even started) and indicate where I think there are interesting recent developments.

But choosing a more “classical topic” can still be interesting.

So I hope todays discussion will be helpful in choosing topics.

Last week, Alex and David had chat messages about the use of the Harmonic algorithm in some real world systems. Please raise hand or in any way please do interupt. I lost the chat message when I ended the zoom meeting and didn’t copy the chat. David mentioned SLAB and Xiaoxu also mentioned SLAB. Alex mentioned Hoard.

A recurring theme

One recurring theme is bridge the gap between theory and practice. There are different ways this theme is encounterd:

- Assume some partial information is available allowing for more optimistic results. This partial information can be assumed definite knowledge, knowledge that is not fully trusted, or stochastic assumptions.
- The online framework can be relaxed. For example the assumption that decisions are irreversible can be relaxed or one can argue for less powerful adversaries (e.g., the random order model).
- Alternative measures other than the competitive ratio.
- New applications are defined to better model real world applications. One prominent example are models relating to online advertising and other examples of online matching.

We can keep these themes in mind as we walk through the chapters. If anything seems like a potential topic then please stop me and we can elaborate. I will elaborate on some things close to my current research interests.

Chapters 1-3

We already skimmed the content in Chapters 1,2,3.

- I didn't have time last week to say more about the Yao principle for establishing lower bounds on randomized algorithms. This method is used in both online and offline algorithms. For offline the method is used for establishing time bounds (in restricted models) and in online algorithms we use it for establishing competitive ratio inapproximations as for example in showing that $H_k \approx \ln k$ is the best possible ratio for paging/caching .
- In later chapters we will see extensions of ski rental and caching (to address other settings for virtual memory) as well as motivating the interest in alternative measures (e.g., to distinguish between different paging algorithms.)

Chapter 4: “classical” problems

Of course nothing in this topic is classical by the standards of say mathematics. What I mean is problems that have been studied since say the start of active TCS interest in online algorithms (say 1985).

- The list accessing problem was in the seminal Sleator and Tarjan paper. The competitive ratio for MTF is established by the potential function method. This is a common method used in data structure analysis and other areas of algorithm analysis. Understanding potential funcitions is an art form and there is no compelling approach to how to create appropriate potential functions for a given application.
- Soon after the Sleator and Tarjan paper, two abstract problem formulations were defined, namely first the MTS formulation and then the k-server problem. Both problems are still actively studied. The current main interest is with regard to establishing the best randomized algorithm for both problems.

The lasting influence of MTS and k server problems.

- The MTS and k-serever problems led to Bartal's concept of Hierarchical Separated Trees (a randomized class of metric spaces) which allows one to embed arbitrary metric spaces into HSTs. This has found applications outside of online algorithms.
- The current approach to randomized algorithms for the MTS and k-server problems is to first obtain a bound for HSTs and then to embed a given metric spece into HSTs. The optimal embedding incurs a cost of $O(\log n)$ where n is the size of the metric space.
- This makes the analysis more challenging for the k serever problem where there is a conjecture that there is an $O(\log k)$ randomized algorithm for every metric space (i.e. independent of n). There is now a $O(\log^6 k)$ algorithm.
- This has led to connections between online algorithms analysis and online learning and convex optimization.

What else is in chapter 4

- The Koutsoupias and Papadimitriou [1995] work function algorithm is given and this is the latest word on the deterministic k-server conjecture provding a $2k$ competitve ratio which "nearly" solves the k-server conjecture which states that k is the optimal bound for all metric spaces.
- Other makespan models are presented, namely the related machines, the restricted machines and the ubnrelated machines models.
- Call admission and routing problems

Chapter 5: graph problems

We know that many (most?) otpimization problems for graphs become hard (in the sense of NP hardness) to approximate wrt arbitrary graphs.

This does not imply that we could not obtain (an inefficient) online algorithm with a good competitive ratio. But in fact these NP hardness results are generally speaking matched by corresponding competitive rastio inapproximations. For the vertex cover problem we can obtain a 2-approximation result matching a known hardness conjecture.

The field of graph theory and graph algorithms focuses on classses of graphs (e.g. planar graphs, interval graphs, trees, etc.)

For graph problems there are reasonable alternative ways to say what is the input model. We discuss the common input models.

The graph colouring problem is interesting for different graph classes.

The online bipartite matching problem

The online graph problem that has attracted the most attention is probably the online bipartite matching problem and vafriants of this problem.

This is due to the fact the bipartitie matching is the basic problem underlying many variants that are aimed at online advertising.

Another reason is the surprising seminal result for unweighted online bipartite matching, the randomized Ranking algorithm of Karp, Vazirani and Vazirani (KVV) algorithm that achieves competitibe ratio $1 - 1/e$. .

The Ranking algorithm can be extended to the case where each offline vertex has a weight. More generally we want to deal with edge weighted bipartite graphs graphs in the context of online advertising. But one cannot obtain constant bounds for online edge weighted bipartite matching if we are assuming adversarial input sequences.

We can obtain a $\frac{1}{e}$ ratio if we consider inputs in the random order model (ROM). This is done by extending the secretary problem algorithm.

Online advertising problems: adwords

Consider online vertices as queries (i.e. targets for ads based on these queries) and offline vertices as advertisers. As each query arrives, advertisers bid on the query (i.e., have a weight w_j for an edge between advertiser v_j and the query u_i).

The goal now is not a matching but rather an assignment of online vertices where each offline vertex can be assigned many online vertices. Let U be the set of online vertices and V the offline vertices.

- The *adwords problem*: Each advertiser v_j has a budget B_j. Let $M \subseteq U \times V$ be the matching returned by an 'algorithm. The objective function to be maximized is $\sum_j \min\{B_j, \sum_{i:(u_i,v_j)\in M} w_j\}$. The standard assumption is the small bids assumption: $w_j << B_j$
- There is a less studied version with a hard budget constraint. That is, an ad cannot be matched with an advertiser if it would the added bid would exceed the budget.

Online advertsing problems: display ads

- In the *display ads (with free disposal)* problem, each advertiser is a *unit demaand buyer* meaning many ads can be assigned to an advertiser but the advertiser is only interested in the most valuable ad assigned. (More genrally, each advertiser v_j is interested in at most s_j items.)

The adwords and display ads are both special case of a *submodular welfare* problem. In this probliem each offline vertex v_j is associated with a monotone submodular function f_j. The objective is to maximize $\sum_j f_j(V_j)$ where V_j is the set of online vertices assigned to v_j

These problems has been studied from the perspective of worst case inputs (input sequences in adversarial or random order), and also when online inputs are generated $i.i.d$ from a known or unknown distribution.

Bipartite matching advertising problems

	ADV	ROM	Unknown i.i.d.	Known i.i.d.
Unweighted Bipartite Matching	$1 - 1/e$ (opt)[16]	0.696 [13]	0.696	0.7299 [1]
Offline-Vertex Weighted	$1 - 1/e$ (opt)[6]	0.6534 [8]	0.6534	0.7299 [1]
Edge Weighted with FD	0.5086 [4]	0.51 [15]	$1 - 1/e$	0.705
Edge Weighted without FD	0	$1/e$ (opt)[17]	$1/e$	0.705 [1]
AdWords	$\frac{1}{2}$	0.51 [15]	$1 - 1/e$	$1 - 1/e$
Submodular Welfare	$\frac{1}{2}$ (opt)[12]	0.5096 [2]	$1 - 1/e$ (opt)[12]	$1 - 1/e$

[Table due to Chris Karavasilis]

Chapter 6: Maximizing a submodular function and the Max-Sat problem

This chapter is based in some sense on one problem in a paper by Buchbinder et al. The problem is maximize a non monotone submodular set function (e.g., max cut, and max di-cut). Note that this makes sense with or without constraints on the allowable sets.

The Buchbinder et al algorithm is a conceptually simple and efficient randomized algorithm, the first to achieve a $\frac{1}{2}$ competitive ratio which is optimal for *arbitrary* non-monotone submodular functions. Optimal with regard to *NP* hardness and subexponential query complexity.

The randomized algiorithm utilizes a nice general idea, that of a *natural randomization* of a deterministic algorithm.

There is some evidence that the algorithm cannot be “de-randomized” to obtain the same $\frac{1}{2}$ ratio.

In the same paper, Buchbinder et al show how to extend their algorithm to obtain a $\frac{3}{4}$ ratio for the *submodular max-sat problem.*

Max-sat

The standard weighted Max-sat problem is a classic NP hard optimization problem and just one example of a constraint satisfaction problem. The naive randomized algorithm can be de-randomized to achieve a $\frac{2}{3}$ ratio.
Historical note It turns out that Johnsons' determistic algorithm was known since around 1974. After 15 years Yannakakis showed that the de-randomization of the naive algorithm becomes Johnson's algorithm.

Indpendently, Poloczek and Schnitger, and van Zuylen presented randomized online algorithms that achieve the same $\frac{3}{4}$ ratio. It turns out that the van Zuylen algorithm is equivalent to the Buchbinder et al algorithm when that algorithm is restricted to the standard weighted max-sat problem.

It is open if the $\frac{3}{4}$ competitve ratio can be improved.(The best known offline approximation ratio is .7968.) It is open if there is any determinstic online algorithm with ratio $\frac{3}{4}$. There is some evidence that such a deterministic online algorithm does not exist but based on a restricted (but still general enough to obtain the randomized $\frac{3}{4}$ ratio) input model.

Chapter 7: Recent progress

This chapter contains extensions to some of the classic problems, some new applications, and as well as (hopefully) a proof of the latest randomized k server results.

We present some extensions of the ski rental and bin packing

Related to the k-server is the k-taxi problem and the uber problem. We also consider some work related to page migfration and and other applications related to distributed computation.

The recent progress for the k-server results is based on convex optimization as developed in online learning. This is a very substantial development.