ARBITRARLY LIMITED PACKINGS IN TREES

M. P. Dobson[♭], V. Leoni[♭, †] and G. Nasini[♭, †]

[♭]*Departamento de Matemática, Universidad Nacional de Rosario, Av. Pellegrini 250, Rosario, Argentina*
[†]*Consejo Nacional de Inestigaciones Científicas y Técnicas, Argentina*

Abstract: In a recent work, Gallant et al. introduced the notion of k-limited packing in a graph, generalizing a 2-packing in a graph. They relate L_k, the size of a maximum k-limited packing, with γ, the domination number of the graph and particularly focus on 2-limited packings. They show that all trees satisfying $L_2 = 2\gamma$ can be built via a sequence of certain operations.

In this work, generalizing k-limited packings, we introduce limited packings with arbitrary capacities and forbidden vertices, and derive a linear time algorithm that finds a maximum k-limited packing in a given tree, for any k. A simple modification of our algorithm also decides if a given tree satisfies $L_k = k\gamma$, for any k, in linear time.

Keywords: *k-limited packing, graph theory, algorithm*
2000 AMS Subject Classification: 05C05 - 05C85

1 INTRODUCTION

In this work we consider a location graph problem which models several scenarios, previously considered by Gallant et al. in [3], as the location of sensors in a network security, or of obnoxious facilities (such as garbage) in a city.

If a graph G models the scenario, we consider a subset $\mathcal{A}$ of vertices of G, representing the possible locations for the facilities. We are interested in the subsets of $\mathcal{A}$ satisfying that, for every vertex v, the number of facilities located inside the closed neighborhood of v does not exceed c_v, a non negative integer representing the capacity of v.

Gallant et al. studied the case where $c_v = k$ for every vertex v, and $\mathcal{A}$ is the whole set of vertices of the graph [3]. They called *k-limited packings* the subsets of feasible locations for the facilities, and L_k the size of a k-limited packing of maximum cardinality. Following the terminology and notation in that work, in this work the subsets of feasible locations are called $(\mathbf{c}, \mathcal{A})$*-limited packings* and the size of a $(\mathbf{c}, \mathcal{A})$-limited packing of maximum cardinality is denoted by $L_{\mathbf{c},\mathcal{A}}$.

A *dominant set* D in a graph G is a subset of vertices such that every vertex not in D is adjacent to some vertex in D. Given a cost vector $\mathbf{c}$ associated with the vertices of G, the cost of a dominating set D is $\sum_{v\in D} c_v$, and $\gamma(\mathbf{c})$ denotes the minimum cost over all dominating sets. If $\mathbf{c} = \mathbf{1}$, $\gamma(\mathbf{1}) = \gamma$ is the well-known dominating number of the graph.

By using combinatorial arguments, Gallant et al. [3] proved that $L_k \leq k\gamma$, for any k. They focused on the problem of characterizing trees which satisfy this inequality by equality when $k = 2$, and showed that any of such trees can be built via a sequence of certain operations. Two of these operations require that one knows if a vertex is not in some maximum 2-limited packing. Unfortunately, from the computational complexity point of view, their implementation could be "as difficult" as finding the number L_2 itself, or as finding a 2-limited packing of maximum cardinality. This is one of the reasons that motivates us to focus on the problem of finding a $(\mathbf{c}, \mathcal{A})$-limited packing of maximum cardinality on trees. We refer this problem as ALPT.

It is known that $\gamma(\mathbf{c})$ can be obtained in polynomial time on trees, for any integer vector $\mathbf{c}$ [2]. It can also be proved that, in the context of totally balanced matrices, ALPT can be solved in time $O(|V(G)|^3)$.

In this work we prove that ALPT can be solved in *linear* time and show that, by a simple adaptation of the involved algorithm, we can also decide in linear time if $L_k = k\gamma$, for any k.

2 DEFINITIONS AND FIRST RESULTS

Throughout this work, graphs are simple and for a graph G, $V(G)$ and $E(G)$ denote respectively its vertex and edge sets. For $v \in V(G)$, $N(v)$ and $N[v]$ denote the (open) *neighborhood* and *closed neighborhood* of v, respectively. The *degree* of v in G is defined as $|N(v)|$.

Given a subset of vertices $R \subseteq V(G)$, $G \setminus R$ denotes the induced subgraph of G obtained by the *deletion* of the vertices in R, that is, the subgraph with vertex set $V(G) - R$ and edge set, the subset of $E(G)$ of edges with both endpoints included in $V(G) - R$.

We formally present a generalization of k-limited packings:

Definition 1 *Let G be a graph, $\mathbf{c} \in \mathbb{Z}_+^{V(G)}$ and $\mathcal{A} \subseteq V(G)$. A set of vertices $B \subseteq V(G)$ is a $(\mathbf{c}, \mathcal{A})$-*limited packing *in G if $B \subseteq \mathcal{A}$ and $|B \cap N[v]| \leq c_v$, for every $v \in V(G)$.*

Given a graph G, let us consider the $0, 1$ matrix $N = (a_{ij})$ with columns and rows indexed by $V(G)$ and where $a_{ij} = 1$ if and only if $j \in N[i]$. Clearly, N is a symmetric matrix.

The numbers $L_{\mathbf{c},\mathcal{A}}$ and $\gamma(\mathbf{c})$ can be obtained as the optimal values of the following integer programs:

$$L_{\mathbf{c},\mathcal{A}} = \max\{\mathbf{1}x : Nx \leq \mathbf{c},\ x_v = 0,\ \forall v \notin \mathcal{A},\ x \in \{0,1\}^{V(G)}\},$$

and

$$\gamma(\mathbf{c}) = \min\{\mathbf{c}x : Nx \geq \mathbf{1},\ x \in \{0,1\}^{V(G)}\}.$$

By linear programming arguments, it can be proved that $L_{\mathbf{c},V(G)} \leq \gamma(\mathbf{c})$.

It is known [2] that, when G is a tree, N is a *totally balanced matrix*, i.e, every hole submatrix of N is the 2×2 submatrix of all 1s. It is also known that the rows and columns of a totally balanced matrix of order $m \times n$ can be permuted into *standard greedy form* in time $O(nm^2)$ ([1], [4] and [5]). Besides, if a matrix is in standard greedy form, any linear program associated with it can be solved by a greedy algorithm. Moreover, when every right hand side in the restrictions involved in the linear program is an integer number, the optimal solution given by a greedy algorithm is integral.

The previous results allow us to conclude that the calculus of the parameters $L_{\mathbf{c},\mathcal{A}}$ and $\gamma(\mathbf{c})$, as well as a certificate that the equality between them is satisfied for $\mathbf{c} = k\mathbf{1}$, can be obtained in polynomial time on trees.

Moreover, from *complementary slackness*, the conditions for optimality stated by Gallant et al. in [3] can be generalized in the following way:

Lemma 1 *Given a graph G, $\mathcal{A} \subseteq V(G)$ and $\mathbf{c} \in \mathbb{Z}_+^{V(G)}$, $L_{\mathbf{c},\mathcal{A}} = \gamma(\mathbf{c})$ if and only if for any maximum $(\mathbf{c}, \mathcal{A})$-limited packing B in G and any minimum cost dominating set D in G, the following statements hold:*

1. *for any $b \in B$, $|N[b] \cap D| = 1$,*
2. *for any $d \in D$, $|N[d] \cap B| = c_d$.*

In the next section we provide a linear algorithm which solves ALPT and, in the particular case where $\mathbf{c} = k\mathbf{1}$ and $\mathcal{A} = V(T)$, it decides whether the equality $L_k(T) = k\gamma(G)$ holds.

3 Towards the algorithms

Given a tree T, a *leaf* is a vertex with degree one and a *stem* is a vertex having a leaf in its neighborhood. $L(T)$ and $S(T)$ denote the leaf and stem sets of T, respectively. For $l \in L(T)$, $s(l)$ denotes the stem adjacent to l. Also, for each $v \in S(T)$, $L(v) = L(T) \cap N(v)$ and $n_v = |L(v)|$.

We can state the following result, whose proof is omitted due to the space constraints:

Theorem 1 *Let T be a tree, $\mathbf{c} \in \mathbb{Z}_+^{V(T)}$ and $\mathcal{A} \subseteq V(T)$, with $L(T) \subseteq \mathcal{A}$. The following items hold:*

1. *if $l \in L(T)$ with $c_l = 0$ then, B is a maximum $(\mathbf{c}, \mathcal{A})$-limited packing in T if and only if B is a maximum $(\mathbf{c}, \mathcal{A} - \{l, s(l)\})$-limited packing in $T \setminus \{l\}$;*
2. *if $v \in S(T)$ with $c_v < n_v$ and $c_l \geq 1$ for every $l \in L(v)$ then, B is a maximum $(\mathbf{c}, \mathcal{A})$-limited packing in T if and only if B is a maximum $(\mathbf{c}, \mathcal{A} - H_v)$-limited packing in $T \setminus H_v$, where $H_v \subseteq L(v)$ with $|H_v| = n_s - c_s$;*

3. *if* $c_l \geq 1$ *for all* $l \in L(T)$ *and* $c_s \geq n_s$ *for every* $s \in S(T)$*, then there exists a maximum* $(\mathbf{c}, \mathcal{A})$*-limited packing* B *in* T *verifying:*
   - $L(T) \subseteq B$*, and*
   - $B - L(T)$ *is a maximum* $(c', \mathcal{A}')$*-limited packing in* $T \setminus L(T)$*, where*
     - $c'_v = c_v - n_v$ *for every* $v \in S(T)$*,* $c'_v = c_v$ *otherwise, and*
     - $\mathcal{A}' = \mathcal{A} - (L(T) \cup \{s(l) : c_l = 1,\ l \in L(T)\})$.

This theorem yields a linear time algorithm to solve ALPT. The proposed algorithm incorporates in each iteration, the set of allowed leaves to the set B, actualizes the capacity of the leaves and stems, and finally deletes the leaves. All these tasks can be done in linear time on the size of $L(T)$. Clearly, in each iteration, we deal with a tree with at least two vertices less than the quantity of vertices in the tree in the previous iteration. Since the number of iterations is at most the height of the given tree T, the algorithm is linear on the size of $V(T)$.

In order to answer the question whether a given tree T satisfies $L_k(T) = k\gamma(T)$, for certain $k \in \mathbf{Z}_+$, we adapt our algorithm, by starting with $\mathcal{A} = V(G)$ and $\mathbf{c} = k\mathbf{1}$ and making use of the following result, whose proof is also omitted due to the space constraints:

Theorem 2 *Given a tree* T*, there exists a dominating set* D *of minimum cardinality such that* $S(T) \subseteq D$ *and* $D - S(T)$ *is a minimum dominating set of* $T \setminus \bigcup_{v \in S(T)} N[v]$.

The adapted version of the primary algorithm incorporates in each iteration, vertices to the set D, and checks whether any of the conditions in lemma 1 is violated. If so, it stops. If both conditions are always fulfilled, it finishes having computed the numbers $L_k(T)$ and $\gamma(T)$ in linear time.

REFERENCES

[1] R. ANSTEE AND M. FARBER, *Characterization of totally balanced matrices*, Journal of Algoritms **5** (1984), 215-230.

[2] M. FARBER, *Domination, independent domination, and duality in strongly chordal graphs*, Discrete Applies Mathematics **7** (1984), 115-130.

[3] R. GALLANT, G. GUNTHER, B. HARTNELL AND D. RALL, *Limited Packings in graphs*, Electronic Notes in Discrete Mathematics **30** (2008), 15-20, and to appear in Discrete Applied Mathematics.

[4] A. J. HOFFMAN, A. KOLEN AND M. SAKAROVITCH, *Characterizations of totally balanced and greedy matrices*, SIAM Journal of Algebraic and Discrete Methods **6** (1985), 721-730.

[5] A. LUBIW, *Doubly lexical ordering of matrices*, SIAM Journal of Computing **16** (1987), 854-879.