点容量付き内向木詰込問題の計算複雑度

今堀 慎治　宮本 裕一郎　橋本 英樹　佐々木 美裕　柳浦 睦憲
東京大学　上智大学　名古屋大学　南山大学　名古屋大学

概要 本研究では，有向グラフ，根，点容量関数，枝消費関数が与えられたとき，根付き全域内向木をいくつ詰め込めるかという問題を扱う．枝消費関数によって根付き全域内向木の各点における消費が決まる．制約は，各点において，根付き内向木の消費の合計が点容量を超えないことである．この問題を点容量付き内向木詰込問題とよぶことにする．この問題は，センサーネットワークの研究における最も重要な問題である，ネットワークライフタイム問題の1つである．本研究では，グラフが非巡回的な場合，消費関数が距離に依存する場合，などにおける点容量付き内向木詰込問題の計算複雑度を明らかにした．

The Hardness of the Vertex Capacitated Directed-in-Tree Packing Problem

Shinji IMAHORI　Yuichiro MIYAMOTO　Hideki HASHIMOTO
Univiversity of Tokyo　Sophia Univiversity　Nagoya Univiversity

Mihiro SASAKI　Mutsunori YAGIURA
Nanzan Univiversity　Nagoya Univiversity

Abstract In this paper, we deal with a vertex capacitated arborescence packing problem. The input consists of a directed graph, a root vertex, a vertex capacity function and edge consumption functions. The problem is to find the maximum number of rooted arborescences such that the total consumption of arborescences at each vertex does not exceed the capacity of the vertex. The problem is one of the network lifetime problems that are among the most important issues in the context of sensor networks. We reveal the computational complexity of the problem in several cases; e.g., the given graph is acyclic or not, the instance is metric dependent.

1 Introduction

In this paper, we consider a *vertex capacitated arborescence packing problem.* The input consists of a directed graph, a root vertex, a vertex capacity function and edge consumption functions. The problem is to find the maximum number of rooted arborescences such that the total consumption of arborescences at each vertex does not exceed the capacity of the vertex.

Formally, let $G = (V, E)$ be a digraph. We call a subset $A \subseteq E$ spanning *arborescence* if (V, A) is a directed spanning rooted in-tree. Let $\mathbb{R}_+$ be the set of nonnegative real numbers. Let $h : E \to \mathbb{R}_+$ and $t : E \to \mathbb{R}_+$ be a head and a tail consumption function on directed edges, respectively. The consumption $c(A, v)$ of an arborescence A at a vertex $v \in V$ is defined as

$$c(A,v) := \sum_{e\in\delta_A^-(v)} h(e) + \sum_{e\in\delta_A^+(v)} t(e),$$

where $\delta_{E'}^-(v)$ (resp., $\delta_{E'}^+(v)$) is the set of edges in E' entering (resp., leaving) v. We call the first term of the above equation *head consumption*, and the second term *tail consumption.* Let $b : V \to \mathbb{R}_+$ be a vertex capacity function. The vertex capacitated arborescence packing problem is to find the maximum number of arborescences $\mathcal{A}$ rooted at the given root $r \in V$ such that

$$\sum_{A\in\mathcal{A}} c(A,v) \le b(v),\ \forall v \in V.$$

Note that the set of arborescences $\mathcal{A}$ is a multiset; i.e., it may include same arborescences.

The aim of this paper is to reveal the computational complexity of several variations of the problem. Our results are summarized as follows:

- packing one arborescence
  - without head consumptions: polynomially solvable
  - with head consumptions on acyclic graphs: strongly NP-hard
  - on complete graphs embedded in a space with head consumptions depending only on the distance between end vertices: strongly NP-hard
- packing arborescences (in general)
  - without head consumptions
    - on acyclic graphs: polynomially solvable
    - on general graphs: strongly NP-hard
    - on complete graphs embedded in a space with tail consumptions depending only on the distance between end vertices: NP-hard
  - with head consumptions: strongly NP-hard

Recently, several kinds of graph packing problems are studied in the context of ad hoc wireless networks and sensor networks. They are called *network lifetime problems*. Among the important problems in this category are the vertex capacitated spanning subgraph packing problems considered in [1]. In their formulation, head consumptions are not considered, and the consumption at each vertex is the maximum tail consumption among the edges leaving the vertex. There are variations of the problem with respect to additional conditions on the spanning subgraph such as strong connectivity, symmetric connectivity, and directed out-tree rooted at a given vertex. They discussed the hardness of the problem and proposed several approximation algorithms. Our problem without head consumptions is another kind of the vertex capacitated spanning subgraph packing problem; the spanning subgraphs are arborescences.

These network lifetime problems, including our problem, are similar to the well-known *edge-disjoint spanning arborescence packing problem*: Given a directed graph $G = (V, E)$ and a root r in V, find the maximum number of edge-disjoint spanning arborescences rooted at r. Note that the given graph G may have parallel edges. The edge-disjoint spanning arborescence packing problem is solvable in polynomial time [2, 6]. Its capacitated version is also solvable in polynomial time [3, 7, 8].

The rest of this paper is organized as follows. In Section 2, we show the computational complexity of a special case of our problem, packing one arborescence. In Section 3, we show the complexity of the problem in general.

In this paper, we assume that the given root vertex is reachable from all other vertices; otherwise the vertex capacitated arborescence packing problem is obviously infeasible.

2 Packing One Arborescence

In this section, we consider the decision problem of the vertex capacitated one arborescence packing problem; given an instance of the vertex capacitated arborescence packing problem, decide whether the instance has a packing with one arborescence.

We first consider the problem without head consumptions; i.e., $h(e) = 0$, $\forall e \in E$. In this case, the consumption of an arborescence at each vertex is caused by exactly one edge leaving the vertex. Let $E' \subseteq E$ be the set of all edges whose tail consumptions are at most the capacity of their tail vertices; i.e., $\forall e' \in E',\ t(e') \leq b(\mathrm{tail}(e'))$, where tail($e$) is the tail vertex of a directed edge e. Because the vertex capacitated one arborescence packing problem on (V, E) is equivalent to the problem of finding an arborescence in (V, E'), the following lemma holds.

Lemma 1 *The vertex capacitated one arborescence packing problem without head consumptions is solvable in $O(|E|)$ time.*

Next we consider the problem with head consumptions. In this case, the decision problem of the vertex capacitated one arborescence packing problem is strongly NP-hard.

Theorem 1 *The decision problem of the vertex capacitated one arborescence packing problem is strongly NP-hard. The problem is still strongly NP-hard, even when the given graph is acyclic and there are no tail consumptions.*

Proof (Outline): We show that the bin-packing problem polynomially transforms to the above decision problem without tail consumptions. The decision problem of the bin-packing is known to be strongly NP-complete [4].

Let $\{1, \ldots, k\}$ be the set of bins whose capacities are 1. Let n be the number of items to be packed into bins and s_i $(i = 1, \ldots, n)$ be the sizes of items satisfying $0 \leq s_i \leq 1$.

For the set of k bins, we introduce a set of vertices $U := \{u_1, \ldots, u_k\}$. For each item i, we introduce a vertex w_i. Let $W := \{w_1, \ldots, w_n\}$. We also introduce a root vertex r. Edges are emanating from each vertex in W to all vertices in U, and emanating from each vertex in U to the root r. We set a head consumptions $h(e) := s_i$ if the edge e is leaving $w_i \in W$. We set the capacity of a vertex to 1 if the vertex corresponds to a bin, and 0 otherwise. See Figure 1 for an example. Then the arborescence packing has a feasible packing of size one if and only if the bin-packing has a feasible packing. □

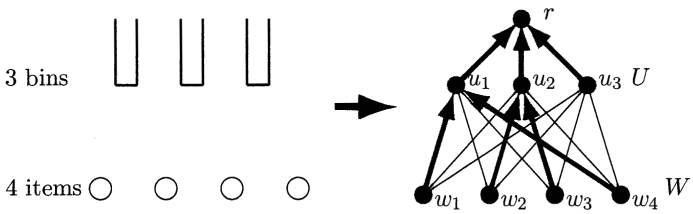

Figure 1: An example of the transformation from the bin-packing

The problem is strongly NP-hard when the graph is complete and embedded in the 1-dimensional space $\mathbb{R}^1$, and the head consumption of each edge depends only on the distance between its end vertices. Though we showed that the bin-packing problem polynomially transforms to this special case, we omit the details.

3 Packing Arborescences (in General)

In this section, we deal with the general case of the vertex capacitated arborescence packing problem. From the results in the previous section, the vertex capacitated arborescence packing problem with head consumptions is strongly NP-hard. We therefore concentrate on the case without head consumptions in this section.

We first consider the case that the given graph is acyclic. When a given graph is acyclic, an arborescence of the graph is easily found.

Proposition 1 *When a graph $G = (V, E)$ is acyclic, a set of edges $E' \subseteq E$ is an arborescence if the outdegree in (V, E') is one for all vertices except for the root.*

From Proposition 1, we can easily see that the vertex capacitated arborescence packing problem without head consumptions is solvable in $O(|E|)$ time.

Lemma 2 *When the given graph is acyclic and there are no head consumptions, the vertex capacitated arborescence packing problem is solvable in $O(|E|)$ time.*

Proof: Let $G = (V, E)$ be the given graph. Let $r \in V$ be the given root vertex. For all $v \in V \setminus \{r\}$, we let $e_{\min}(v)$ be an edge leaving v such that the tail consumption is minimum among all edges leaving v; i.e.,

$$e_{\min}(v) := e \in \delta_E^+(v) \quad \text{such that} \quad t(e) \leq t(e'),\ \forall e' \in \delta_E^+(v).$$

Then the set of edges $\{e_{\min}(v) : v \in V \setminus \{r\}\}$ is an arborescence from Proposition 1. □

Theorem 2 *The following problem is strongly NP-hard: Given an instance of the vertex capacitated arborescence packing problem and a number n, decide whether the instance has a packing of size n. The problem is still strongly NP-hard, even when there are no head consumptions.*

Proof (Outline): We show that the bin-packing problem (known to be strongly NP-complete) polynomially transforms to the above decision problem without head consumptions.

Let $\{1, \ldots, k\}$ be the set of bins whose capacities are 1. Let n be the number of items to be packed into bins and s_i $(i = 1, \ldots, n)$ be the sizes of items satisfying $0 \leq s_i \leq 1$.

For the set of k bins, we introduce a set of vertices $U := \{u_1, \ldots, u_k\}$. For an item i, we introduce a vertex w_i. Let $W := \{w_1, \ldots, w_n\}$. We also introduce a root vertex r. Edges are emanating from each vertex in W to the root r, and connecting vertices in U and W each other, and connecting all vertices in U. We set a tail consumption s_i to edges entering w_i, and set a tail consumption 1 to edges entering the root. We set the capacity of each vertex to 1 except for the root. See Figure 2 for an example. Then the arborescence packing has a feasible packing of size n if and only if the bin-packing has a feasible packing. □

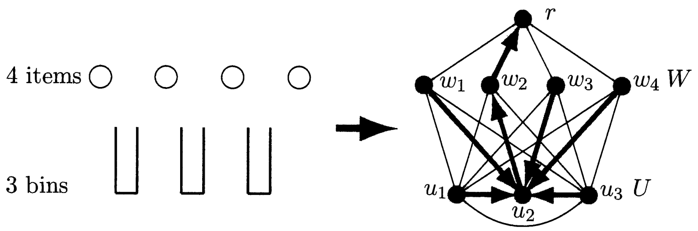

Figure 2: An example of the transformation from the bin-packing

In the following, we show that the vertex capacitated arborescence packing problem is still hard when the problem is *metric dependent*.

Definition 1 (Vertex Capacitated Metric Dependent Arborescence Packing) *A vertex capacitated metric dependent arborescence packing problem is a special case of the vertex capacitated arborescence packing problem such that the given graph is complete and embedded in the d-dimensional space $\mathbb{R}^d$ and that the tail consumption $t((v, w))$ of each edge (v, w) depends only on the distance between its end vertices v and w.*

To show the hardness of the problem in this case, we define a *Difference-Increasing Sequence Partition problem* (DIS Partition), which is a special case of the set partition problem.

Definition 2 (Difference-Increasing Sequence Partition Problem) *Given a set of numbers* $S = \{s_1, s_2, \dots, s_n\}$ *such that*

$$0 < s_1 < s_2/2, \tag{1}$$
$$s_i - s_{i-1} < s_{i+1} - s_i \ (i = 2, 3, \dots, n-1), \tag{2}$$

the difference-increasing sequence partition problem is to decide whether there exists a subset $S' \subseteq S$ *such that*

$$\sum_{s_i \in S'} s_i = \sum_{s_j \notin S'} s_j.$$

Lemma 3 *DIS Partition is NP-complete.*

Proof (Outline): The set partition problem, known to be NP-complete [4], transforms to DIS Partition in linear time. □

Let $(x_1(v), \dots, x_d(v)) \in \mathbb{R}^d$ be the d-dimensional coordinate of the vertex v. We denote the L^k norm between v and w as $L^k(v,w) = (|x_1(v) - x_1(w)|^k + \cdots + |x_d(v) - x_d(w)|^k)^{1/k}$.

Theorem 3 *The following problem is NP-hard for any fixed constant* $k \in \{1, \dots, \infty\}$*: Given a number* n *and an instance of the vertex capacitated metric dependent arborescence packing problem whose tail consumption is* L^k *norm between its end vertices, decide whether the instance has an arborescence packing of size* n*. The problem is still NP-hard, even when there are no head consumptions and the given graph is embedded in the 1-dimensional space* $\mathbb{R}^1$*.*

Proof: First, we show that DIS Partition polynomially transforms to the above problem such that there are no head consumptions and the given graph is embedded in $\mathbb{R}^1$. DIS Partition was shown to be NP-complete in Lemma 3.

Let $S = \{s_1, s_2, \dots, s_n\}$ be an instance of DIS Partition. For each number s_i, we introduce a vertex u_i whose coordinate is s_i. Let $U := \{u_1, \dots, u_n\}$. We introduce a set of vertices $W := \{w, u_0\}$ whose coordinates are 0, and we introduce a root vertex r whose coordinate M is sufficiently large. All pairs of vertices are connected each other. The tail consumption of each edge is the distance between its end vertices. The capacity $b(v)$ of each vertex v is defined as follows:

$$b(v) := \begin{cases} 0 & (v = r), \\ \sum_{i=1}^{n} s_i/2 & (v \in W), \\ M + (n-2)s_j - (n-1)s_{j-1} & (v = u_j \in U), \end{cases}$$

where $s_0 = 0$. See Figure 3 for an example ($n = 4$).

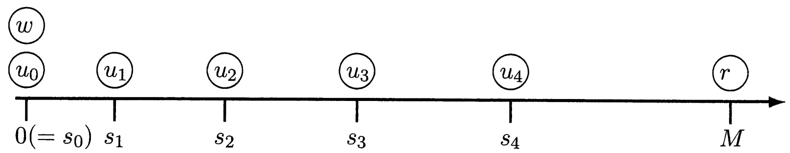

Figure 3: An example of embedded vertices

Suppose the metric dependent arborescence packing has a packing $\{A_1, \dots, A_n\}$. Because M is sufficiently large, edges connecting vertices in W and r never appear in the arborescences. Because $b(u_i) = M + (n-2)s_i - (n-1)s_{i-1}$ ($u_i \in U$) (and M is sufficiently large), for an index i, an edge (u_i, r) appears at most once in the arborescences. From the setting of $b(u_i)$, once the edge (u_i, r)

is used, (u_i, u_{i-1}) must be used in the rest of arborescences (note that u_{i-1} is the nearest vertex of u_i). As a result, for $i = 1, \dots, n$, $A_i = \{(u_i, r), (u_0, u_i), (w, u_0)\} \cup \{(u_j, u_{j-1}) : j > 0, j \neq i\}$ or $A_i = \{(u_i, r), (w, u_i), (u_0, w)\} \cup \{(u_j, u_{j-1}) : j > 0, j \neq i\}$. See Figure 4 for an example ($i = 3$). Then DIS Partition has a feasible subset S' composed of numbers s_i such that (u_0, u_i) appears in

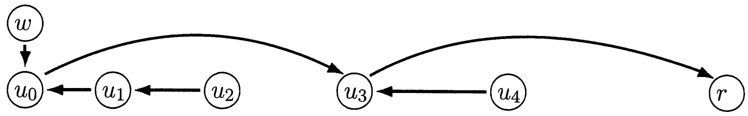

Figure 4: An arborescence of a feasible packing

these arborescences. Similarly, the opposite holds.

When the graph is embedded in $\mathbb{R}^d$ and the tail consumption is L^k norm, the above 1-dimensional case is included. □

In Theorem 3, we assumed that the tail consumption of the edge (v, w) is $L^k(v, w)$. We extended the above result to the case that the tail consumption is $(L^k(v, w))^\alpha$ for any positive constant α. We omit the details here because the idea is similar to the above proof. These assumptions are quite natural in the context of wireless radio networks, because the radio intensity decreases according to the distance [5]. In other words, as the distance between vertices becomes larger, a radio transmitter consumes more energy.

We leave open the *strong* NP-hardness of the vertex capacitated metric dependent arborescence packing problem without head consumptions.

References

[1] G. Calinescu, S. Kapoor, A. Olshevsky, and A. Zelikovsky. Network lifetime and power assignment in ad hoc wireless networks. In *Proceedings of the 11th European Symposium on Algorithms*, Vol. 2832 of *Lecture Notes in Computer Science*, pp. 114–126. Springer, 2003.

[2] J. Edmonds. Edge-disjoint branchings. In *Combinatorial Algorithms*, pp. 91–96. Algorithmic Press, 1973.

[3] H. N. Gabow and K. S. Manu. Packing algorithms for arborescences (and spanning trees) in capacitated graphs. *Mathematical Programming*, 82:83–109, 1998.

[4] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, 1979.

[5] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan. An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1:660–670, 2002.

[6] B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Algorithms.* Springer-Verlag, 4th edition, 2007.

[7] W. Mader. On n-edge-connected digraphs. *Annals of Discrete Mathematics*, 17:439–441, 1983.

[8] A. Schrijver. *Combinatorial Optimization.* Springer-Verlag, 2003.