Approximating Incremental Combinatorial Optimization Problems

Michel X. Goemans[1] and Francisco Unda[2]

1   MIT, Cambridge, MA, USA
     goemans@math.mit.edu
2   MIT, Cambridge, MA, USA
     funda@mit.edu

────── **Abstract** ──────

We consider incremental combinatorial optimization problems, in which a solution is constructed incrementally over time, and the goal is to optimize not the value of the final solution but the average value over all timesteps. We consider a natural algorithm of moving towards a global optimum solution as quickly as possible. We show that this algorithm provides an approximation guarantee of $(9 + \sqrt{21})/15 > 0.9$ for a large class of incremental combinatorial optimization problems defined axiomatically, which includes (bipartite and non-bipartite) matchings, matroid intersections, and stable sets in claw-free graphs. Furthermore, our analysis is tight.

1   Introduction

Usually, in the context of combinatorial optimization, a single solution is sought which optimizes a given objective function. This for example could be designing (or upgrading) a network satisfying certain properties. But the solution might be large, and implementing it may mean proceeding in steps. As the adage says "Rome wasn't built in a day". Therefore it becomes important to consider not just the value of the (final) solution, but also the values at intermediate steps. Such incremental models have gained popularity in the last years [7, 1, 9], because of their practical applications to network design problems, disaster recovery, and planning.

As a first approximation to this extra level of complexity, we consider the setting in which we want to evaluate our solution at each time step, and would like to maximize the sum of the values of the intermediate solutions. To formalize this, consider a finite ground set E of q elements, together with a valuation $v : 2^E \to \mathbb{Z}_+$. The valuation function measures some quantity of interest over a subset of E, for example, the size of a maximum matching, the maximum value of an independent set in a matroid, or a maximum flow. Our goal is to find a permutation $\sigma : E \to \{1, \dots, |E|\}$ that maximizes

$$f(\sigma) = \sum_{i=0}^{q} v\left(\{e \in E : \sigma(e) \leq i\}\right). \tag{1}$$

This is a very general class of problems, which also includes for example scheduling problems, production planning problems and routing problems; such problems typically involve finding a permutation of tasks to perform.

Even for simple, polynomially computable valuations v, the problem of finding the best σ might be NP-hard. This applies for example to the situation in which E corresponds to some of the edges of a directed or undirected graph $G = (V, E_0 \cup E)$ with capacities on its edges, and $v(F)$ represents the maximum flow value from s to t (where $s, t \in V$) in the graph $(V, E_0 \cup F)$. The NP-hardness of this incremental problem was shown by Nurre and Sharkey [9], see also Kalinowski et al. [7].

On the tractable side, the incremental problem (1) can be solved efficiently if $v(F)$ represents the weight of a maximum-weight independent subset of F in a matroid M with ground set E. Indeed, an optimum permutation can be obtained from a maximum-weight independent set B for the entire ground set E in the following way. First, order B in order of non-increasing weight followed by all elements of $E \setminus B$ in an arbitrary order. In the case of the incremental spanning tree problem, this was also derived in [6].

In this paper, we consider a class of valuations v which arise naturally from unweighted combinatorial optimization problems, and for which we are able to provide a worst-case analysis of a greedy-like algorithm. This class of valuations is defined axiomatically. First, we require that v takes nonnegative integer values,

(A1): $\forall F \subseteq E : v(F) \in \mathbb{N}$

and is monotonically non-decreasing and can only increase by at most 1 when an element is added:

(A2): $\forall A \subset E, \forall e \in E \setminus A : v(A) \leq v(A \cup \{e\}) \leq v(A) + 1$.

Additionally, we assume that for any k with $v(\emptyset) = \min_F v(F) \leq k \leq \max_F v(F) = v(E)$, there exists a set of cardinality at most k achieving the value k:

(A3): For all $k : v(\emptyset) \leq k \leq v(E), \exists A \subseteq E : |A| \leq k$ and $v(A) = k$.

Consider, for example, an independence system $\mathcal{I}$ on $E_0 \cup E$, i.e. $\mathcal{I} \subseteq 2^{E_0 \cup E}$ and $\mathcal{I}$ is closed under taking subsets. Then if we define $v(F)$ for $F \subseteq E$ as the cardinality of the largest independent subset of $E_0 \cup F$, we can easily see that $v(\cdot)$ satisfies $(A1)$, $(A2)$ and $(A3)$. This generalizes the matroid setting mentioned previously.

We further assume one additional key property, that v satisfies the following *discrete convexity* property:

(A4) Discrete Convexity: $\forall A, C \subseteq E$ with $v(C) - v(A) > 1, \exists B : v(B) = v(A) + 1$ and $|B| - |A| \leq \frac{|C| - |A|}{v(C) - v(A)}$. Furthermore if $A \subset C$ then $A \subset B \subset C$.

This discrete convexity is not satisfied by all independence systems. However, we show in Section 2 that if a certain family of polyhedra is integral then the discrete convexity property is satisfied.

▶ **Theorem 1.** *Let $\mathcal{I} \subseteq 2^{E_0 \cup E}$ be any independence system. Let $P(\mathcal{I}) \subseteq \mathbb{R}^{E_0 \cup E}$ be the convex hull of incidence vectors of all independent sets in $\mathcal{I}$. If for every integer k,*

$$P(\mathcal{I}) \cap \left\{ x \mid \sum_{e \in E_0 \cup E} x_e = k \right\}$$

is integral then $(A4)$ holds for $v : 2^E \to \mathbb{Z}_+$ defined as $v(F) = \max\{|I| \mid I \in \mathcal{I}, I \subset E_0 \cup F\}$.

In Section 2, we show that this holds for example when $\mathcal{I}$ corresponds to the matchings in a (not necessarily bipartite) graph, or to the independent sets common to two matroids, or to the (independent or) stable sets in a claw-free graph. This last example, although more esoteric, is interesting as a complete description of $P(\mathcal{I})$ by linear inequalities is unknown but we can nevertheless rely on the above theorem. The incremental valuation problem in the case of *bipartite* matchings was already considered in [7], where the authors propose

---

Algorithm 1: Quickest-To-Ultimate for Incremental Valuation

Input   : A valuation function v as above.
Output: A permutation σ of E

1 Compute $O \subseteq E$ of minimum cardinality such that $v(O) = v(E)$;
2 Set $F = \emptyset$;
3 **for** $i = 1$ *to* $v(E) - v(\emptyset)$ **do**
4   Compute $S \subset O \setminus F$ such that $v(F \cup S) \geq v(\emptyset) + i$ and $|S|$ is minimum;
5   Set $F = F \cup S$;
6 **end**
7 Output σ consistent with how elements were added to S;

---

several approximation algorithms, the best achieving an approximation ratio of 3/4. Other problems falling under the framework discussed here were not considered before.

For any valuation satisfying $(A1) - (A4)$, we provide an approximation algorithm for the incremental valuation problem. For an efficient implementation, we assume that we can compute efficiently (or have oracle access to) the valuation $v(\cdot)$ and we can also find efficiently a minimum cardinality set O with $v(O) = v(E)$. Our algorithm first computes a smallest set $O \subseteq E$ achieving $v(O) = v(E)$, and then starting from $S = \emptyset$ with value $v(\emptyset)$, repeatedly and greedily adds a smallest subset of O to increase $v(S)$ by 1 until all elements of O have been added and then finishes the ordering with the elements of $E \setminus O$. This algorithm is formally described in Section 3 and in Algorithm 1. In Section 3, we present a worst-case analysis of this algorithm:

▶ **Theorem 2.** *For any valuation satisfying* $(A1) - (A4)$*, Algorithm 1 (Quickest-to-Ultimate) is a γ-approximation for the incremental valuation problem, where*

$$\gamma = \frac{9 + \sqrt{21}}{15} > 0.9055.$$

The proof of this result is given in Section 4. We also show that the bound of γ is tight in the sense that there are instances of the valuation problem in which the algorithm cannot do better.

2   Problems in this Framework

2.1   Maximum matchings

One of the basic problems that falls in this framework is the *Incremental Matching Problem*. Given a graph $G = (V, E_0 \cup E)$, where E_0 denotes the edges already present at the start, we would like to find an ordering of the edges of E so as to maximize the average size of the maximum matching in E_0 union the edges already selected. This corresponds to the valuation with $v(F) = \mu(E_0 \cup F)$ where $\mu(A)$ equals the size of the maximum matching in the graph (V, A). The bipartite version of this problem is considered in [7], and two different greedy approximation algorithms are presented. The first one, QUICKEST-INCREMENT, is a locally greedy algorithm that seeks to minimize the number of edges needed to increase the size of the matching by one, until we reach a maximum matching of the entire graph. Kalinovski et al. [7] prove an approximation guarantee of $\frac{2}{3}$ for this algorithm. Their second algorithm, QUICKEST-TO-ULTIMATE, is globally greedy, in the sense that it first computes a maximum matching of the entire G, and then only adds edges from this matching, in a locally greedy fashion. For this algorithm, [7] prove an approximation bound of $\frac{3}{4}$. In this

paper, we generalize this algorithm to a larger class of incremental problems and improve the guarantee to $0.9055\cdots$.

This matching problem, even in the non-bipartite case, fits in the framework discussed here. One can show that this valuation $v(F) = \mu(E_0 \cup F)$ satisfies the discrete convexity property ($A4$), by considering maximum matchings in A and C, and their symmetric difference and carefully arguing about it. Although this is possible, this does not generalize easily to other problems.

Discrete convexity, however, is easier to argue polyhedrally as we show next.

2.2   Polyhedral characterization for discrete convexity

Let $\mathcal{I} \subseteq 2^{E_0 \cup E}$ be any independence system, and let $v(F) = \max\{|I| : I \in \mathcal{I} \text{ and } I \subseteq E_0 \cup F\}$. Let $P = \operatorname{conv}\{\chi(I) : I \in \mathcal{I}\}$ be the convex hull of all independent sets, and as we will see, we do not necessarily need to know a complete description of P in terms of linear inequalities. We will show Theorem 1 that discrete convexity ($A4$) holds if, for any integer k,

$$P \cap \{x : x(E \cup E_0) = k\}$$

is integral.

Proof of Theorem 1. For A (resp. C), let I_A (resp. I_C) be a maximum independent subset of $E_0 \cup A$ (resp. $E_0 \cup C$). So, $v(A) = |I_A|$ and $v(C) = |I_C|$. Let $\ell = |I_C| - |I_A|$. Now consider $y = \frac{1}{\ell}\chi(I_C) + (1 - \frac{1}{\ell})\chi(I_A)$. By convexity $y \in P$ and by construction, we have $y(E \cup E_0) = |I_A| + 1$. Thus, $y \in P \cap \{x : x(E \cup E_0) = |I_A| + 1\}$, and by integrality of this polytope, we have that there exists $x = \chi(S) \in P \cap \{x : x(E \cup E_0) = |I_A| + 1\}$ with

$$
\begin{aligned}
|S \cap E| &= \min\{x(E) : x \in P \cap x(E \cup E_0) = |I_A| + 1\} \\
&\leq y(E) = \frac{1}{\ell}|I_C \cap E| + (1 - \frac{1}{\ell})|I_A \cap E| \\
&\leq \frac{1}{\ell}|C| + (1 - \frac{1}{\ell})|A| = |A| + \frac{|C| - |A|}{v(C) - v(A)}.
\end{aligned}
$$

Thus $B = S \cap E$ satisfies the first part of the claim in Theorem 1.

Now consider the case in which $A \subseteq C$. Proceeding as before, we get

$$y \in P \cap \{x : x(E \cup E_0) = |I_A| + 1\} \cap \{x : x_e = 0 \,\forall e \in E \setminus C\},$$

and this is again an integral polytope since it is the face of an integral polytope. Now minimizing $x(E \setminus A)$ over

$$P \cap \{x : x(E \cup E_0) = |I_A| + 1\} \cap \{x : x_e = 0 \,\forall e \in E \setminus C\},$$

we get $x = \chi(T) \in P \cap \{x : x(E \cup E_0) = |I_A| + 1\}$ with $T \subseteq E_0 \cup C$ and

$$|T \cap (E \setminus A)| \leq y(E \setminus A) = \frac{1}{\ell}|I_C \cap (E \setminus A)| \leq \frac{|C| - |A|}{v(C) - v(A)}.$$

This means that $B = A \cup (T \cap E)$ is such that $A \subseteq B \subseteq C$,

$$|B| \leq |A| + \frac{|C| - |A|}{v(C) - v(A)}$$

and $v(B) \geq v(T) = |I_A| + 1$. Thus either $v(B) = |I_A| + 1$, or we can eliminate one by one elements of $B \setminus A$ as long as $v(\cdot)$ is not equal to $v(A) + 1$. Eventually, we find a set with the right requirements.   ◀

2.3 Maximum stable set in claw-free graphs

A graph $G = (V, E)$ is claw-free if it does not include $K_{1,3}$ (the star on 4 vertices) as an induced subgraph. The line graph of any graph is claw-free, but the converse is not true as there exist claw-free graphs which are not line graphs. Minty [8] and Sbihi [10] have shown that the maximum stable (or independent) set in a claw-free graph is polynomially solvable. When the claw-free graph is a line graph, this extends Edmonds' algorithm [4, 3] for maximum matching, as the maximum matching problem in a graph is equivalent to the maximum stable set problem in its line graph.

By taking the line graph, we can extend the incremental matching problem to an incremental stable set problem in a claw-free graph $G = (V, E)$ in which we are given an initial vertex set V_0 and our task is to choose an ordering of the remaining vertices in $V \setminus V_0$ so to maximize the average size of a maximum stable set in the corresponding prefix. Thus, here $v(F)$ denotes the size of the largest stable set in $G[V_0 \cup F]$. As said before, if the claw-free graph is not a line graph, this is a strictly more general problem than the incremental matching problem.

A complete description of the stable set polytope P for claw-free graphs is still unknown (see, eg, section 69.4a in Schrijver [11]), but we can nevertheless use Theorem 1 to show that (A4) holds (the other conditions (A1), (A2) and (A3) obviously hold).

▶ **Theorem 3.** *Let P be the stable set polytope of a claw-free graph $G = (V, E)$. Then for any integer k, we have that*

$$P \cap \left\{ x \in \mathbb{R}^V \mid \sum_{v \in V} x_v = k \right\}$$

is integral.

Proof. We exploit the known adjacency properties of the stable set polytope (of any graph). Chvátal [2] has shown that two stable sets S_1 and S_2 in G are adjacent vertices in the stable set polytope if and only if their symmetric difference $S_1 \triangle S_2$ induces a connected subgraph of G. When the graph is claw-free, this connected subgraph $G[S_1 \triangle S_2]$ must be a path, and therefore this means that $-1 \leq |S_1| - |S_2| \leq 1$.

Consider any vertex x^* of $P \cap \left\{ x \in \mathbb{R}^V \mid \sum_{v \in V} x_v = k \right\}$. x^* must lie on a face of dimension at most 1 of P, and therefore must be in the line segment between two adjacent vertices of P. But since the sizes of these stable sets can differ by at most 1, we derive that x^* must be a vertex of P, and integrality follows. ◀

Thus our approximation algorithm result applies to the incremental maximum stable set problem in claw-free graphs.

The adjacency argument in the proof of Theorem 3 generalizes in the sense that Theorem 1 is equivalent to imposing that any pair of adjacent vertices of $P(\mathcal{I})$ differ in cardinality by at most one unit.

2.4 Matroid intersection

Another generalization of the incremental version of the bipartite matching problem is to consider the incremental version of matroid intersection. Let M_1 and M_2 be matroids defined on the same ground set, say $E_0 \cup E$, and for $F \subseteq E$, let $v(F)$ be the cardinality of the largest common independent set to the two matroids within $E_0 \cup F$.

For matroid intersection, we can directly use Theorem 1 to show that the discrete convexity holds. The matroid intersection polytope P has been characterized by Edmonds [5], and the integrality of $P \cap \{x | \sum_i x_i = k\}$ follows simply by truncating both matroids to size k. Thus Theorem 1 can be used to prove (A4) and Theorem 2 can be used to derive a better than 0.9-approximation algorithm for the incremental maximum matroid intersection problem.

## 3   Quickest-To-Ultimate for Incremental Problems

Algorithm Quickest-To-Ultimate (Q2U in short, see Algorithm 1) was introduced by Kalinowski et al. [7] for the problem of incremental flows (defined in the introduction). The general idea behind this algorithm is to reach the maximum valuation possible in the shortest number of steps. In the setting of incremental flows, finding the smallest set O of edges whose addition gives a maximum flow is a hard problem, and they resort to a mixed integer program for finding O. In this direction it is known that the incremental flow problem is NP-hard even if the capacities are restricted to be one or three [9]. In the case of unit capacities, Q2U becomes a polynomial approximation algorithm, and in [7] it is shown that it finds a solution with at least half the value of the optimum for the incremental flow problem with unit capacities, and they also show a matching family of examples in which this approximation ratio is attained as the size of the graph grows. In the case of bipartite matchings, a further restriction of the problem, Q2U is shown to find a solution to the incremental matching problem of value at least $3/4$ of the optimum, and they show an instance in which the value obtained by Q2U is $\frac{68}{69}$ of the value attained by the optimal solution. Theorem 2 and the example given in Section 3.1 below close this gap for a more general class of valuations, which includes the incremental matching problem.

We show Theorem 2, that for any valuation satisfying (A1)-(A4) the performance guarantee of the algorithm is at least $\frac{9+\sqrt{21}}{15} > 0.9055$. The proof appears later in this section, and our analysis is tight as we show next.

3.1   Bad instance for Quickest-to-Ultimate

In the special case of matchings, and even bipartite matchings (or any setting which includes bipartite matchings), the analysis is tight. Consider, indeed, a graph formed by a disjoint copies of P_3, a path with 3 edges, and one copy of P_{4b+3}, a path with $4b+3$ edges, see Figure 1. The edges of E_0 correspond to each middle edge in the copies of P_3, and every fourth edge of P_{4b+3}, starting with the second one. The remaining edges are edges of E. The valuation is $v : E \to \mathbb{N}$, where $v(S)$ is the size of a maximum matching using edges from $E_0 \cup S$.

In this graph, we have $q := |E| = 2a + 3b + 2$ edges to be added, the original matching has size $m := a + b + 1$, and it can grow by $r := a + b + 1$. The minimum number of edges we need to add to reach this maximum matching is $c_r := 2a + 2b + 2$. Quickest-to-Ultimate adds these edges in pairs that increase the matching, so it adds two edges for each increment in the maximum matching. The value it attains is then $f_{alg} = (q+1)(m+r) - \sum_{i=1}^{a+b+1} 2i = 3a^2 + 5b^2 + 8ab + 7a + 9b + 3$. On the other hand, here is a better solution. The solution first adds the b edges of P_{4b+3} that increase the maximum matching from m to $m+b$, then it adds a pairs of edges to increase the matching by a and then it adds $2b+2$ edges to increase the matching by one. This gives a value f with $f_{opt} \le f = (q+1)(m+r) - \sum_{i=1}^{b} i - \sum_{i=b+1}^{a+b}(2(i-b)+b) - (2a+b+2b+2) = 3a^2 + \frac{11}{2}b^2 + 9ab + 7a + \frac{17}{2}b + 4$. A straightforward optimization over a and b, yields that the minimum value of $\frac{f_{alg}}{f_{opt}}$ is attained when a and b
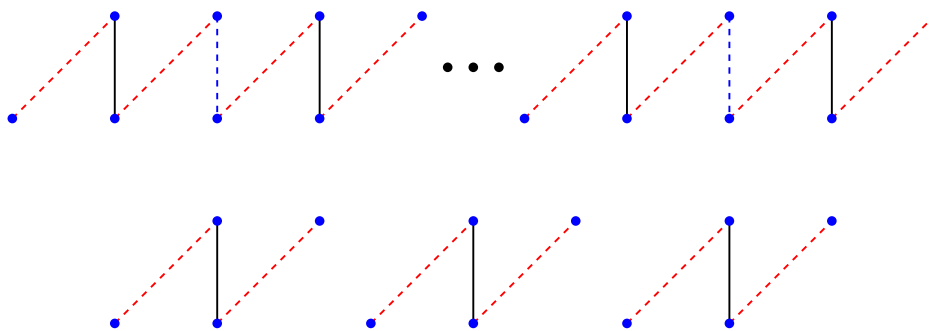
Figure 1 Solid black edges are edges of E_0 and dashed edges are the edges of E.

go to infinity, with $a = \delta b$, with $\delta = \frac{\sqrt{21}}{6} - \frac{1}{2}$, with value of $\frac{9+\sqrt{21}}{15}$. This is the worst case for Q2U and matches the bound we prove in Theorem 2.

4 Analysis

Before diving into the analysis of Q2U, we introduce some notation and exhibit some convexity properties of various sequences associated with these incremental problems.

We denote $v(\emptyset)$ by m, $v(E)$ by $m + r$, and $|E|$ by q. For any permutation σ of E, define

$$d_i(\sigma) := |\{j \in \{0, \ldots, q\} \, : \, v(\{e \in E \, : \, \sigma(e) \leq j\}) \leq m + i - 1\}|,$$

which is the number of elements needed for the solution σ to get to a valuation $m + i$. We will denote by d_i^* the values of $d_i(\sigma^*)$ for an optimal solution σ^* to (1), and by d_i the values of $d_i(\sigma)$ for the permutation σ output by Q2U.

Define for each $i \in \{0, \ldots, r\}$

$$c_i := \min\{|S| \, : \, v(S) \geq m + i\}.$$

By definition, we have $c_i \leq d_i$ and similarly $c_i \leq d_i^*$. Also by $(A2)$ we must have $c_i \geq i$, and by $(A3)$,

$$c_i \leq m + i, \tag{2}$$

for $i \in \{0, \ldots, r\}$.

We show that our assumptions imply that both the sequence $\{c_i\}_{i=1}^r$, and the sequence $\{d_i\}_{i=1}^r$ satisfy a convexity property.

▶ **Lemma 4.** *The sequence $\{c_i\}_{i=1}^r$ satisfies*

$$c_{i+1} - c_i \geq c_i - c_{i-1}, \quad 1 \leq i \leq r - 1.$$

Proof. To see this, apply $(A4)$ to the respective solutions S_{i-1}, S_i, S_{i+1} where

$$S_j = \arg\min_S \{|S| : v(S) \geq m + j\}.$$

Note first that by $(A2)$ we have that $v(S_j) = m + j$ for $j = i - 1, i, i + 1$. This implies $v(S_{i+1}) - v(S_{i-1}) = 2 > 1$, and so by $(A4)$, there exists B such that $v(B) = v(S_{i-1}) + 1 = m + i$ and $2(|S_i| - |S_{i-1}|) \leq 2(|B| - |S_{i-1}|) \leq (|S_{i+1}| - |S_{i-1}|)$. This implies Lemma 4. ◀

The solution given by Q2U also satisfies this same convexity property.

▶ **Lemma 5.** *The sequence $\{d_i\}_{i=1}^r$ corresponding to Q2U satisfies*

$$d_{i+1} - d_i \geq d_i - d_{i-1}, \quad 1 \leq i \leq r - 1.$$

Proof. To see this, denote by S_i the set computed in the inner loop of the algorithm at step i. That is $S_i \subset F_r \setminus (S_1 \cup \ldots \cup S_{i-1})$ such that $|S_i|$ is minimum and $v(S_1 \cup \ldots \cup S_i) \geq m + i$. Minimality of $|S_i|$, and property $(A2)$ imply that $v(S_1 \cup \ldots \cup S_i) = m + i$, and then $d_i = |S_1 \cup \ldots \cup S_i|$. Now take $i \in \{1, \ldots, r-1\}$. Then $v(S_1 \cup \ldots \cup S_{i+1}) - v(S_1 \cup \ldots \cup S_{i-1}) = 2 > 1$, and then by property $(A4)$ there is a B such that $v(B) = m + i$ and $2(|B| - d_{i-1}) \leq d_{i+1} - d_{i-1}$. Finally by minimality of $|S_i|$ we must have $d_i \leq |B|$, which implies the claim. ◀

We could also show that any optimum ordering σ^* satisfies the same convexity property: $d_{i+1}^* - d_i^* \geq d_i^* - d_{i-1}^*$ for all i, although we will not need this. This requires the second part of (A4) which says that if $A \subseteq C$ then B can be chosen to be sandwiched by A and C.

4.1   Local minima

We also show that the convexity property (A4) implies a relationship between local and global optima, that will be used to derive the optimal upper bound for the Quickest-To-Ultimate Algorithm 1.

▶ **Lemma 6.** *Let S and T be two subsets of E, such that $v(S) = m + |S|$ and $v(T) = m + |T|$, and let S be maximal with this property, that is for any $S' \supset S$ we have $v(S') < m + |S'|$. Then, $2|S| \geq |T|$.*

This is a generalization of the well-known result that any maximal matching is at least half the size of a maximum matching.

Proof. If $|S| \leq |T|$, there is nothing to prove, so we assume that $|T| > |S|$. Now use (A4) with $A = S$ and $C = S \cup T$. Then there is a set B with $S \subset B \subset S \cup T$ such that $v(B) = v(S) + 1$ and

$$|B| - |S| \leq \frac{|S \cup T| - |S|}{v(S \cup T) - v(S)} \leq \frac{|T|}{|T| - |S|}.$$

On the other hand, by the maximality of S, we must have $|B| - |S| \geq 2$. Putting these two together yields

$$2 \leq \frac{|T|}{|T| - |S|},$$

or equivalently

$$2|S| \geq |T|. \qquad \blacktriangleleft$$

4.2   Quickest-To-Ultimate

To analyze Quickest-To-Ultimate, we need to introduce some additional parameters related to the instance being considered. Define

$$p = \max\{|P| \ : \ P \subset E, \ v(P) = m + |P|\},$$

the maximum size of a set that, if added sequentially, increases the valuation at each step. In other words, $p = \max\{i : c_i = i\}$. Clearly $p \leq r$. Also, by maximality of p, we have that

$$\begin{cases} c_i = i & i \leq p \\ c_i \geq p + 2(i - p) & i > p. \end{cases} \tag{3}$$

For Q2U, we are interested in the quantity s, the number of times the set S in the inner loop is a singleton. Note that by Lemma 5, these s iterations occur at the beginning, so an equivalent way to define s is

$$s = \max\{i \in \{1, \ldots, r\} \ : \ d_i = i\}.$$

Our objective is to relate the quantities s and p, which will give us some control over the approximation ratio f_{alg}/f_{opt}. We must have $s \leq p$, since otherwise it would contradict the maximality of p. To get a lower bound on s, define S to be the set of the first s elements added by the algorithm, and $T = P \cap O$, where O is the set of elements chosen by Q2U to first reach $v(E)$ and P is a set of p elements with $v(P) = m + p$. Note that we have $v(S) = m + |S|$ and $v(T) = m + |T|$, and S must be maximal, by definition of s. Then by using Lemma 6, we conclude that $2|S| \geq |T| = |P \cap O|$. This implies that

$$q = |E| \geq |O \cup P| = |O| + |P| - |O \cap P| \geq c_r + p - 2s. \tag{4}$$

And we also know that

$$q \geq c_r. \tag{5}$$

Finally, we need the following inequality. Observe that all the elements that are used by the algorithm come from O, and conversely, all the elements of O must be used by the algorithm to reach valuation $v(E)$, by minimality of c_r. This means that $|O| = c_r = d_r = \sum_{i=1}^{r}(d_i - d_{i-1})$, and using the definition of s, then $c_r = s + \sum_{i=s+1}^{r}(d_i - d_{i-1})$, from which it follows that

$$c_r \geq 2r - s. \tag{6}$$

We are now ready to prove Theorem 2.

Proof of Theorem 2. For any permutation σ, we can rewrite $f(\sigma)$ as

$$f(\sigma) = (q+1)(m+r) - \sum_{i=1}^{r} d_i(\sigma). \tag{7}$$

In particular, for the optimum permutation σ^* and its optimum value $f_{opt} = f(\sigma^*)$, we have:

$$f_{opt} = (q+1)(m+r) - \sum_{i=1}^{r} d_i^* \leq (q+1)(m+r) - \sum_{i=1}^{r} c_i. \tag{8}$$

Using (3) and distinguishing between $i \leq p$, $p < i < r$ and $i = r$, we can write:

$$f_{opt} \leq (q+1)(m+r) - p^2/2 - p/2 + pr - r^2 + r - c_r. \tag{9}$$

Now, denoting the value obtained by Q2U as f_{alg}, and using the definition of s, we have

$$f_{alg} = (q+1)(m+r) - \sum_{i=1}^{r} d_i = (q+1)(m+r) - \sum_{i=1}^{s-1} i - \sum_{i=s}^{r} d_i. \tag{10}$$

To upper bound the last term of (10), we use the following lemma, whose proof is given in the appendix.

▶ **Lemma 7.** *Let $f : \{0, \ldots, a\} \to \mathbb{N}$ be a discrete convex function, i.e. $f(i + 1) - f(i) \geq f(i) - f(i - 1)$, such that $f(0) = 0$ and $f(a) = b$. Furthermore let $b = ka + t$, where $t \in \{0, \ldots, a - 1\}$. Then,*

$$\sum_{i=0}^{a} f(i) \leq (b + ka^2 + t^2)/2 = \frac{(a + 1)b}{2} - \frac{t(a - t)}{2}.$$

Applying this to $f(i) = d_{s+i} - s$, we obtain

$$f_{alg} \geq (q + 1)(m + r) - s(s - 1)/2 - (r - s + 1)s - (r - s + 1)(c_r - s)/2 + t(r - s - t)/2,$$

where $t = c_r - s \bmod r - s$. Or after simplification:

$$f_{alg} \geq (q + 1)(m + r) - rs/2 - (r - s - 1)c_r/2 + t(r - s - t)/2. \tag{11}$$

We need to find the minimum value attainable by $f_{alg}/f_{opt} \leq 1$, which is a lower bound on the approximation ratio. We will show that this lower bound coincides with the upper bound given by the example in Section 3.1. Denote by P_{opt} (resp. P_{alg}) the right-hand-side of inequality (9) (resp. (11)). To find this lower bound, we maximize P_{opt}/P_{alg} over all integral q, m, c_r, r, p, s and t satisfying the conditions:
1. $r \geq p \geq s \geq 0$
2. (5): $q \geq c_r$
3. (4): $q \geq c_r + p - 2s$
4. (6): $c_r \geq 2r - s$
5. $m + r \geq c_r$
6. $t = c_r - s \bmod r - s$.
We first show that the we can ignore all but the quadratic terms in the variables q, m, r, p, s, t. If we double each of q, m, c_r, r, p, s, then t also doubles by **6**, and all inequalities **1-5** are still satisfied. Furthermore, if we denote P'_{opt} and P'_{alg} the respective values of the bounds after doubling, we have

$$\frac{P'_{opt}}{P'_{alg}} = \frac{4P_{opt} - 2(m + r) + p - 2r + 2c_r}{4P_{alg} - 2(m + r) + c_r} \geq \frac{4P_{opt} - 2(m + r) + c_r}{4P_{alg} - 2(m + r) + c_r} \geq \frac{P_{opt}}{P_{alg}},$$

where in the first inequality we have used that $c_r - 2r + p \geq c_r - 2r + s \geq 0$, by **4**, and the second inequality follows from **5.**. So, for the extremum, we can assume there are no linear terms:

$$\frac{P_{opt}}{P_{alg}} \leq \frac{q(m + r) - p^2/2 + pr - r^2}{q(m + r) - rs/2 - (r - s)c_r/2 + t(r - s - t)/2}.$$

Now, using the inequality **5**, we can eliminate m, and obtain

$$\frac{P_{opt}}{P_{alg}} \leq \frac{qc_r - p^2/2 + pr - r^2}{qc_r - rs/2 - (r - s)c_r/2 + t(r - s - t)/2}.$$

The remaining constraints are now **1–4** and **6**. If $s > 0$, and $q > c_r + p - 2s$ we can decrease all variables by one unit, and preserve the above inequalities. In so doing, the value of t does not change, and both the numerator and denominator decrease by the same amount

$$c_r + q - r - 1/2 \geq 0.$$

This implies we can decrease all variables by the same amount until one of two things happen. Either $s = 0$, or $q = c_r + p - 2s$. In the latter case, since we also have that $q \geq c_r$ by **2**, this

implies that $2s \leq p$. At this point, after eliminating q (and replacing it by $c_r + p - 2s$), the ratio becomes:

$$\frac{P_{opt}}{P_{alg}} \leq \frac{(c_r + p - 2s)c_r - p^2/2 + pr - r^2}{(c_r + p - 2s)cr - rs/2 - (r - s)c_r/2 + t(r - s - t)/2}.$$

If we decrease all remaining variables by one unit, both the denominator and numerator of the above fraction decrease by

$$c_r + p - r - 2s + 1/2 \geq 0,$$

since $p \geq 2s$ and $c_r \geq r$. And we can continue this process until $s = 0$. In both cases we obtain

$$\frac{P_{opt}}{P_{alg}} \leq \frac{(c_r + p)c_r - p^2/2 + pr - r^2}{(c_r + p)c_r - rc_r/2 + t(r - t)/2}. \tag{12}$$

And we need to maximize this over integral solutions to $r \geq p \geq 0$, $c_r \geq 2r$ and $t = c_r \bmod r$. We consider two cases, depending on the value of c_r.

1. If $c_r = 3r + k$, for $k \geq 0$, and we discard the (nonnegative) term involving t in (12), we obtain:

$$\frac{P_{opt}}{P_{alg}} \leq \frac{8r^2 + 4rp - p^2/2 + k(6r + p + k)}{15r^2/2 + 3rp + k(11r/2 + p + k)}.$$

As an upper bound, we can take the maximum of this value for $k = 0$ and the ratio of the terms involving k, and therefore obtain that:

$$\frac{P_{opt}}{P_{alg}} \leq \max\left(\frac{8r^2 + 4rp - p^2/2}{15r^2/2 + 3rp}, \frac{6r + p + k}{11r/2 + p + k}\right).$$

The second term on this maximum is at most $\frac{12}{11} < \frac{1}{\gamma}$ where γ is our desired bound. The first one, by setting $\alpha = r/p \geq 0$, is equal to

$$\frac{8\alpha^2 + 4\alpha - 1/2}{15\alpha^2/2 + 3\alpha}.$$

This ratio is maximized for $\alpha = \frac{5}{8} + \frac{\sqrt{41}}{8}$, and it achieves a value of

$$\frac{112\sqrt{41} + 656}{99\sqrt{41} + 615} < \frac{1}{\gamma}.$$

2. If $2r \leq c_r < 3r$, then $c_r = 2r + t$, and (12) becomes:

$$\frac{P_{opt}}{P_{alg}} \leq \frac{3r^2 + t^2 - p^2/2 + 3pr + pt + 4rt}{3r^2 + t^2/2 + 2pr + pt + 4rt}.$$

It is easy to see that for any constant C, and fixed values of r and p, the set

$$I = \{t \in [0, r] : \frac{3r^2 + t^2 - p^2/2 + 3pr + pt + 4rt}{3r^2 + t^2/2 + 2pr + pt + 4rt} \leq C\},$$

is a convex set, and so the maximum value of this ratio is achieved at either $t = 0$ or $t = r$. If we set $t = r$, we obtain

$$\frac{8r^2 + 4pr - p^2/2}{15r^2/2 + 3pr} < \frac{1}{\gamma},$$

Algorithm 2: Quickest-Increment for Incremental Valuation

Input   : A valuation function v as above.
Output: A permutation σ of E

1 Set $F = \emptyset$;
2 **for** $i = 1$ *to* r **do**
3 | Compute $S \subset E \setminus F$ such that $v(F \cup S) \geq v(\emptyset) + i$ and $|S|$ is minimum;
4 | Set $F = F \cup S$;
5 **end**
6 Output σ consistent with how elements were added to S.;

---

as we have already verified. If $t = 0$, we obtain

$$\frac{3r^2 + 3pr - p^2/2}{3r^2 + 2pr},$$

which is maximized at $\alpha = r/p = \frac{1}{2} + \frac{\sqrt{21}}{6}$, with value $\frac{1}{\gamma} = \frac{9}{4} - \frac{\sqrt{21}}{4}$, or $\gamma = \frac{9+\sqrt{21}}{15}$.   ◀

This settles the question of how well Quickest-to-Ultimate approximates the maximum incremental matching problem.

5   Upper bound for Quickest-Increment

Quickest-Increment (QI) is another algorithm suggested in [7]. The idea is to increase the size of the current solution by adding as few elements as possible. In that paper, among other results, it was shown that QI has a performance guarantee of $2/3$ in the case of bipartite matchings, and also they claim a bound of $3/4$ if $r \geq 70$. It is also conjectured there that, as $r \to \infty$, the approximation guarantee for Quickest-Increment approaches 1. We show a family of instances that show that this is false.

Consider the instance H formed by P_7, the path with seven edges, in which the only edges of E_0 are the second and the second to last. Observe that both Q2U and QI have the same performance on this small graph, and it is even optimal. In this small graph we have $q = 5$, $r = 2$ and $m = 2$. There are two incomparable choices for d_i. The first one, given by Q2U, is $d_1 = 2 = d_2$. The second one is given by QI and it is $d_1 = 1$, $d_2 = 4$. They both have value 18, which is optimum.

Now consider the instance G, which is a copies of H. Both algorithms fail to realize the optimum. When considering a copies, we obtain $q = 5a$, $r = 2a$ and $m = 2a$. Algorithm Q2U returns $d_i = 2i$ for $i = 1, \ldots, 2a$, with a value of $f = (5a + 1)(4a) - \sum_{i=1}^{2a} 2i = 16a^2 + 2a$. Algorithm QI return $d_i = i$ for $i = 1, \ldots, a$ and $d_i = 4(i - a) + a$ for $i = a + 1, \ldots, 2a$, with a value of $f = (5a + 1)(4a) - \sum_{i=1}^{a} i - \sum_{i=a+1}^{2a} (4i - 3a) = 33a^2/2 + 3a/2$.

Now suppose we use the QI strategy on k of the a copies and the Q2U strategy on the rest. Then we get $d_i = i$ for $i = 1, \ldots, k$, $d_i = k + 2(i - k)$ for $i = k + 1, \ldots, 2a - k$ and $d_i = 4(i - 2a + k) + 4(a - k) + k$ for $i = 2a - k + 1, \ldots, 2a$, and a value of

$$f = (5a + 1)(4a) - \sum_{i=1}^{k} i - \sum_{i=k+1}^{2a-k} (2(i - k) + k) - \sum_{i=2a-k+1}^{2a} (5(i - 2a + k) + 4(a - k) + k)$$

$$f = 16a^2 + 2a - (3k^2/2 + k/2 - 2ak).$$

Optimizing over k we obtain that for $k = 2a/3$, this solution has a value of $f = 50a^2/3 + 5a/3$. So asymptotically as we take $a \to \infty$ the approximation factor for Q2U approaches $\frac{24}{25}$ and for QI is approaches $\frac{99}{100}$. Note that this family of examples has $r = 2a \to \infty$, and so contradicts the conjecture about QI in [7]. Note this also shows that the approximation guarantee of Q2U is bounded, even when $r \to \infty$. It is possible to show a family of examples that show that when $r \to \infty$, the approximation guarantee for QI approaches $\frac{7}{8}$.

References

1 M. Baxter, T. Elgindy, A. T. Ernst, T. Kalinowski, and M. W. P. Savelsbergh. Incremental network design with shortest paths. *European Journal of Operational Research*, 242:51–62, 2015. `doi:10.1016/j.ejor.2014.04.018`.

2 V. Chvátal. On certain polytopes associated with graphs. *Journal of Combinatorial Theory, Series B*, 18:138–154, 1975. `doi:10.1016/0095-8956(75)90041-6`.

3 J. Edmonds. Maximum matching and a polyhedron with 0,1 vertices. *Journal of Research National Bureau of Standards, Section B69*, pages 125–130, 1965. `doi:10.6028/jres.069B.013`.

4 J. Edmonds. Paths, trees and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965. `doi:10.4153/CJM-1965-045-4`.

5 J. Edmonds. Submodular functions, matroids, and certain polyhedra. *Combinatorial Structures and Their Applications*, pages 69–87, 1970. `doi:10.1007/3-540-36478-1_2`.

6 K. Engel, T. Kalinowski, and M. W. P. Savelsbergh. Incremental network design problem with spanning trees. *Journal of Graph Algorithms and Applications*, 2013. `arXiv:0902.0885, doi:10.7155/jgaa.00423`.

7 T. Kalinowski, D. Matsypura, and M. W. P. Savelsbergh. Incremental network design with maximum flows. *European Journal of Combinatorial Research*, 3:675–684, 2014. `doi:10.1016/j.ejor.2014.10.003`.

8 G. J. Minty. On maximal independent sets of vertices in claw-free graphs. *Journal of Combinatorial Theory, Series B*, 28:284–304, 1980. `doi:10.1016/0095-8956(80)90074-X`.

9 S. G. Nurre and T. C. Sharkey. Integrated network design and scheduling problems with parallel identical machines: Complexity results and dispatching rules. *Networks*, 63:306–326, 2014. `doi:10.1002/net.21547`.

10 N. Sbihi. Algorithme de recherche d'un stable de cardinalité maximum dans un graphe sans étoile. *Discrete Mathematics*, 29:53–76, 1980. `doi:10.1016/0012-365X(90)90287-R`.

11 A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency.* Springer, 2003.

A   Lower bound for integrals of integer convex functions

We prove Lemma 7. Suppose we have a discrete convex function $f : \{0, \dots, a\} \to \mathbb{N}$, that is, for each $i = 1, \dots, a - 1$, we have

$$f(i + 1) - f(i) \geq f(i) - f(i - 1).$$

Suppose furthermore that $f(0) = 0$ and define $b = f(a)$. We compute a tight upper bound on the value of $\sum_{i=0}^{a} f(i)$ that depends only on a and b. To this end, define

$$n_k = |\{j \in \{1, \dots, a\} : f(j) - f(j - 1) = k\}|.$$

We must have $n_1 + n_2 + \dots = a$, and $n_1 + 2n_2 + \dots = b$, and since f is convex as a sequence, the value of $\sum_{i=0}^{a} f(i)$ in terms of n_k is given by

$$\sum_{i=1}^{n_1} i + \sum_{i=1}^{n_2} (n_1 + 2i) + \sum_{i=1}^{n_3} (n_1 + 2n_2 + 3i) + \dots = \frac{1}{2} \left(n^T v + n^T A n \right),$$

where n is the vector of the n_k, $v_k = k$, and

$$A = \begin{bmatrix} 1 & 1 & 1 & \ldots \\ 1 & 2 & 2 & \ldots \\ 1 & 2 & 3 & \ldots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix},$$

or $A_{k\ell} = \min(k, \ell)$. This is a symmetric, positive definite matrix, since its Schur complement with respect to entry $(1, 1)$ is just a smaller version of the same matrix, and all its coefficients are positive integers. Note that $n^T v = b$ and so it is a constant independent of the vector of n. The solution given by the following optimization problem gives the required upper bound

$$\begin{array}{ll} \text{maximize} & n^T A n \\ \text{subject to} & n_1 + n_2 + \ldots = a \\ & n_1 + 2n_2 + \ldots = b \\ & n_k \in \mathbb{N} \quad k = 1, \ldots, b. \end{array}$$

We will show with the following lemma that a solution n to this problem has at most two consecutive non zeros.

▶ **Lemma 8.** *If there are two positive integers i and j such that $j - i \geq 2$, and $n_i > 0, n_j > 0$, then defining*

$$m = n + (e_{i+1} - e_i) - (e_j - e_{j-1})$$

we have that m is feasible and $m^T A m > n^T A n$.

Proof. Note that m is feasible. On the other hand, since A is symmetric

$$m^T A m - n^T A n = (m + n)^T A(m - n).$$

Now, $m - n = (e_{i+1} - e_i) - (e_j - e_{j-1})$, and then $A(m - n) = \sum_{k=i+1}^{j-1} e_k$. The coefficients of m and n are nonnegative integers, and furthermore $(m + n)_{i+1} > 0$, which implies the result. ◀

This implies a closed form solution to the problem above.

▶ **Theorem 9.** *Suppose $b = ka + t$, for some integer k, and $t \in \{0, \ldots, a - 1\}$. Then the solution to*

$$\begin{array}{ll} \text{maximize} & n^T A n \\ \text{subject to} & n_1 + n_2 + \ldots = a \\ & n_1 + 2n_2 + \ldots = b \\ & n_k \in \mathbb{N} \quad k = 1, \ldots, b \end{array}$$

is given by $n_k = (k + 1)a - b = a - t$, $n_{k+1} = b - ka = t$, and its value is $ka^2 + t^2$.

Proof. By the lemma above, the solution has at most two non zeros, and they are adjacent. Let these be ℓ and $\ell + 1$. The solution is given by the solution to $n_\ell + n_{\ell+1} = a$ and $\ell n_\ell + (\ell + 1)n_{\ell+1} = b$. Given that n has two nonzeros, we can compute

$$n^T A n = k(a - t)^2 + 2k(a - t)t + (k + 1)t^2 = k(a - t + t)^2 + t^2 = ka^2 + t^2. \qquad ◀$$

Lemma 7 is a simple corollary to the above.