SIGACT News Online Algorithms Column 24: 2014 so far

Rob van Stee
University of Leicester
United Kingdom

In this column, I will discuss some recent papers in online algorithms. It is pleasing to see there were a number of papers about online algorithms in the top conferences this year. If I have missed your paper and you want to write about it or about any other topic in online algorithms, don't hesitate to contact me!

1 SODA

There is a long history of research into giving online algorithms extra power compared to the offline algorithms that they are compared to. This can help to give more insight into problems for which the strict competitive ratio is relatively high. Probably the best-known version of this approach is resource augmentation. For instance in machine scheduling, it is sometimes possible to achieve significantly better results by giving the online algorithm only slightly faster machines than the offline algorithm.

Another well-known approach is to tweak the revocability of the decisions of the online algorithm, giving algorithms the chance to make minor changes in each step instead of requiring them to make irrevocable decisions as the input arrives. Two papers on this topic appeared in this year's SODA.

Online Steiner tree with deletions

The Steiner tree problem is well known: given a metric space with a set of designated points (vertices), the goal is to connect these vertices using a tree of minimal cost (total edge length). Here it is allowed to add additional vertices, that is, not every edge must connect two of the designated vertices. In the online version of this problem, the vertices arrive one by one and need to be connected. The competitive ratio of this problem is $\Theta(\log n)$, which drops to $O(1)$ if we can also make one edge swap whenever a vertex arrives.

Gupta and Kumar [11] consider the case where vertices may also leave. In the case where there are only deletions, they give an $O(1)$-competitive algorithm using $O(1)$ changes per departure. The authors first provide an algorithm which uses $O(1)$ amortized changes per departure, and then carefully adjust it for the more challenging non-amortized setting.

They also give an algorithm for the fully dynamic case where vertices may both arrive and leave. This algorithm is different (also for deletions) and is slightly weaker: it uses $O(1)$ *amortized* changes per event to be $O(1)$-competitive.

Maintaining assignments online: matching, scheduling, and flows [12]

This paper, by the same authors as above, joined by Cliff Stein, achieves a very similar result as the first one: the competitive ratio of a problem is reduced from logarithmic to constant or doubly logarithmic using an (amortized) constant number of changes per step. The problem and the methods are quite different, however.

In the easiest version of the problem considered here, edges of a graph arrive online and need to be directed in order to maintain a low indegree for each vertex. Here a ratio of 2 can be achieved using $O(1)$ amortized changes per step. The paper also considers online matching. In the version considered here, the left vertices arrive online and must be matched to the right vertices. The authors get a constant competitive ratio, using $O(1)$ amortized rematchings of left vertices. This result can be extended to restricted machine scheduling at the cost of a slight increase in the competitive ratio.

Finally, a sink flow problem is considered, where each new vertex is a source that requires a path to the sink to send its flow. Rerouting flow along $O(F^*)$ edges, where F^* is the total length of the optimal flow paths, is sufficient to achieve a constant competitive ratio.

2 STOC

Online local learning via semidefinite programming

There was a very nice, quite short paper in STOC about online local learning [7]. It considers the problem of online local learning, where you are interested in getting local information rather than a global ranking of some universe of items. That is, each time you want to compare two items, and you will get a payoff depending on how good your comparison is. Paul Christiano shows that you can do this essentially as well as if you could solve the global optimization problem exactly, which is typically intractable. Here the performance measure is the amount of regret compared to the best fixed global ranking of the items.

Formally, each item has a constant number of possible labels, and the learner has to guess the labels of two items in each step. An example of such a problem is the online max-cut problem, where the learner has to output the probability that each new pair of vertices belongs to the same set or not.

Another example is online gambling, where in each round the learner has to put a probability distribution on the outcome of a head-to-head contest between a pair of teams before learning that outcome, and the goal is to get a payoff close to that of the best global ranking of all teams. However, in this example the number of possible labels is superconstant.

The paper uses a semidefinite relaxation for the space of labelings and solve an appropriately regularized problem. Using the log determinant regularizer improves the previous bound and simplifies the analysis, since it can be used as an estimate for the entropy given a matrix of moments. I feely admit, however, that all of this is far above my head and I hope it makes more sense to you than it does to me. If you are interested, I recommend reading the paper in full (all seven pages).

Primal beats dual on online packing LPs in the random-order model

In addition, there was a new paper [14] about the random-order model, where the adversary generates the instance but the requests of the instance appear in a random order. This is somewhat related to the matroid problems that were discussed in this space previously [8]. The authors were Thomas Kesselheim, Klaus Radke, Andreas Tönnis and Berthold Vöcking, and this is therefore one of the last ever papers coauthored by Berthold, who sadly passed away in June of this year. He has made many contributions to online algorithms (see also later in this column, in Section 4), and indeed to algorithms in general, and will be sorely missed. I met him many times at conferences, most recently at the annual summer school ADFOCS in 2012, where he was one of the lecturers. It was always a pleasure to meet him and it is strange to think that it will not happen again.

The paper [14] considers online packing linear programs. These are LPs where the right-hand side is fixed and the variables appear online. Each request reveals a column of the constraint matrix and the corresponding entries in the objective function, and the algorithm must select a column to allocate without violating any packing constraints. This models general problems where requests with profits arrive online and must be served immediately, consuming some resources, or be discarded. A prime example is online ad allocation.

The best known online algorithms for this problem use the primal-dual method, and this paper makes no exception. All these algorithms are $(1-\epsilon)$-competitive given enough capacity. This is expressed by the capacity ratio B, which is the ratio between the capacity of a resource and the maximum demand for it. Previously, it was shown that B must be $\Omega(\frac{1}{\epsilon^2}\log m)$ in order to be able to be $(1-\epsilon)$-competitive, but the best known algorithm unfortunately required exponentially larger values of B.

This paper closes the gap by providing a $(1-\epsilon)$-competitive for all $B = \Omega\frac{1}{\epsilon^2}\log d)$. Here d is the column sparsity, which is the maximum number of resources that occur in a single column. The algorithm requires no prior knowledge about B or d or any other information about the instance apart from the capacity vector and the number of requests n. This is in contrast to all previous algorithms.

3 SWAT

July was computer science month in Copenhagen, with SEA (starting on June 29th) immediately followed by SWAT and ICALP, with a one-day workshop on online algorithms (TOLA) in-between. I would have liked to attend TOLA, but for complicated personal reasons it was quite impossible! There were several nice papers there and they even accepted one of mine [18] which I may discuss in some future column.

Online makespan minimization with parallel schedules

Albers and Hellwig [2] consider the classical problem of minimizing the makespan on parallel (identical) machines, with a twist to the model. They assume that it is possible to maintain parallel schedules and select the best schedule at the end. This can be seen as another form of resource augmentation (see above), in this case giving the online algorithm a power which does not really exist in an offline environment.

They achieve a competitive ratio of $4/3 + \varepsilon$ using a constant number of parallel schedules (a function of ε). This is the best that can be achieved using a sublinear amount of schedules. Using polynomially many schedules, they show that it is even possible to achieve a competitive ratio of $1 + \varepsilon$.

Competitive online routing on Delaunay triangulations

Bose et al. [6] consider the problem of finding a shortest path online. That is, a path from s to t is constructed by taking for every intermediate node only decisions based on local information. They consider this problem on geometric graphs in the plane consisting of line segments (edges) and points (nodes), where the weight of each edge is its length. In their model, a message m with header h originates at s and needs to be sent to t. Each intermediate node u selects a neighbor to whom to forward m, and the goal is to minimize the total length traveled by m.

At any time, the header h only contains the coordinates of the source, the sink, and the node that sent the message to u, and possibly one additional value that is computed from distances between vertices visited by the message and may be modified by the algorithm during computation. The authors improve the best known competitive ratio for routing on Delaunay triangulations from 45.7 to 15.5. For points that are in convex position, they achieve 7.6. They use ideas from Bose and Morin [5] as well as several other papers to achieve this result.

4 ICALP

Online stochastic reordering buffer scheduling

The first ICALP paper I want to discuss is again about the random-order model. Esfandiari et. al [9] considers the classical online buffer scheduling problem. In this problem [17], items with colors arrive online, to be processed by a machine with a buffer of size k. Each item must be processed by switching the machine to its color. There are two models: in the standard model, the machine remains in a certain color until the buffer is full of items of different colors, and in the block operation model, the machine only serves the items which are currently in the buffer before the next color switch.

Esfandiari et al [9] present constant competitive online algorithms for both these models, which cannot be achieved in the adversarial setting. They also show that in fact, in this random order setting, the two models are in a sense equivalent in that getting a constant competitive algorithm for one of them will also give a constant competitive algorithm for the other one. This is the first time such a result has been shown.

Finally, they show that in case the input comes from a uniform distribution, a simple greedy algorithm is the best online algorithm in both models. Famously, greedy algorithms do not perform at all well in the adversarial setting, and it was a longstanding open problem whether a constant competitive ratio could be achieved, until the problem was finally resolved recently by Avigdor-Elgrabli and Rabani [4] and Adamaszek et al. [1].

For a long time, this was one of those rare problems where the best known approximation ratio is achieved by an online algorithm. This also came to an end recently when Avigdor-Elgrabli and Rabani [3] presented an approximation algorithm with constant approximation ratio for this problem.

On a personal note, I can't say how relieved I am that this problem is now resolved, because it means that I can stop spending time on it! The problem had interested me ever since I first heard of it at ESA 2002, but I never made any progress.

Online independent set beyond the worst-case: secretaries, prophets, and periods

A slightly different approach was considered by Göbel et al. [10]. The authors also use a stochastic analysis, but do not focus on any specific stochastic input model. Instead, they use a sampling approach, which generalizes both the secretary model and the prophet-inequality model.

In the *graph sampling model* which the authors introduce, the only algorithm initially has a sample graph that is stochastically similar to the actual input graph, which will be revealed in an online fashion. Formally, for a given graph, the input graph and the sample graph are both derived by drawing nonnegative weights at random for each node.

Importantly, the weights do not need to be drawn according to exaclty the same probability distributions, but the distributions should be "similar" (this is made formal in the paper), and the weight of any node in either graph may not depend on any other weight. The two weights of one node need *not* be independent but may be correlated, thus covering the secretary model. The nodes are then presented to the online algorithm in an adversarial order.

The result is a constant competitive online algorithm for the maximum independent set problem on interval and disk graphs. These problems have strict competitive ratios which are $\Omega(n)$. This algorithm is inspired by Korula and Pál [15].

The paper also considers weighted independent set, for which an online algorithm cannot be competitive at all in the classical sense. It achieves near-optimal results for various models (secretary, prophet inequality, period, unifying). The results are functions of the inductive independence number ρ of a graph, which is the smallest number for which there is an order such that for any independent set $S \subseteq V$ and any $v \in V$, the number of nodes in S succeeding v (in this order) and connected to it is at most ρ.

Near-optimal online algorithms for prize-collecting Steiner problems

HajiAghayi et al. [13] presents another result for online Steiner problems (see also above). In the prize-collecting version of these problems, not all nodes need to be connected. Every node comes with its own penalty which needs to be paid in case the node is not connected. The goal of an algorithm is to minimize the sum of the total connection cost and the total penalty paid.

The authors begin by showing how to reduce online prize-collecting Steiner tree problems to their fractional non-prize-collecting versions while losing only a logarithmic factor in the competitive ratio. They improve on these results using an online dual-fitting approach, getting a competitive ratio of $O(\log n)$ for the edge-weighted version and $O(\log^3 n)$ for the node-weighted version. Both results hold even against a fractional adversary.

For the node-weighted version, no poly-logarithmic competitive ratio was known previously. In contrast, a competitive ratio of $O(\log n)$ for the edge-weighted version had been achieved already before by Qian and Williamson [16]; the contribution of the present paper is to present a new method with a (much) simpler analysis for this problem.

References

[1] Anna Adamaszek, Artur Czumaj, Matthias Englert and Harald Räcke. Optimal online buffer scheduling for block devices. STOC 2012:589–598.

[2] Susanne Albers and Matthias Hellwig. Online Makespan Minimization with Parallel Schedules. SWAT 2014:13–25.

[3] Noa Avigdor-Elgrabli and Yuval Rabani. A Constant Factor Approximation Algorithm for Reordering Buffer Management. SODA 2013:973–984.

[4] Noa Avigdor-Elgrabli and Yuval Rabani. An improved competitive algorithm for reordering buffer management. FOCS 2013:1–10.

[5] Prosenjit Bose and Pat Morin. Online Routing in Triangulations. *SIAM J. Comput.* 33(4):937–951, 2004.

[6] Prosenjit Bose, Jean-Lou De Carufel, Stephane Durocher and Perouz Taslakian. Competitive Online Routing on Delaunay Triangulations. SWAT 2014:98–109.

[7] Paul Christiano. Online local learning via semidefinite programming. STOC 2014:468–474.

[8] Michael Dinitz. Recent advances on the matroid secretary problem. *SIGACT News* 44(2):126–142, 2013.

[9] Hossein Esfandiari, MohammadTaghi HajiAghayi, Mohammad Reza Khani, Vahid Liaghat, Hamid Mahini and Harald Räcke. Online stochastic reordering buffer scheduling. ICALP 2014:465–476.

[10] Oliver Göbel, Martin Hoefer, Thomas Kesselheim, Thomas Schleiden and Berthold Vöcking. Online independent set beyond the worst-case: secretaries, prophets, and periods. ICALP 2014:508–519.

[11] Anupam Gupta and Amit Kumar. Online Steiner tree with deletions. SODA 2014:455–467.

[12] Anupam Gupta, Amit Kumar and Cliff Stein. Maintaining assignments online: matching, scheduling, and flows. SODA 2014:468–479.

[13] MohammadTaghi HajiAghayi, Vahid Liaghat and Debmalya Panigrahi. Near-optimal online algorithms for prize-collecting Steiner problems. ICALP 2014:576–587.

[14] Thomas Kesselheim, Andreas Tönnis, Klaus Radke and Berthold Vcking. Primal beats dual on online packing LPs in the random-order model. STOC 2014:303–312.

[15] Nitish Korula and Martin Pál. Algorithms for Secretary Problems on Graphs and Hypergraphs. ICALP 2009:508–520.

[16] Jiawei Qian and David P. Williamson. An $O(\log n)$-Competitive Algorithm for Online Constrained Forest Problems. ICALP 2011:37–48.

[17] Harald Räcke, Christian Sohler and Matthias Westermann. Online scheduling for sorting buffers. ESA 2002:820–832.

[18] Martin Böhm, Jiří Sgall, Rob van Stee, Pavel Veselý. Better algorithms for online bin stretching. To appear in WAOA 2014.