Advanced Heuristic and Approximation Algorithms	Grenoble-INP (ORCO)
Lecture 3	
Lecturer: Alantha Newman	*October 12, 2018*

1 Triangle Transversals

Let $G = (V, E)$ be a simple, undirected, unweighted graph. A *triangle* in G is an induced subgraph on the vertex set $\{i, j, k\}$ that contains the edges $\{ij, jk, ik\}$. Let $\mathcal{T}$ denote the set of *triangles* in G. Some well-studied questions are:

1. What is the maximum number of edge disjoint triangles in G?
2. What is the minimum size of a set of edges that intersects all triangles in G?

The first question is a *packing* problem and the second is a *covering* problem and a set of edges that intersects or covers all triangles is sometimes called a *transversal*.

1. $\nu(G)$ denotes the size of a maximum triangle packing of G.
2. $\tau(G)$ denotes the size of a minimum triangle transversal of G.

Observe that $\nu(G) \leq \tau(G)$.

The problem of finding a minimum triangle transversal has an easy 3-approximation algorithm: We can greedily find a triangle t in G, add all edges in t to the solution set F (and remove them from E), and continue until G contains no more triangles. The set of edge disjoint triangles $\mathcal{T}' \subseteq \mathcal{T}$ that we have found at the end of this process is a lower bound on the size of any triangle transversal (i.e., $|\mathcal{T}'| \leq \nu(G) \leq \tau(G)$). Our solution set F has size $|F| = 3|\mathcal{T}'| \leq 3\nu(G) \leq 3\tau(G)$. Therefore, this greedy procedure is a 3-approximation algorithm.

1.1 Linear Programming Relaxation

Consider the following linear programming relaxation for the minimum triangle transversal problem. This is a *covering* LP.

$$\begin{aligned} \min \sum_{e \in E} & x_e \\ \sum_{e \in t} x_e &\geq 1 \text{ for all triangles } t \in \mathcal{T}, \\ x_e &\geq 0. \end{aligned} \tag{$P_{tri\text{-}cover}$}$$

The dual linear program is a relaxation of the triangle packing problem. This is a *packing* LP.

$$\begin{aligned} \max \sum_{t \in \mathcal{T}} & y_t \\ \sum_{t : e \in t} y_t &\leq 1 \text{ for all edges } e \in E, \\ y_t &\geq 0. \end{aligned} \tag{$D_{tri\text{-}pack}$}$$

We use the following notation for the optimal values of these linear programming relaxations on G.

1. $\nu^*(G)$ denotes the value of $(P_{tri\text{-}cover})$.
2. $\tau^*(G)$ denotes the optimal value of $(D_{tri\text{-}pack})$.

Observe that $\nu^*(G) = \tau^*(G)$ by strong duality. Therefore, we have

$$\nu(G) \leq \nu^*(G) = \tau^*(G) \leq \tau(G).$$

1.2 A 2-Approximation Algorithm

Before we present the algorithm, we need some lemmas.

Lemma 1. *G has a triangle transversal $F \subset E$ such that*

1. *$|F| \le \frac{|E|}{2}$, and*
2. *F can be found efficiently.*

Proof. Let $S \subset V$ denote a subset of vertices and let $\bar{S} = V \setminus S$. Note that the set of edges $E \setminus E(S, \bar{S})$ is a triangle transversal. (In fact, it intersects all of the odd cycles in G.)

Now let's show that there exists a set of vertices $S \subset V$ such that

$$|E(S, \bar{S})| \ge \frac{|E|}{2}.$$

Assign each vertex $v \in V$ to the set S with probability $\frac{1}{2}$. Then

$$\begin{aligned} \mathbb{E}[|E(S, \bar{S})|] &= \sum_{ij \in E} (\Pr(i \in S \text{ and } j \notin S) \;+\; \Pr(i \notin S \text{ and } j \in S)) \\ &= \sum_{ij \in E} \left(\frac{1}{4} + \frac{1}{4}\right) \;=\; \frac{|E|}{2}. \end{aligned}$$

Therefore, the expected size of $F = E \setminus E(S, \bar{S})$ is at most $|E|/2$. □

Lemma 2. *Let x^* be an optimal solution for $(P_{tri\text{-}cover})$ on $G = (V, E)$. If $x^*_e > 0$ for all $e \in E$, then*

$$\nu^*(G) \ge \frac{|E|}{3}.$$

Proof. Our goal is to show that when $x^*_e > 0$ for all $e \in E$, we can obtain a large lower bound on $\nu^*(G)$. By complementary slackness, we have that for each edge $e \in E$, $\sum_{t: e \in t} y_t = 1$. In other words,

$$|E(G)| \;=\; \sum_{e \in E} 1 \;=\; \sum_{e \in E} \sum_{t: e \in t} y_t \;=\; \sum_{t \in \mathcal{T}} \sum_{e \in t} y_t \;=\; 3 \sum_{t \in \mathcal{T}} y_t.$$

This implies:

$$\sum_{t \in \mathcal{T}} y_t \;=\; \frac{|E|}{3} \;=\; \nu^*(G).$$

□

Combining Lemmas 1 and 2, we have the following.

Lemma 3. *Let x^* be an optimal solution for $(P_{tri\text{-}cover})$ on $G = (V, E)$ such that $x^*_e > 0$ for all $e \in E$. Then we can efficiently find a triangle transversal F such that*

$$\nu(G) \;\le\; |F| \;\le\; \frac{|E|}{2} \;=\; \frac{|E|}{3}\frac{3}{2} \;=\; \frac{3}{2}\nu^*(G).$$

Now we present an algorithm for finding a triangle transversal.

Triangle-Cover(G)

1. $F \leftarrow \emptyset$.
2. Solve ($P_{tri\text{-}cover}$) on G and let x^* denote an optimal solution.
3. If $x^*_e > 0$ for all $e \in E$, find a triangle cover F such that $|F| \leq \frac{|E|}{2}$.
4. Otherwise:
   (a) For all $e \in E$ such that $x^*_e \geq \frac{1}{2}$,
      i. Add e to F, and
      ii. Delete e from E.
   (b) For all $e \in E$ such that $x^*_e = 0$,
      i. Delete e from E.
   (c) Go to Step 2.
5. Return F.

Theorem 4. Triangle-Cover(G) *returns a triangle transversal F such that $|F| \leq 2\nu^*(G)$.*

Proof. Let $E_{\geq \frac{1}{2}} = \{e \mid x^*_e \geq \frac{1}{2}\}$ and let $E_0 = \{e \mid x^*_e = 0\}$. Now consider the graph $E' = E \setminus (E_0 \cup E_{\geq \frac{1}{2}})$.

We claim that every triangle in G containing an edge $e \in E_0$ also contains an edge in $E_{\geq \frac{1}{2}}$. It remains to find a triangle transveral for triangles with all edges e where $0 < x^*_e < \frac{1}{2}$, which are exactly the edges in E'. Note that x^* restricted to E' is a feasible solution for ($P_{tri\text{-}cover}$) on $G' = (V, E')$. Thus, the optimal solution for ($P_{tri\text{-}cover}$) on G' is at most that of x^* restricted to E'. In other words,

$$\nu^*(G') \leq x^*(E').$$

Then we have

$$|E_{\geq \frac{1}{2}}| \leq 2(x^*(E_{\geq \frac{1}{2}}) - x^*(E')) \quad \Rightarrow \quad |E_{\geq \frac{1}{2}}| \leq 2(\nu^*(G) - \nu^*(G')).$$

We can re-solve for the optimal solution of ($P_{tri\text{-}cover}$) on G' to obtain a new optimal solution x^* and repeat the procedure of adding edges for which $x^*_e \geq \frac{1}{2}$ to $E_{\geq \frac{1}{2}}$ and deleting edges with $x^*_e = 0$ from the graph. Let F denote the union of the sets $E_{\geq \frac{1}{2}}$ over all of these iterations and suppose that on the last iteration, we either have an empty graph or we have a graph $G' = (V, E')$ such that $x^*_e > 0$ for all $e \in E'$. Then $|F| \leq 2(\nu^*(G) - \nu^*(G'))$. Applying Lemma 3 to G', we obtain a triangle transversal F' for G' such $|F'| \leq \frac{3}{2}\nu^*(G')$. Combining F and F' results in a triangle transversal of size at most $2\nu^*(G)$. □

Theorem 4 implies that $\tau(G) \leq 2\nu^*(G)$. The following famous conjecture is due to Tuza.

Conjecture 1. *$\tau(G) \leq 2\nu(G)$.*

This conjecture, if true, is best possible as can be seen by considering $G = K_4$ or $G = K_5$. As we saw earlier, $\tau(G) \leq 3\nu(G)$. Haxell proved that $\tau(G) \leq (3 - \frac{3}{23})\nu(G)$ [Hax99]. For more on triangle packing and covering, see [Kri95].

2 Feedback Arc Set on Tournaments

Let $T = (V, A)$ be a tournament, which is a complete graph in which each edge is oriented in one direction. The goal of the feedback arc set problem is to find a minimum cardinality subset of arcs $F \subset A$

such that $T' = (V, A \setminus F)$ is acyclic. The primal linear program and its dual are shown below. Let $\mathcal{C}$ denote the set of directed cycles in T. Here (i,j) denotes the directed edge or arc from vertex i to vertex j.

Primal (Covering) LP:

$$\begin{array}{lrcl} \min & \sum\limits_{(i,j)\in A} x_{ij} & & \\ \text{subject to:} & \sum\limits_{(i,j)\in c} x_{ij} & \geq & 1, \quad \forall\, c \in \mathcal{C} \\ & x_{ij} & \geq & 0. \end{array}$$

Dual (Packing) LP:

$$\begin{array}{lrcl} \max & \sum\limits_{c\in\mathcal{C}} y_c & & \\ \text{subject to:} & \sum\limits_{c:(i,j)\in c} y_c & \leq & 1, \quad \forall\, (i,j) \in A \\ & y_c & \geq & 0. \end{array}$$

We consider the following algorithm for the feedback arc set problem on tournaments. Note that this problem is equivalent to finding an ordering of the vertices that minimizes the number of backward edges (i.e., the set F comprises the backward edges in the ordering).

> Algorithm: RandomFAS
> *Input:* $T = (V, A)$.
> *Output:* Ordering of vertices in V.
>
> - Choose $i \in V$ at random.
> - if $(j,i) \in A, \;\; \Rightarrow \;\; j \to L$.
> - if $(i,j) \in A, \;\; \Rightarrow \;\; j \to R$.
> - **return** (RandomFAS$(L, A(L))$, RandomFAS$(R, A(R))$).

We now prove that RandomFAS is a 3-approximation algorithm for the problem of feedback arc set on tournaments.

Let $\mathcal{T}$ be the set of directed triangles: a triangle $t = \{i,j,k\}$ belongs to $\mathcal{T}$ if $\{(i,j),(j,k),(k,i)\} \in A$. Note that $\mathcal{T} \subset \mathcal{C}$. Let A_t be the event that one vertex of $t = \{i,j,k\}$ is chosen before triangle t is broken by the algorithm (i.e., when $\{i,j,k\}$ occur in same recursive call). Let p_t be the probability of event A_t. Then, we have:

$$\mathbb{E}[\text{cost of solution}] = \mathbb{E}[\text{number of backward edges}] = \sum_{t\in\mathcal{T}} p_t.$$

In the next lemma, we will show that setting $y'_c = 0$ for all $c \in \mathcal{C}$, where c is not a triangle (i.e. $c \notin \mathcal{T}$), and $y'_t = \frac{p_t}{3}$ for all $t \in \mathcal{T}$ yields the dual feasible solution $\{y'_c\}$.

Lemma 5. *Setting $y'_c = y'_t = \frac{p_t}{3}$, if $c = t \in \mathcal{T}$ and 0 otherwise is dual feasible.*

Proof. Let B_e be the event that edge e is backwards in the output ordering. Let $B_e \wedge A_t$ be the event that edge e is backwards due to A_t. For example, suppose vertices i,j,k form triangle t which is yet unbroken when vertex k is chosen as the pivot. Then we say that edge $e = (i,j)$ is backwards due to event A_t. Since, given event A_t, each edge in t is equally likely to be a backwards edge, we have:

$$\begin{aligned} Pr(B_e \wedge A_t) &= Pr(B_e|A_t)Pr(A_t) \\ &= \frac{1}{3} \times p_t \\ &= \frac{p_t}{3}. \end{aligned}$$

Note that for any $t \neq t' \in \mathcal{T}$ such that $e \in t$ and $e \in t'$, $B_e \wedge A_t$ and $B_e \wedge A_{t'}$ are disjoint events. Hence, $\sum\limits_{t:e \in t} \Pr(B_e \wedge A_t) \leq 1$. This implies that, for all $e \in A$:

$$\sum_{c:e \in c} y'_c = \sum_{t:e \in t} \frac{p_t}{3} \leq 1.$$

We can therefore conclude that $\{y'_c\}$ is a dual-feasible solution. $\square$

Thus, we obtain a 3-approximation algorithm for the problem of feedback arc set on tournaments.

Theorem 6. *The approximation guarantee of* RandomFAS *is* 3.

Proof.

$$\mathbb{E}[\text{cost of solution}] = \sum_{t \in \mathcal{T}} p_t = \sum_{c \in C} 3y_c = 3 \cdot \sum_{c \in \mathcal{C}} y_c \leq 3 \cdot OPT.$$

Since we have shown that the values $\{y'_c\} = \{p_t/3\}$ are dual-feasible, it follows from weak duality that $\sum_{t \in \mathcal{T}} p_t/3 = \sum_{c \in \mathcal{C}} y'_c$ is a lower bound on the size of minimum feedback arc set. $\square$

2.1 Covering Cycles versus Triangles

Let $\{y^*_t\}$ denote an optimal fractional (directed) triangle packing for a tournament. Note that the argument we presented shows:

$$FAS \leq 3 \cdot \sum_{t \in \mathcal{T}} y^*_t.$$

Let FTS denote the a subset of arcs that intersects all directed triangles in a tournament T. By the arguments from Section 1.2, we can show that:

$$FTS \leq 2 \cdot \sum_{t \in \mathcal{T}} y^*_t.$$

This follows from the fact that we can prove an analogue of Lemma 1 for directed triangles in a tournaments: If we take any ordering of the vertices, both the set of forward arcs and the set of backwards arcs intersects each directed triangle, and at least one of these sets contains at most half the arcs.

References

[ACN08] Nir Ailon, Moses Charikar, and Alantha Newman. Aggregating inconsistent information: Ranking and clustering. *Journal of the ACM*, 55(5):1–27, 2008.

[Hax99] Penny E. Haxell. Packing and covering triangles in graphs. *Discrete mathematics*, 195(1-3):251–254, 1999.

[Kri95] Michael Krivelevich. On a conjecture of Tuza about packing and covering of triangles. *Discrete Mathematics*, 142(1-3):281–286, 1995.

The 2-approximation algorithm for triangle transversals can be found in [Kri95]. The algorithm for the feedback arc set problem on tournaments can be found in [ACN08]. These lecture notes are based in part on the following source: lecture notes from a course on Approximation Algorithms at EPFL (`http://theory.epfl.ch/osven/courses/Approx13`).