Covering deficient trees

Guy Even
Tel-Aviv University
guy@eng.tau.ac.il

Guy Kortsarz*
Rutgers University, Camden
guyk@crab.rutgers.edu

Zeev Nutov
The Open University of Israel
nutov@openu.ac.il

Abstract

1 The lower bound and the algorithm

1.1 The Lower Bound

The lower bound in Lemma ?? can be used to obtain a 1.8-approximation algorithm. We now present the lower bound we use to obtain a 1.5-approximation algorithm.

The up-link up(a) of a node a is the link au such that u is as close as possible to the root; under Assumption ??, u is an ancestor of a.

Definition 1.1 Given a node x and a tree T, T is x-closed if x does not have links to nodes outside T.

Definition 1.2 A leaf a₁ is locked if there exists Tᵥ rooted at v with leaves a₁, b₁, b₂ so that b₁, b₂, a₁ are original leave, the link b₁b₂ exists, and a₁ is T'-'closed. The link b₁b₂, is called the locking link of a₁. The tree Tᵥ is called the locking tree.

Figure 1: Illustration to Definition 1.2. Links are shown by dashed lines and paths by wiggly lines.

*Partially supported by NSF grant number 0728787.
**Definition 1.3** Consider a tree of the shape of Figure 4 (A1). A mixed link is defined to be a link $b_1b_2$ in such a tree.

**Definition 1.4** Let $M$ be a maximum matching among matchings with no twin or locking links, and among such matchings, the one to have the least mixed links.

**Lemma 1.1** Let $F, M_F, X$ be as in Lemma ??, Let $M$ be as in Definition ??, Let $J$ be the set of links in $F$ that are not incident to a leaf locked by a link in $F$. Then:

$$|F| \geq \frac{2}{3}|L| - \frac{1}{3}|M| + \frac{\mu}{3} + \frac{1}{3} \sum_{x \in X} \deg_J(x) \quad (1)$$

**Proof:** Let $M'_F$ be the set of locking links in $M_F$ and note that $M_F \setminus M'_F$ is a matching without locking links and without twin links, of size $|M_F| - |M'_F|$. As $M$ is a maximum matching of this type, we have $|M| \geq |M_F| - |M'_F|$. Thus $|M_F| \leq |M| + |M'_F|$. Substituting the last inequality in (??) gives

$$|F| \geq \frac{2}{3}|L| - \frac{1}{3}(|M| + |M'_F|) + \frac{\mu}{3} + \frac{1}{3} \sum_{x \in X} \deg_F(x)$$

$$= \frac{2}{3}|L| - \frac{1}{3}|M| + \frac{\mu}{3} + \frac{1}{3} \left( \sum_{x \in X} \deg_F(x) - |M'_F| \right).$$

Now we observe that the last term in the obtained bound equals the last term in (1), since for any leaf $a_1$ locked by a link $b_1b_2 \in M'_F$, there is a unique link $a_1m(a_1) \in F$ with $m(a_1) \in X$. $\square$

The term $\frac{2}{3}|L| - \frac{1}{3}|M|$ in (1) can be computed in polynomial time. The other terms depend on $F$ which is not known to us, and are partly revealed during the run of the algorithm. The absence of locking links in $M$ is essential, and is used later in Claim ??.

**Tickets. Rule 2: Not claiming tickets in $m(a_i)$ nodes:** We shall need to see each time that we claim a ticket at some vertex $x$, that its not an $m(a)$ vertex.

***** THE FOLLOWING LEMMA ANSWERS THE COMPLAIN OF CHERIYAN *************

**Claim 1.2** A node $m(a)$ can not be the $m$ node of two leaves

**Proof:** Follows directly from the definition of $m(a)$ via a tree of 3 leaves $\square$

**Lemma 1.3** If we have a tree $T'$ with at least 3 leaves and we show that there exists a node $x \in X$ that covers the root, $x$ can not be an $m(a)$ vertex of some $a$

**Proof:** If $a \in T'$ then clearly $m(a)$ does not cover the root in $T'$ as $am(a) \in F$. If $a \notin T'$ we get a contradiction as every tree that contains $m(a)$ and $a$ has at least 4 leaves $\square$

1. The main idea in the algorithm: We do an exhaustive search over all the possible subsets

2I truly recommend to add the line below to the paper
of links $F$ choose in $T$. We show that no matter what subset of edges $F$ has chosen, $T'$ contains a ticket. Thus we can cover $T'$ by the basic cover and leave a coupon. Note that we do not need to know where is the ticket to do that, but only need the proof that no matter what links $F$ chooses, $T'$ contains a ticket.

**Remark:** We may always assume that $T' \neq T = T_r$ as otherwise we do not need to leave a coupon since we cover all the tree. Thus we always have to cover the root of $T'$.

*************** I leave this subsection since I like these definitions you can remove if you want **************

### 1.2 Help nodes

Note that unmatched leaves in $M$ in a minimally semi-closed tree $T'$ are $T'$-closed. Thus for an unmatched leaf $a$ the other endpoint of the tree covering it belongs to $T'$.

**Definition 1.5** A help node in $F$ is a node $x$ that is either a stem, a matched leaf in $M$ or an $m(a_i)$ node that is connected to some unmatched in $M$ leaf $a$. In such a case we say that $x$ helps $a$ (namely, helps $a$ to avoid a ticket).

Note that if an unmatched leaf is not linked to a help node, $T'$ has a ticket and we are done.

**Notation 1.4** We denote the number of matched nodes, plus the number of stems plus the number of $m(a_i)$ nodes in $T'$ by $h$.

**Claim 1.5** If $F$ has a stem and a link from one of the twins to a node that is not the other twin, then $s$ can not help the $|U(T')|$ unmatched nodes. Also, an $m(a_i)$ node can only help $a_i$. If it is touched by another link, there is a ticket at $m(a_i)$.

**Proof:** The first claim follows by shadow minimality. Note that by the definition of a locked leaf the tree has 3 leaves and so $m(a_i)$ can help a unique leaf, thus a second link touching $m(a_i)$ implies a ticket. ??.

*************** THIS LEMMA I USE A LOT **************

**Claim 1.6** $T'$ has at least $|U(T')| - h + 1$ tickets.

**Proof:** Consider the node that covers the root. If a stem covers the root, by shadow minimality it can not be connected to any of the $|U(T')|$ unmatched leaves and we loose a help node. If a node matched by $M$ covers the root in $F$, as its degree in $F$ is 1, we loose a help node. If $m(a)$ exists, by Claim ?? if it covers the root, this gives a ticket. The claim follows as for whichever node that covers the root, either it adds a ticket, or we loose a help node. □

***** ZEEV THIS YOU HAVE TO ADD AFTER MOVING IT TO YOUR LANGUAGE *****
2 Covering deficient trees

2.1 Covering trees of Figure 3 type (A) and (B)

Let $T_v$ be a deficient tree of Type Figure (3) (A) or (B)

Claim 2.1 We can find a non deficient tree that contains $T_v$

Proof: We switch coupons. In the tree of Figure 3 (A) we place 1.5 coupons on $ab_2$ and 1 coupon on $b_1$. In the case Figure 3 (B) we place 1.5 coupons on $ab_1$ and one on $b_2$. In both cases we now have an unmatched leaf that is not $T_v$ closed. Thus any new minimally semi-closed tree found, will be $\{b_1, b_2\}$ closed. If the new $T'$ has one more matched link we clearly can use the basic cover and leave a coupon. In the case of Figure 3 (A) $ab_2$ will be part of the basic cover. This works because all we require for the basic cover to cover the tree is that the contraction of every matched link does not create a leaf. This is true both for $ab_2$ in (A) and $ab_1$ in (B). If no more matched leaves are added then a new unmatched leaf $z$ is added. Consider the original tree $T_v$ trees with the new leaf $z$. Now, $U(T') \geq 3$ and $h = 3$ giving a ticket. □

Claim 2.2 We can reduce the tree in Figure 4 (A2) to the tree from Figure 3 (A) and the tree (B1) (B2) to the tree in Figure 3 (B)

Proof: Consider Figure 4 (B). We add the twin links and pay for this with the coupon of the unmatched twin. Let $x$ be the new node. The tree now has exactly the shape of the tree in Figure 3 (A). Also, as the stem has a link outside $T_v$, the tree is not $x$-close. We still have 2.5 units of credit. Thus we can switch coupons putting 1.5 coupons on $ab_2$, and 1 on $x$ completing the reduction. In case of Figure (B1) we add the twin link and the tree is reduced to the tree of Case Figure 3 (B1) since we still have 2.5 credit. The same goes for (B2). □

2.2 Covering trees Figure 4 type (A1)

Let $a'$ be the second twin. In this case things are different because $a'$ can be locked in $T - w$ and can be covered by $a'm(a')$ without a ticket.

Claim 2.3 The link $aa'$ can not exist

Proof: Note that $b_2$ is linked outside $T_v$ thus a link $a'a$ would not be a locking link. Thus we can add it to $M$, and get a larger matching. Contradiction. □

Claim 2.4 If $b_1$ is linked in $M$ to a node outside $T_v$, we can find a larger tree $T'$ containing $T_v$ and cover it and leave a coupon.

Proof: Add the twin link (paying for it with the coupon of $a'$). Note that we still have 2.5 credit. Let $x$ be the new leaf. Make $x$ an unmatched node giving it one coupon (it is not $T_v$ closed due to $b_1$) Make $ab_2$ the temporary matched link and give it credit 1.5. Note that the contraction of
ab₂ does not create a leaf. When we compute a new sem-closed tree that contains b₂ is will be semi-closed. If it has a new matched link the basic cover will do. Else let z be the new leaf.

The first case is that a’m(a’) ∈ F. Then s is not matched. In such case s is not linked in F. Thus |U(T’)| ≥ 2 because of a and z and h = 2 b₁, b₂ and thus we et a ticket. If a’m(a’) is not used we have |U(T’)| ≥ 3 and h = 3, giving a ticket.

Claim 2.5 Say that Tᵥ is b₁ closed. If the Tᵥ is s closed, the tree is not deficient.

Proof: By the previous claim, b₁ is linked inside Tᵥ. The first possibility is that b₁a’ ∈ F. If s is linked to b₂ there is no node to to cover a without a ticket. Indeed m(a’) does not exist, as a’ was covered by the link a’b₁. This means that as ∈ F is the only way to avoid a ticket. But then contracting of as, b₁a’ gives a leaf x. The leaf has to be covered by another link. Another link touching s can not exist (by shadow minimality). Thus covering x gives a ticket.

The second possibility is that b₁a ∈ F. Then s is not touched by links in F. Thus we have h = 1 (just b₂) and |U(T’)| = 1 (the node a’). This gives a ticket.

The last possibility is that b₁b₂ ∈ F. But then we get two unmatched leaves a, a’ and no help nodes

If b₁ is matched to a node in X, even if a’m(a’) ∈ F, a second link to m(a’) means a ticket as m(a’) can not be the m node of two leaves (see Claim 1.2).

Thus, we may assume that s has a link outside Tᵥ.

Claim 2.6 We can reduce the tree to the case of Figure 3 (A)

Proof: The proof is similar to Claim 2.4 except that a’m(a’) ̸∈ F. Thus when a new leaf z enters the tree we have U(T’) ≥ 3 and h = 3 giving a ticket

Acknowledgment: We thank André Linhares, Zhihan Gao, Joseph Cheriyan, Chaitanya Swamy, and two anonymous referees for many useful comments, and Jon Feldman and Samir Khuller for some useful discussions.

References


