# CSC2420 Fall 2012: Algorithm Design, Analysis and Theory

Allan Borodin

November 29, 2012; Lecture 11

# Miscellaneous topics to end the term

Clearly "algorithm design, analysis, and theory" is an extremely broad subject (and one might say it is much of what CS does) so we have only discussed a few topics and even then only discussed them briefly.

As promised or threatened last class, here are a few topics with which we will conclude the course:

1. An optimal randomized algorithm for the unconstrained submodular maximization problem.
2. Matroids and comments on the monotone submodular maximization problem (subject to a constraint).
3. A return to non-oblivious local search and its power.
    1. The Khanna et al non-oblivious algorithm for Max-$k$-Sat
    2. The Filmus and Ward non-oblivious local search for monotone submodular maximization subject to a matroid constraint.
    3. The Berman WMIS algorithm for $k+1$ claw free graphs.
4. The constructive Lovász Local Lemma for 3SAT when variables do not appear in too many clauses.
5. Spectral methods: We will not get to this but you could look at the various notes that can be found on the web.

# Submodular maximization problems

- A set function $f : 2^U \rightarrow \Re$ is submodular if $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$ for all $S, T \subseteq U$.
- Equivalently, $f$ is submodular if it satisfies decreasing marginal gains; that is, $f(S \cup \{x\}) - f(S) \geq f(T \cup \{x\}) - f(T)$ for all $S \subseteq T \subseteq U$ and $x \in U$
- We will always assume that $f$ is *normalized* in that $f(\varnothing) = 0$.
- Submodular functions arise naturally in many applications and has been a toipic of much recent activity.
- Probably the most frequent application of (and papers about) submodular functions is when the function is also monotone (non-decreasing) in that $f(S) \leq f(T)$ for $S \subseteq T$.
- Note that linear functions (also called modular) functions are a special case of monotone submodular functions.
  Aside: Any linear function maximiation problem can be reformulated more generally as a (monotone) submodular function problem.

# Submodular maximization

In the submodular maximization problem, we want to compute $S$ so as to maximize $f(S)$.

- For monotone functions, we are maximizing $f(S)$ subject to some constraint (otherwise just choose $S = U$).
- For the non monotone case, the problem is already interesting in the unconstrained case. Perhaps the most prominent example of such a problem is Max-Cut (and Max-Di-Cut).
- Max-Cut is an NP-hard problem. Using an SDP approach just as in the Max-2-Sat problem yields the same approximation ratio $\alpha = \frac{2}{\pi} \min_{\{0 \leq \theta \leq \pi\}} \frac{\theta}{(1-cos(\theta))} \approx .87856$. Assuming UGC, this is optimal.
- For a submodular function, we may be given an explicit representation (when a succinct representation is possible as in Max-Cut) or we access the function by an oracle such as the *value oracle* which given $S$, outputs the value $f(S)$ and such an oracle call is considered to have $O(1)$ cost. Other oracles are possible (e.g. given $S$, output the element $x$ of $U$ that maximizes $f(S \cup \{x\}) - f(S)$).

# Unconstrained (non monotone) submodular maximization

- Feige, Mirrokni and Vondrak [2007] began the study of approximation algorithms for the unconstrained non monotone submodular maximization (USM) problem establishing several results:
    1. Choosing $S$ uniformly at random provides a $1/4$ approximation.
    2. An oblivious local search algorithm results in a $1/3$ approximation.
    3. A non-oblivious local search algorithm results in a $2/5$ approximation.
    4. Any algorithm using only value oracle calls, must use an exponential number of calls to achieve an approximation $(1/2 + \epsilon)$ for any $\epsilon > 0$.
- The Feige et al paper was followed up by improved local search algorithms by Gharan and Vondrak [2011] and Feldman et al [2012] yielding (respectively) approximation ratios of .41 and .42.
- The $(1/2 + \epsilon)$ inapproximation was augmented by Dobzinski and Vondrak showing the same bound for an explicitly given instance under the assumption that $RP \neq NP$.

# The Buchbinder et al (1/3) and (1/2) approximations for USM

In this years FOCS [2012] conference, Buchbinder et al give an elegant linear time deterministic 1/3 approximation and then extend that to a randomized 1/2 approximization. The conceptually simple form of the algorithm is (to me) as interesting as the optimality (subject to the proven inapproximation results) of the result. Let $U = u_1, \ldots u_n$ be the elements of $U$ *in any order*.

## The deterministic 1/3 approximation for USM

$X_0 := \varnothing$; $Y_0 := U$

For $i := 1 \ldots n$

  $a_i := f(X_{i-1} \cup \{u_i\}) - f(X_{i-1})$; $b_i := f(Y_{i-1} \setminus \{u_i\}) - f(Y_{i-1})$

  **If** $a_i \geq b_i$

    **then** $X_i := X_{i-1} \cup \{u_i\}$; $Y_i := Y_{i-1}$

    **else** $X_i := X_{i-1}$; $Y_i := Y_{i-1} \setminus \{u_i\}$

  **End If**

**End For**

# The randomized 1/2 approximation for USM

- Notwithstanding the previous "debate" as to whether or not an algorithm is being improved by the natural randomization of a deterministic algorithm, Buchbinder et al show that the "natural randomization" of the previous deterministic algorithm achieves approximation ratio $1/2$.
- That is, the algorithm chooses to either add $\{u_i\}$ to $X_{i-1}$ with probability $\frac{a'_i}{a'_i+b'_i}$ or to delete $\{u_i\}$ from $Y_{i-1}$ with probability $\frac{b'_i}{a'_i+b'_i}$ where $a'_i = \max\{a_i, 0\}$ and $b'_i = \max\{b_i, 0\}$.
- If $a_i = b_i = 0$ then add $\{u_i\}$ to $X_{i-1}$.
- Note: Part of the proof for both the deterministic and randomized algorithms is the fact that $a_i + b_i \geq 0$.

# Monotone submodular function maximization

- As previously mentioned, the monotone problem is only interesting when the submodular maximization is subject to some constraint.
- Probably the simplest and most widely used constraint is a cardinality constraint; namely, to maximize $f(S)$ subject to $|S| \leq k$ for some $k$ and since $f$ is monotone this is the same as the constraint $f(S) = k$.
- Following Cornuéjols, Fisher and Nemhauser [1977] (who study a specific submodular function), Nemhauser, Wolsey and Fisher [1978] show that the standard greedy algorithm achieves a $1 - \frac{1}{e}$ approximation for the cardinality constrained monotone problem. More precisely, for all $k$, the standard greedy is a $1 - (1 - \frac{1}{k})^k$ approximation for a cardinality $k$ constraint.

**Standard greedy algorithm**

$S := \varnothing$
**While** $|S| < k$
  Let $u$ maximize $f(S \cup \{u\}) - f(S)$
  $S := S \cup \{u\}$
**End While**

# Generalizing to a matroid constraint

- Nemhauser and Wolsey [1978] showed that the $1 - \frac{1}{e}$ approximation is optimal in the sense that an exponential number of value oracle queries would be needed to beat the bound for the cardinalily constraint.
- Furthermore, Feige [1998] shows it is NP hard to beat this bound even for the explicitly represented maximum $k$-coverage problem.
- Following their first paper, Fisher, Nemhauser and Wolsey [1978] extended the cardinality constraint to a matroid constaint. Matroids are an elegant abstraction of independence in a variety of settings.
- Fisher, Nemhauser and Wolsey show that both the standard greedy algorithm and the 1-exchange local search algorithm achieve a $\frac{1}{2}$ approximation for an arbitrary matroid constraint.
- They also showed that this bound was tight for greedy and for the 1-exchange local search.

# Matroids and independence systems

- Let $M = (U, \mathcal{F})$, where U is a set of elements, $\mathcal{F} \subseteq 2^{|U|}$; $I \in \mathcal{F}$ is called an independent set.
  An independence system satisfies the following properties:
  1) $\varnothing \in \mathcal{F}$; often stated although not necessary if $\mathcal{F} \neq \varnothing$
  2) $S \subseteq T, T \in \mathcal{F} \Rightarrow S \in \mathcal{F}$
- A matroid is an independence system that also satisfies:
  3) $S, T \in \mathcal{F}, |S| < |T|$, then $\exists x \in T \setminus S$ such that $S \cup \{x\} \in \mathcal{F}$
- Sets having at most $k$ elements constitute the independent sets in a uniform matroid
- Other common examples, include
  1. partition matroids where $U$ is the disjoint union $U_1 \cup U_2 \ldots \cup U_r$ and there are individual cardiality constraints $k_i$ for each block $U_i$ of the partition.
  2. Graphic matroids where $U$ is the set of edges $E$ in a graph $G = (V, E)$ and $E' \subseteq E$ is independent if $G = (V, E')$ is acyclic.
  3. Linear matroids where $U$ is a set of vectors in a vector space and $I$ is independent in the usualy sense of linear independence.

# Achieving the $1 - \frac{1}{e}$ approximation for arbitrary matroids

- An open problem for 30 years was to see if the $1 - \frac{1}{e}$ approximation for the cardinality constraint could be obtained for arbittrary matroids.
- Calinsecu et al [2007, 2011] positively answer this open problem using a very different (than anything in our course) algorithm consiting of a continuous greedy algorithm phase followed by a pipage rounding phase.
- Following Calinsecu et al, Filmus and Ward [2012A, 2012B] develop (using LP analysis to guide the development) a sophisticated non-oblivious local search algorithm that is also able to match the $1 - \frac{1}{e}$ bound, first for the maximum coverage problem and then for arbitrary monotone submodular functions.

# The power of non-oblivious local search

In Lecture 4 we very briefly mentioned non-oblivious local search. As stated, for some applications it turns to be beneficial to search for a solution (in the local neighbourhood) that improves a related potential function rather than the given objective function. This has been termed non-oblivious local search.

- The first place I encountered non-oblivious local search was in Khanna et al [1994,1998] where they considered the weighted exact Max-$k$-Sat problem and other CSP problems. Independently Alimonti [1994,1995,1997] utlilized the same non-oblivious local search and terminology for various problems.
- Khanna et al show that
  1. The 1-flip (flip the truth assignment of one variable) oblivous local search achieves a locality gap (and really a totality ratio as defined in Lecture 8) of $\frac{k}{k+1}$.
  2. For (unweighted) Max-2-Sat, this ratio is tight even for $r$-flip oblivous local search for any $r = o(n)$.
  3. For every $k$, there is a non-oblvious 1-flip local search algorithm that achieves totality ratio $\frac{2^k-1}{2^k}$, the same ratio achieved by Johnson's algorithm (and the the naive randomized algorithm)

# The inutition for the non-oblivious local search

- When there are two more different truth assignments which achieve (say roughly) the same objective value, which if any truth assignment should you prefer?
- Intuitively clauses satisfied by two literals are better than those satisifed by one literal. So thinking of the objective of (weighted) exact Max-2-Sat as being the weight $W_1$ of clauses satisfied by one literal plus the weight $W_2$ of clauses satisfied by two literals, we are led to a potential function $\alpha_1 W_1 + \alpha_2 W_2$ or $W_1 + \alpha W_2$ for some $\alpha > 1$.
- By considering the proof of the totality ratio, we can see that the best choice of $(\alpha_1, \alpha_2) = (3/2, 2)$ or $\alpha$ is $\frac{4}{3}$ leading to a totality ratio of $3/4$. More generally, there is a choice of scaling factors $\alpha_i$ (for clauses satisfied by $i$ literals) for the exact Max-$k$-Sat problem that yields the stated ratio.
- In this and other results we are often omitiing discussion of approximate local optimal that introduces an $\epsilon$ into the results which can then usually be removed in standard ways.

# Another application of non-oblivious local search: weighted max coverage

**The weighted max coverage problem**

Given: A universe $E$, a weight function $w : E \rightarrow \Re^{\geq 0}$ and a collection of of subsets $\mathcal{F} = \{F_1, \ldots, F_n\}$ of $U$. The goal is to find a subset of indices $S$ (subject to a matroid constraint) so as to maximize $f(S) = w(\cup_{i \in S} F_i)$ subject to some constraint (often defined as the cardinality constraint).
Note: $f$ is a monotone submodular function.

- In matroid, all maximal independent sets have the same size; the rank of a matroid is the size of the largest maximal independent set. Conversely, if all maximal independent sets in an independence system $M$ have the same size, then $M$ is a matroid.
- For $\ell < r = rank(M)$, the $\ell$-flip oblivious local search for max coverage has locality gap $\frac{r-1}{2r-\ell-1} \rightarrow \frac{1}{2}$ as $r$ increases. (Recall that greedy achieves $\frac{1}{2}$.)

# The non-oblivious local search for max coverage

- Given two solutions $S_1$ and $S_2$ with the same value for the objective, we again ask, when one solution is better than the other?
- Similar to the motivation used in Max-$k$-Sat, solutions where various elements are covered by many sets is intuitively better so we are led to a potential function of the form $g(S) = \sum \alpha_{\kappa(u,S)} w(u)$ where $\kappa(u, S)$ is the number of sets $F_i$ ($i \in S$) such that $u \in F_i$ and $\alpha : \{0, 1, \ldots, r\} \rightarrow \Re^{\geq 0}$.
- The interesting and non-trivial development is in defining the appropriate scaling functions $\{\alpha_i\}$ for $i = 0, 1, \ldots r$
- Filmus and Ward derive the following recurrence for the choice of the $\{\alpha_i\} : \alpha_0 = 0, \alpha_1 = 1 - -\frac{1}{e}$, and $\alpha_{i+1} = (i + 1)\alpha_i - i\alpha_{i-1} - \frac{1}{e}$.

# The very high level idea and the locality gap

- The high-level idea behind the derivation is like the factor revealing LP used by Jain et al [2003]; namely, they formulate an LP for an instance of rank $r$ that determines the best obtainable ratio (by this approach) and the $\{\alpha_i\}$ obtaining this ratio.

**The Filmus-Ward locality gap for the non oblivious local search**

The 1-flip non oblivious local search has locality gap $O(1 - \frac{1}{e} - \epsilon)$ and runs in time $O(\epsilon^{-1} r^2 |\mathcal{F}||U| \log r)$
The $\epsilon$ in the ratio can be removed using partial enumeration resulting in time $O(r^3 |\mathcal{F}|^2 |U|^2 \log r)$.

# A non oblivious local search for an arbitrary monotone submodular function

- The previous development and the analysis needed to obtain the bounds is technically involved but is aided by having the explicit weight values for each $F_i$. For a general monotone submodular function we no longer have these weights.
- Instead, Filmus and Ward define a potential function $g$ that gives extra weight to solutions that contain a large number of good sub-solutions, or equivalently, remain good solutions on average even when elements are randomly removed.
- A weight is given to the average value of all solutions obtained from a solution $S$ by deleting $i$ elements and this corresponds roughly to the extra weight given to elements covered $i+1$ times in the max coverage case.
- The potential function is :
  $g(S) = \sum_{k=0}^{|S|} \sum_{T:T\subseteq S,|T|=k} \frac{\beta_k^{(|S|)}}{\binom{|S|}{k}} f(T) = \sum_{k=0}^{|S|} \beta_k^{(|S|)} \mathbf{E}_T[f(T)]$

# One more non oblivious local search

- We consider the weighted max (independent) vertex set in a $k+1$ claw free graph. Note that this is the standard graph theoretic notion of an indepedent set of vertices and this is not independence in a matroid. The problem is that of finding an independent set $S$ of vertices so as to maximize a linear function $f(S)$ (i.e. weights given to vertices).
- The concept of an independent set in a $k+1$ claw free graph has been abstracted by Feldman et al [2011] to an independence system called $k$-exchange systems which are a proper subcase of Mestre's [2006] $k$-extendible systems which are a subcase of Jenkyn's [1976] $k$ systems.
- The work of Jenkyns and Nemhauser et al show that the standard greedy algorithm is a $\frac{1}{k}$ approximation for weighted max independent set in a Jenkyn's $k$ system.
- It remains an open problem to improve upon the greedy approximation for Mestre's $k$ extendible systems and Jenkyn's $k$ systems.

# Oblivious and non-oblivious local search for $k+1$ claw free graphs

- The standard greedy algorithm and the 1-swap oblivious local search both achieve a $\frac{1}{k}$ approximation for the WMIS in $k+1$ claw free graphs. Here we define an "$\ell$-swap" oblivous local search by using neighbrourhoods defined by bringing in a set $S$ of up to $\ell$ vertices and removing all vertices adjacent to $S$.
- The standard greedy and 1-swap oblivious local search can be extended to the case of submodular (rather than linear) functions on the vertex sets. This results in a $\frac{1}{k+1}$ approximation (locality gap). The idea is to use marginal gain of an element (relative to the current solution).
- For the unweighted MIS, Halldórsson shows that a a 2-swap oblivious local search will yield a $\frac{2}{k+1}$ approximation.

# Berman's [2000] non-oblivious local search

- For the weighted MIS, the "$\ell$-swap" oblivous local search results (essentially) in an $\frac{1}{k}$ locality gap for any constant $\ell$.
- Chandra and Halldórson [1999] show that by first using a standard greedy algorithm to initialize a solution and then uses a "greedy" $k$-swap oblivious local search improves the approximation ratio to $\frac{3}{2k}$.
- Can we use non-oblivious local search to improve the locality gap? Once again given two solutions $V_1$ and $V_2$ having the same weight, when is one better than the other?
- Intuitively, if one vertex set $V_1$ is small but vertices in $V_1$ have large weights that is better than a solution
- Berman chooses the potential function $g(S) = \sum_{v \in S} w(v)^2$. Ignoring some small $\epsilon$'s, his $k$-swap non-oblivious local search achieves a locality gap of $\frac{2}{k+1}$ for WMIS on $k+1$ claw-free graphs.
- Linear function (resp. monotone submodular) maximization is extended to $k$ exchangeable systems in Feldman et al [2011] (resp. Ward [2012]). Note: For the submodular case, the potential function introduces some obstacles in using the marginal weight.

# A constructive Lovász Local Lemma for 3-SAT

- Suppose we have a set of random events $E_1, \ldots, E_m$ with $Prob[E_i] \leq p < 1$ for each $i$. Then if these events are independent we can easily bound the probability that none of the events has occurred; namely, it is $(1-p)^m > 0$.
- Suppose now that these events are not independent but rather just have limited dependence. Namely suppose that each $E_i$ is dependent on at most $r$ other events. Then the Lovász local Lemma (LLL) states that if $e \cdot p \cdot (r+1)$ is at most 1, then there is a non zero probability that none of the events $E_i$ occurred.
- As stated this is a non-constructive result

# A somewhat canonical application of the LLL

- Let $F = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ be a an exact $k$ CNF formula. From our previous discussion of the exact Max-$k$-Sat problem and the naive randomized algorith, we know that if $m < 2^k$, then $F$ must be satisfiable.
- Suppose instead that we have an arbitrary number of clauses but now for each clause $C$, at most $r$ other clauses share a variable with $C$.
- If we let $E_i$ denote the event that $C_i$ is not satisfied for a random uniform assignment (and hence having probability $1/(2^k)$), then we are interested in having a non zero probability that none of the $E_i$ occurred (i.e. that $F$ is satisfiable).
- The LLL tells us that if $r + 1 \leq \frac{2^k}{e}$, then $F$ is satisfiable.
- As nicely stated in Gebauer et al [2009]: "In an unsatisable CNF formula, clauses have to interleave the larger the clauses, the more interleaving is required."

# A constructive algorithm for the previous proof of satisfiability

- Here we will follow a somewhat weaker version (for $r \leq 2^k/8$) proven by Moser [2009] and then improved by Moser and G. Tardos [2010] to give the tight LLL bound. This proof was succinctly explained in a blog by Lance Fortnow
- This is a constructive proof in that there is a randomized algorithm (which can be de-randomized) that with high probability (given the limited dependence) will terminate and produce a satisfying assignment in $O(mlogm)$ evaluations of the formula.
- Both the algorithm and the analysis are very elegant. The algorithm is in essence a local search search algorithm and it seems that this kind of analysis (an information theoretic argument using Kolmogorov complexity to bound convergence) should be more widely applicable.

# The Moser algorithm

We are given an exact $k$-CNF formula $F$ with $m$ variables such that for every clause $C$, at most $r \leq 2^k/8$ other clasues share a variable with $C$.

### Algorithm for finding a satisfying truth assignment

Procedure SOLVE

  Let $\tau$ be a random assignment

  **While** there is a clause $C$ not satisfied

    Call FIX(C)

  **End While**

Procedure FIX(C)

  **While** there is a neighbouring unsatisfied clause $D$

    Randomly set all the variables occuring in $D$

    Call FIX(D)

  **End While**