IEICE TRANS. INF. & SYST., VOL.E99–D, NO.10 OCTOBER 2016

PAPER

A Linear Time Algorithm for Finding a Spanning Tree with Non-Terminal Set V_{NT} on Cographs

Shin-ichi NAKAYAMA[†a)] ***and*** **Shigeru MASUYAMA**[††], ***Members***

SUMMARY Given a graph $G = (V, E)$ where V and E are a vertex and an edge set, respectively, specified with a subset V_{NT} of vertices called a *non-terminal set*, the spanning tree with non-terminal set V_{NT} is a connected and acyclic spanning subgraph of G that contains all the vertices of V where each vertex in a non-terminal set is not a leaf. In the case where each edge has the weight of a nonnegative integer, the problem of finding a minimum spanning tree with a non-terminal set V_{NT} of G was known to be NP-hard. However, the complexity of finding a spanning tree on general graphs where each edge has the weight of one was unknown. In this paper, we consider this problem and first show that it is NP-hard even if each edge has the weight of one on general graphs. We also show that if G is a cograph then finding a spanning tree with a non-terminal set V_{NT} of G is linearly solvable when each edge has the weight of one.

key words: *spanning tree, cograph, algorithm*

1. Introduction

Consider a graph $G = (V, E)$ and a function w from its edge set to the set of nonnegative integers. By V and E we denote the vertex and edge sets of G, respectively. For any subgraph $G_i = (V_i, E_i)$ of G where V_i and E_i are the vertex and edge sets of G_i, let $w(G_i) = \sum_{e \in E_i} w(e)$ be its weight.

Given a graph G and subset V_{NT} of its vertices called a *non-terminal set*, a minimum spanning tree with a non-terminal set (MSTNT) is a connected and acyclic spanning subgraph of G that contains all the vertices of V with the minimum weight where each vertex in the non-terminal set is not a leaf [8]. Zhang and Yin [8] showed that the problem for finding an MSTNT is NP-hard and describe an approximation algorithm for finding an MSTNT on general graphs. This problem can be applied to the design of computer networks where the devices used for relays and those used for terminals are different.

In this paper, we only consider graphs with $w(e) = 1$ for each edge. We first prove that, on general graphs, the problem for finding a spanning tree with a non-terminal set (STNT) where $w(e) = 1$ for each edge is also NP-hard. Therefore, we assume that a given graph always has a weight of one and that the weight $w(e)$ is omitted unless otherwise specified. As finding an STNT on general graphs is NP-hard, we restrict a given graph to be a cograph. The class of cographs has been intensively studied since their definition by Seinsche [7]. Cographs are recursively defined as the class of graphs constructed from a single vertex under the closure of the operations of disjoint union and join operations [1], [4]. A cograph has a unique tree representation, called a cotree, which shows how the cograph can be recursively constructed. This provides the basis for fast polynomial time algorithms for problems such as isomorphism, colouring, clique detection, clustering, minimum weight dominating sets, minimum fill-in and Hamiltonicity [1]–[3]. Here we show that for a cograph G, a linear time (that is, $O(|V| + |E|)$ time) algorithm for finding an STNT of G exists.

Manuscript received January 14, 2016.
Manuscript revised June 7, 2016.
Manuscript publicized July 12, 2016.

[†]The author is with the Department of Mathematical Sciences, Faculty of Integrated Arts and Sciences, Tokushima University, Tokushima-shi, 770–8502 Japan.
[††]The author is with Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi-shi, 441–8580 Japan.
a) E-mail: shin@tokushima-u.ac.jp

DOI: 10.1587/transinf.2016EDP7021

2. Complexity of the STNT Problem

In this section, we prove that the problem for finding an STNT on general graphs is NP-hard.

In the problem for finding STNT $\mathcal{T}$ of $G = (V, E)$, V is divided into two disjoint vertex sets; one is a *non-terminal set* V_{NT} and the other is a *potential terminal set* V_T in which each vertex may, but not necessarily be, a leaf of $\mathcal{T}$.

Theorem 1: On general graphs, the problem for finding an STNT is NP-hard.

Proof. We prove this by polynomial-time reduction from the Hamiltonian path problem that is NP-hard, to our problem. An instance I of the Hamiltonian path problem is as follows: Given a graph $G = (V, E)$ where $V = \{s, v_1, \cdots, v_{n-2}, t\}$ and $E \subset V \times V$, find a path P from s to t that passes through each vertex of $V - \{s, t\}$ exactly once.

We transform an instance I of the Hamiltonian path problem to an instance I' of the STNT problem. On an instance I' of the STNT problem, let s and t be potential terminals of G and $\{v_1, \cdots, v_{n-2}\}$ be the set V_{NT} of non-terminal vertices of G. We show that an instance I of the Hamiltonian path problem has a solution if and only if an instance I' of the STNT problem has a solution.

To prove the only-if part, we assume that an instance I of the Hamiltonian path problem has a solution, that is, a Hamiltonian path $P = s, v_1, \cdots, v_{n-2}, t$ exists. This is obviously a spanning tree of G. Since P is a path, only s and t are leaves. s and t are potential terminals, then $P = \{s, v_1, \cdots, v_{n-2}, t\}$ is an STNT of G. It means that if I has a solution, then the I' of the STNT problem on G has a solution.

To prove the if-part, we next assume that, on an instance I', G has an STNT $\mathcal{T}$. Since $V_{NT} = \{v_1, \cdots, v_{n-2}\}$ and potential terminals are only s' and t', each vertex in $\{v_1, \cdots, v_{n-2}\}$ is not a leaf of the STNT. Therefore, the STNT is a path starting from s to t. For the above reasons, $P = \{s, \cdots, t\}$ is a Hamiltonian path on G.

Consequently, as the Hamiltonian path problem is NP-hard, the STNT problem is NP-hard. □

3. Cographs and Cotrees

A cograph can be constructed from isolated vertices by consecutive application of disjoint union and join operations [1]. We introduce here a cotree that is a tree where each of its internal vertices is labeled with either number 0 or 1. On cotree T^C, we call a vertex labeled 1 and 0 a *label*-1 and *label*-0 *vertex*, respectively. Every cotree T^C defines a cograph G having the leaves of T^C as vertices, and in which the subtree rooted at each vertex v of T^C corresponds to the induced subgraph in G defined by the set of leaves descending from v. A subgraph corresponding to a subtree of T^C rooted at vertex v is denoted by G_v.

- A subtree of T^C consisting of a single leaf corresponds to an induced subgraph of G with a single vertex.
- A subtree of T^C rooted at a label-0 vertex v corresponds to the disjoint union of subgraphs of G defined by the children of v.
- A subtree of T^C rooted at a label-1 vertex v corresponds to the join of subgraphs of G defined by the children of v; that is, we construct the disjoint union and add an edge between each pair of two vertices corresponding to leaves in the different subtrees of T^C. (As illustrated in Fig. 2 (a), label-1 vertex v_2^1 of cotree T^C has two children corresponding to subgraphs $G_{v_4^0}$ with three isolated vertices 8, 9, 10 and $G_{v_5^0}$ with two isolated vertices 11, 12. Then, the join operation of $G_{v_4^0}$ and $G_{v_5^0}$ are executed at label-1 vertex v_2^1 of cotree T^C, that is, we add edges $\{8, 11\}$, $\{8, 12\}$, $\{9, 11\}$, $\{9, 12\}$, $\{10, 11\}$ and $\{10, 12\}$.)

In the following, let $G = (V, E)$ be a cograph with a vertex set $V = V_{NT} \cup V_T$ and $\widehat{G} = (V_{NT}, \widehat{E})$ be a cograph induced by a non-terminal set V_{NT}.

Figures 1 (a) and (b) illustrate a cograph G and a cograph $\widehat{G}$, respectively. Moreover, Figs. 2 (a) and (b) illustrate a cotree T^C of G and a cotree $\widehat{T}^C$ of $\widehat{G}$, respectively. On G shown in Fig. 1 (a), vertices 6, 8, 15 are in V_T and other vertices are in V_{NT}.

$\widehat{G}$ is derived from G by removing potential terminals and each edge that is adjacent to them. For deriving a cotree $\widehat{T}^C$ of $\widehat{G}$ from T^C of G, we first remove leaves v_p's corresponding to potential terminals and all edges adjacent to such leaves v_p's. After this process, we remove vertices whose descendants have no non-terminal vertex, so we derive $\widehat{T}^C$. Thus, each vertex v of $\widehat{T}^C$ corresponds to v of T^C.

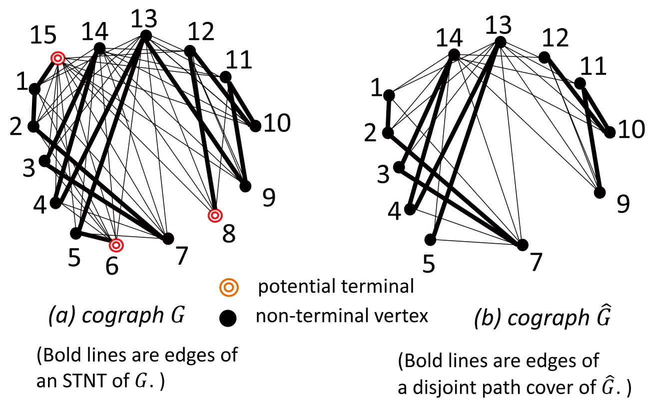


Fig. 1 Cographs G and $\widehat{G}$.

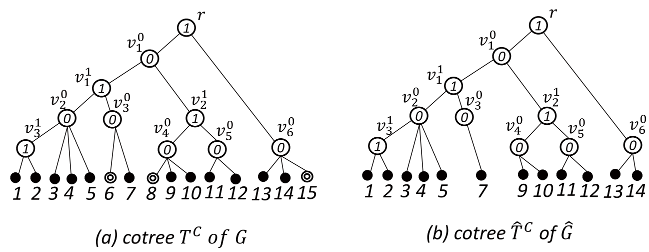


Fig. 2 Cotrees T^C of G and $\widehat{T}^C$ of $\widehat{G}$.

4. The Idea on Which the Proposed Algorithm Is Based

Our algorithm first finds a spanning tree $\widehat{\mathcal{T}}$ on $\widehat{G}$ induced by non-terminal vertices. Then, in this section, we assume that a cograph $\widehat{G}$ is induced by a non-terminal set V_{NT} and $\widehat{G}$ is connected. A case where $\widehat{G}$ is not connected will be considered in Sect. 6.

We first find a spanning tree $\widehat{\mathcal{T}}$ on $\widehat{G}$. As all leaves of $\widehat{\mathcal{T}}$, that are non-terminal vertices, must be finally connected to potential terminals for constructing an STNT of G, it is desirable that the number of leaves of $\widehat{\mathcal{T}}$ is as few as possible. (In the following, *connecting* a vertex v to a vertex u means that we connect v to u by using an edge $\{v, u\}$.) A spanning tree of $\widehat{G}$ with the least number of leaves is a Hamiltonian path on $\widehat{G}$. On a cograph G, a linear time algorithm for finding a Hamiltonian path is known [1]. This algorithm finds the minimum number of disjoint paths that cover all the vertices of G if G has no Hamiltonian path. Therefore, we first find a Hamiltonian path on $\widehat{G}$, if it exists, and regard it as a spanning tree of $\widehat{G}$. When no Hamiltonian path exists on $\widehat{G}$, we find disjoint paths $q_1, \cdots, q_j$ that cover all vertices of $\widehat{G}$. In this case, as will be described in the next section, a spanning tree $\widehat{\mathcal{T}}$ is constructed by connecting one terminal for each of q_i, $i = 2, \cdots, j$ to a vertex of q_1 so that it contains no cycle. Each leaf of $\widehat{\mathcal{T}}$ on $\widehat{G}$ needs to be connected to a distinct potential terminal on G, because it is a non-terminal vertex. Then, for constructing an STNT on G, we construct a spanning tree $\widehat{\mathcal{T}}$ so as to connect each leaf of $\widehat{\mathcal{T}}$ to a distinct potential terminal on G. Moreover, we show that if at least one of leaves of $\widehat{\mathcal{T}}$ cannot be connected to a distinct potential terminal on G, then G has no STNT.

4.1 A Method for Finding a Hamiltonian Path in a Cograph

Our algorithm for finding an STNT employs the algorithm for examining whether a Hamiltonian path exists or not in a cograph [1]. So, we explain this algorithm proposed by Corneil et al. [1].

We first define the technical terminology. For a finite graph $G = (V, E)$, we call $s(G) = \max(c(G-S) - |S| : S \subseteq V$ and $c(G-S) \neq 1)$ the *scattering number* of G where $c(G-S)$ denotes the number of connected components of $G-S$; a set $S \subseteq V$ is called a *scattering set* of G, if $c(G-S) - |S| = s(G)$, $c(G-S) \neq 1$ holds and no other set S' with $s(G)$ exists such that $S' \supset S$ which denotes that S is a proper subset of S'. We call the vertices in the scattering set S *scattering vertices*. The minimum number of disjoint paths that cover the all vertices of G is denoted by $\pi_0(G)$.

On cographs, the following theorem on relation among scattering number $s(G)$, Hamiltonian paths, Hamiltonian cycles and the minimum number $\pi_0(G)$ of disjoint paths is known [6].

Theorem 2 ([6]): Let $G = (V, E)$ be a cograph. Then
(1) G has a Hamiltonian path if and only if $s(G) \leq 1$,
(2) G has a Hamiltonian cycle if and only if $s(G) \leq 0$ and $|V| \geq 3$,
(3) $\pi_0(G) = \max(1, s(G))$. □

Based on this theorem, Corneil et al. [1] show that $s(G)$ can be calculated by applying the following bottom-up traversal for a cotree.

We assign $a_i = -1$, $b_i = 1$ to each leaf i of a cotree. As for internal vertices of a cotree, two values a_v, b_v are assigned to each vertex by the following procedures. We assume that an internal vertex v has k (> 0) children.

- For a label-1 vertex:

$$a_v = \max_{i=1}^{k}\left(a_i - \sum_{j\neq i} b_j\right) \quad (1), \qquad b_v = \sum_{i=1}^{k} b_i \quad (2).$$

- For a label-0 vertex:

$$a_v = \sum_{i=1}^{k} \max(a_i, 1) \quad (3), \qquad b_v = \sum_{i=1}^{k} b_i \quad (4).$$

For a cotree, $s(G)$ can be calculated by bottom-up traversal from leaves to the root. After calculating $s(G)$, the value a_i of each internal vertex i corresponds to the scattering number $s(G_i)$ of a subgraph G_i and the value of the root corresponds to $s(G)$ of a cograph G.

Figure 3 illustrates the calculation results of the scattering number $s(\widehat{G})$ of $\widehat{G}$ shown in Fig. 1 (b). The value a $(= 2)$ of the root corresponds to $s(\widehat{G})$. As $s(\widehat{G}) = 2$, no Hamiltonian path exists in $\widehat{G}$ and two disjoint paths (e.g., P_1 = 1 2 7 3 14 4 13 5, P_2 = 10 11 9 12) can cover all vertices of $\widehat{G}$ by Theorem 2. Moreover, a scattering set $\widehat{S}$ of $\widehat{G}$ is {13, 14}.

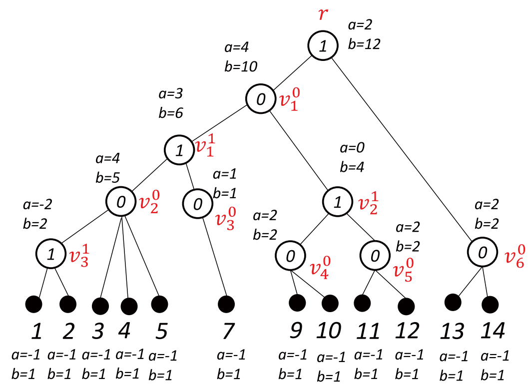


Fig. 3 Cotree of $\widehat{G}$.

Our algorithm first finds a Hamiltonian path or a disjoint path cover on $\widehat{G}$. In the following, we explain in detail how to construct a Hamiltonian path or a disjoint path cover on $\widehat{G}$.

We find a Hamiltonian path or a disjoint path cover by applying bottom-up traversal for $\widehat{T}^C$ from leaves to the root. On a subgraph corresponding to a subtree rooted at a label-1 vertex or a label-0 vertex of $\widehat{T}^C$, we explain how we construct a Hamiltonian path or a disjoint path cover. However, subgraphs corresponding to subtrees rooted at a label-0 vertices v^0 are a disjoint union of subgraphs $\widehat{G}_{v_1^1}, \cdots, \widehat{G}_{v_k^1}$ corresponding to subtrees rooted at children $v_1^1, \cdots, v_k^1$ of v^0, that is, each structure of $\widehat{G}_{v_1^1}, \cdots, \widehat{G}_{v_k^1}$ does not change after executing the disjoint union at v^0. Then, we describe how to construct a Hamiltonian path or a disjoint path cover only on subgraphs corresponding to subtrees rooted at label-1 vertices.

A subtree $\widehat{T}^C_{v_i^1}$ of $\widehat{T}^C$ rooted at a label-1 vertex v_i^1 corresponds to a subgraph $\widehat{G}_{v_i^1}$. On $\widehat{G}_{v_i^1}$, a Hamiltonian path, if it exists, or a disjoint path cover is found as follows. Let $v_1^0, v_2^0, \cdots, v_k^0$ be label-0 vertices that are children of v_i^1 and $\widehat{G}_{v_1^0}, \widehat{G}_{v_2^0}, \cdots, \widehat{G}_{v_k^0}$ be corresponding subgraphs, respectively. We assume that each Hamiltonian path or each disjoint path cover on $\widehat{G}_{v_1^0}, \cdots, \widehat{G}_{v_k^0}$ has been found. (See Fig. 4 (a).) As $\widehat{G}_{v_j^0}$, $j = 1, \cdots, k$, are subgraphs corresponding to children of label-1 v_i^1, each vertex of $\widehat{G}_{v_j^0}$ is adjacent to all vertices of each of the other subgraphs $\widehat{G}_{v_i^0}$, $i \neq j$. When constructing a Hamiltonian path or a disjoint path cover on $\widehat{G}_{v_i^1}$, a subgraph $\widehat{G}_{v_h^0}$ that has a_l disjoint paths $p_1, p_2, \cdots, p_{a_l}$ is selected among $\widehat{G}_{v_1^0}, \cdots, \widehat{G}_{v_k^0}$ by applying formula (1), that is, a subgraph satisfying formula (1) is $\widehat{G}_{v_h^0}$. (As illustrated in Fig. 4 (a), $\widehat{G}_{v_1^0}$ is selected by applying formula (1).) Vertices of the other subgraphs $\widehat{G}_{v_j^0}$, $j \neq h$ are scattering vertices. A Hamiltonian path or a disjoint path cover on $\widehat{G}_{v_i^1}$ is constructed by aligning paths $p_1, p_2, \cdots, p_{a_l}$ in $\widehat{G}_{v_h^0}$ and scatter-

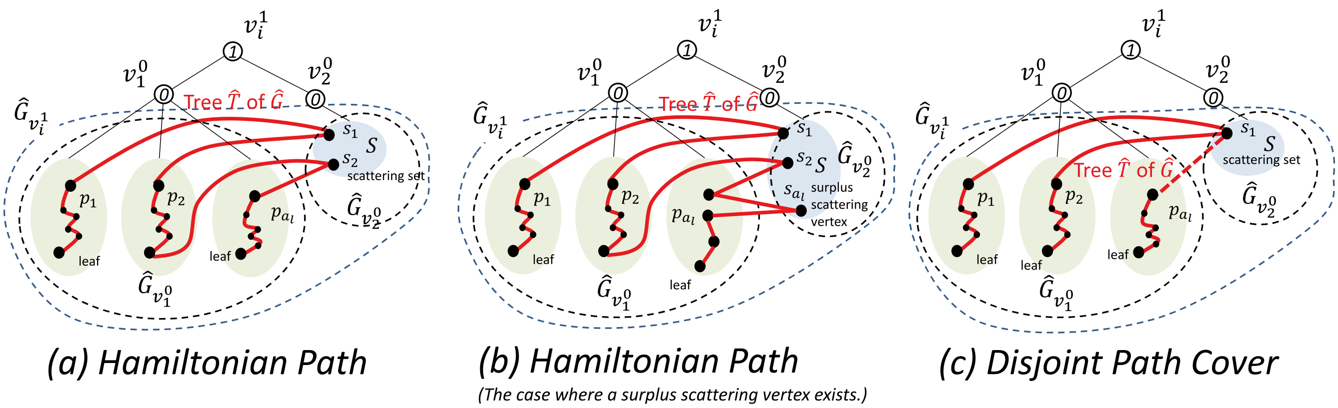


Fig. 4 Construction of a spanning tree.

ing vertices $s_1, s_2, \cdots, s_q$ in the other subgraphs $\widehat{G}_{v_j^0}$, $j \neq h$ alternately, i.e., align them in the following manner: p_1, s_1, p_2, s_2, $\cdots$. If the number of scattering vertices is $a_l - 1$, then we can construct a Hamiltonian path $H = p_1, s_1, p_2, s_2, \cdots, s_{a_{l-1}}, p_{a_l}$. When the number of scattering vertices is more than $a_l - 1$, a Hamiltonian path is constructed by aligning the vertices of path p_i and surplus scattering vertices s_{a_l}, $s_{a_{l+1}}, \cdots, s_q$ alternately, so that $l_1, s_{a_l}, l_2, s_{a_{l+1}}, \cdots, l_p$ is obtained where $l_1, \cdots, l_p$ are the vertices of p_i. (See Fig. 4 (b).) If the number a' of scattering vertices is less than $a_l - 1$, each vertex of $\widehat{G}_{v_i^1}$ are covered by exactly one of $a_l - a'$ disjoint paths. (See Fig. 4 (c).)

Our algorithm constructs a spanning tree on $\widehat{G}$ by using a Hamiltonian path or by connecting obtained disjoint paths. When $\widehat{G}$ has a Hamiltonian path H, H is a spanning tree on $\widehat{G}$. If $\widehat{G}$ has no Hamiltonian path, then we construct a spanning tree of $\widehat{G}$ by connecting obtained disjoint paths as follows: When we cannot construct a Hamiltonian path, then the number of scattering vertices is less than $a_l - 1$. Therefore, $\widehat{G}$ has a new path $P = p_1, s_1, p_2, s_2, \cdots, s_q$, p_{q+1} and let other paths be $p_{q+2}, \cdots, p_{a_l}$ on $\widehat{G}_{v_h^0}$. Each vertex of $\widehat{G}_{v_h^0}$ is adjacent to all vertices in the other subgraphs $\widehat{G}_{v_j^0}$, $j \neq h$, by the definition of the cograph. Then, as each vertex of $p_{q+2}, \cdots, p_{a_l}$ is adjacent to all scattering vertices in $\widehat{G}_{v_j^0}$, $j \neq h$, we can construct a spanning tree $\widehat{\mathcal{T}}$ by connecting either end(terminal) of each path of $p_{q+2}, \cdots, p_{a_l}$ to s_1. (See Fig. 4 (c).)

Example: We explain how to find the minimum disjoint path cover of $\widehat{G}$ as shown in Fig. 1 (b). By executing the procedure at v_1^1 of $\widehat{T}^C$ as shown in Fig. 2 (b), we find the minimum disjoint path cover of $\widehat{G}_{v_1^1}$. Since $\widehat{G}_{v_2^0}$ that includes 4-disjoint paths $p_1 = 1, 2$, $p_2 = 3$, $p_3 = 4$ and $p_4 = 5$ satisfies the formula (1), then a scattering vertex is 7. Paths $p_1 = 1, 2$ and $p_2 = 3$ are connected by the scattering vertex 7. Therefore, the minimum disjoint path cover of $\widehat{G}_{v_1^1}$ is $p'_1 = 1, 2, 7, 3$, $p'_2 = 4$ and $p'_3 = 5$.

When we execute the procedure at v_2^1 of $\widehat{T}^C$, scattering vertices are 11, 12, since $\widehat{G}_{v_4^0}$ that includes 2-disjoint paths $p_1 = 9$, $p_2 = 10$ satisfies the formula (1). Therefore, the minimum disjoint path cover 9, 11, 10, 12 of $\widehat{G}_{v_2^1}$ is constructed by aligning $p_1 = 9$, $p_2 = 10$ and scattering vertices 11, 12 alternately.

When we execute the procedure at root r of $\widehat{T}^C$, as $\widehat{G}_{v_1^0}$ that includes 4-disjoint paths $p_1 = 1, 2, 7, 3$, $p_2 = 4$, $p_3 = 5$ and $p_4 = 9, 11, 10, 12$ satisfies the formula (1), scattering vertices are 13, 14 in $\widehat{G}_{v_6^0}$. Therefore, two paths 9, 11, 10, 12 and 1, 2, 7, 3, 14, 4, 13, 5 constructed by aligning paths p_1, p_2, p_3 and scattering vertices 14, 13 alternately, are the minimum disjoint path cover.

5. A Method for Finding an STNT

By finding a Hamiltonian path or a disjoint path cover on $\widehat{G}$, we can tell which subgraph $\widehat{G}_j$ includes disjoint paths and which subgraph $\widehat{G}_k$ includes scattering vertices by applying equations (1)–(4). Finally, we can construct a spanning tree $\widehat{\mathcal{T}}$ on $\widehat{G}$ by connecting obtained disjoint paths $q_1, \cdots, q_k$ on a subgraph as described in Sect. 4.1.

However, construction of a disjoint path q_i for each subgraph $\widehat{G}_i$, has several possibilities depending on how to connect terminals of disjoint paths $p_1, p_2, \cdots, p_{a_l}$ on a subgraph $\widehat{G}_{v_h^0}$ of $\widehat{G}_i$ and scattering vertices $s_1, \cdots, s_q$. (See Fig. 4.) Each leaf of $\widehat{\mathcal{T}}$ on $\widehat{G}$ needs to be connected to a distinct potential terminal on G, because it is a non-terminal vertex. As a spanning tree $\widehat{\mathcal{T}}$ is constructed by connecting disjoint paths q_i's, terminals of a path q_i on $\widehat{G}_i$ constructed by connecting $p_1, \cdots, p_k$ on $\widehat{G}_{v_h^0}$ may finally become leaves of $\widehat{\mathcal{T}}$. (See Fig. 6.) Then, for constructing an STNT of G, we construct a disjoint path q_i on each subgraph G_i such that one terminal of q_i are connected to a distinct potential terminal. We next explain a method for constructing a disjoint path q_i on each subgraph G_i.

For constructing a spanning tree $\widehat{\mathcal{T}}$, we find a Hamiltonian path or a disjoint path cover by applying bottom-up traversal for $\widehat{T}^C$ from leaves to the root. Similarly, we choose one terminal for each of $q_1, \cdots, q_k$ and connect them to a distinct potential terminal by applying bottom-up traver-

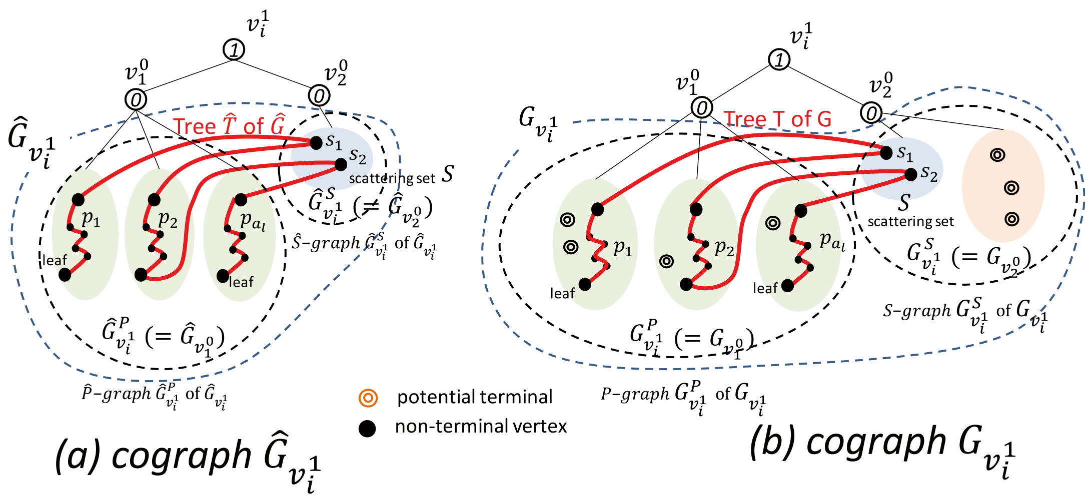


Fig. 5 $\widehat{P}$-graph, $\widehat{S}$-graph, P-graph and S-graph.

sal for cotree T^C. In the following, on a subgraph corresponding to a subtree rooted at a label-1 vertex or a label-0 vertex of T^C, we explain how to choose one terminal for each of $q_1, \cdots, q_k$ and connect them to a distinct potential terminal. However, subgraphs corresponding to subtrees rooted at a label-0 vertices v^0 are disjoint union of subgraphs $G_{v_1^1}, \cdots, G_{v_k^1}$ corresponding to subtrees rooted at children $v_1^1, \cdots, v_k^1$ of v^0, that is, each structure of $G_{v_1^1}, \cdots, G_{v_k^1}$ after executing the disjoint union at v^0 does not change. Then, we describe how to choose one terminal for each of $q_1, \cdots, q_k$ and connect them to a distinct potential terminal on only subgraphs corresponding to subtrees rooted at label-1 vertices.

There is one-to-one correspondence between the leaves of T^C and the vertices of G. We first describe the case where a label-1 vertex is adjacent to only leaves of T^C. As each pair of $v_1, \cdots, v_k$ is adjacent by the definition of cographs, when a label-1 vertex v^1 is adjacent to leaves $v_1, \cdots, v_k$, a subgraph G_{v^1} induced by $v_1, \cdots, v_k$ is a complete graph. (As illustrated in Fig. 3, v_3^1 corresponds to v^1 in this case.) Then, G_{v^1} has a Hamiltonian path P constructed only by non-terminal vertices. In this case, since exactly two terminals of P are leaves of the spanning tree, if two or more potential terminals exist in G_{v^1}, then an STNT on G_{v^1} can be constructed. If only one potential terminal exists in G_{v^1}, no STNT exists in G_{v^1}, but one terminal of P can be connected to a potential terminal in G_{v^1}.

We next describe the case where a label-1 vertex v_i^1 is adjacent to at least one label-0 vertex of T^C. (As illustrated in Fig. 3, v_1^1, v_2^1 and r correspond to v_i^1 in this case.)

As described before, we construct a Hamiltonian path, if it exists, or a disjoint path cover by applying bottom-up traversal for $\widehat{T}^C$. We assume that, on $\widehat{T}^C$, a label-1 vertex v_i^1 has children $v_1^0, v_2^0, \cdots, v_k^0$ that are label-0 vertices. On $\widehat{G}_{v_i^1}$ corresponding to a subtree $\widehat{T}^C_{v_i^1}$ rooted at v_i^1, we call $\widehat{G}_{v_1^0}, \widehat{G}_{v_2^0}, \cdots, \widehat{G}_{v_k^0}$ *children subgraphs of* $\widehat{G}_{v_i^1}$. (See Fig. 5 (a). In this

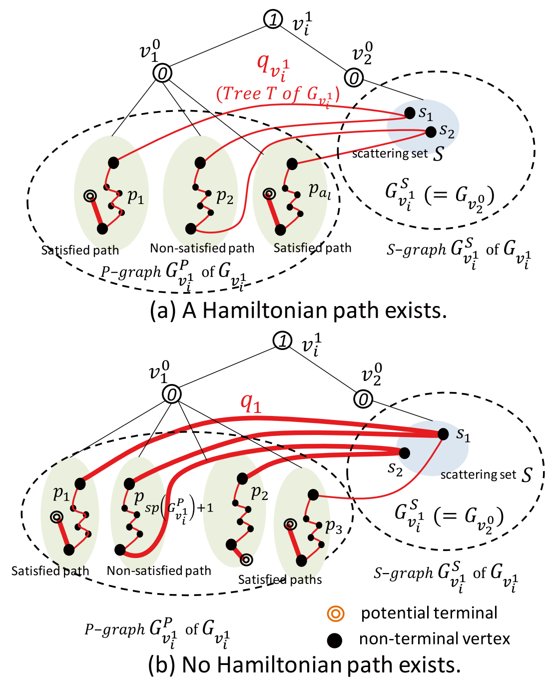


Fig. 6 Case 1.

case, $\widehat{G}_{v_1^0}$ and $\widehat{G}_{v_2^0}$ are children subgraphs of $\widehat{G}_{v_i^1}$.) On children subgraphs $\widehat{G}_{v_1^0}, \widehat{G}_{v_2^0}, \cdots, \widehat{G}_{v_k^0}$, a subgraph $\widehat{G}_{v_h^0}$ that has a_l paths is selected by applying formula (1), and a Hamiltonian path or a disjoint path cover on $\widehat{G}_{v_i^1}$ is constructed by connecting obtained disjoint paths $p_1, \cdots, p_{a_l}$ on $\widehat{G}_{v_h^0}$ and vertices of subgraphs $\widehat{G}_{v_j^0}$, $j \neq h$ in a manner described as follows: By the property of cographs, methods for connecting a terminal of a disjoint path to a potential terminal are classified into the following three cases. Note that each vertex of $\widehat{G}$ corresponds to a vertex of G.

Case 1: A terminal of a disjoint path included in $G_{v_h^0}$ is adjacent to a potential terminal included in $G_{v_h^0}$. (See Fig. 6. In this case, $G_{v_1^0}$ selected by applying formula (1) corresponds

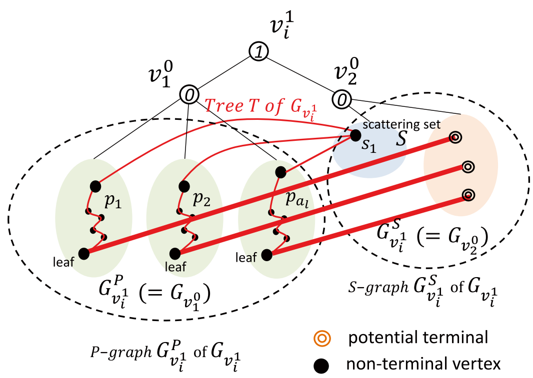


Fig. 7 Case 2.

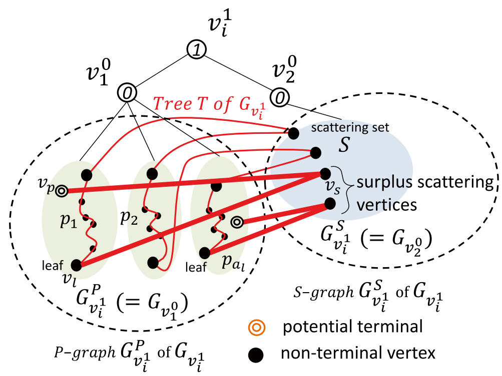


Fig. 8 Case 3.

to $G_{v_h^0}$.)
Case 2: Since each potential terminal in $G_{v_j^0}$, $j \neq h$, is adjacent to each vertex in $G_{v_h^0}$, we can connect a potential terminal in $G_{v_j^0}$, $j \neq h$ and a terminal of a disjoint path in $G_{v_h^0}$. (See Fig. 7.)
Case 3: Even if there is no potential terminal in $G_{v_h^0}$ adjacent to a terminal of a disjoint path, we can connect a potential terminal in $G_{v_h^0}$ to a terminal of a disjoint path indirectly via a scattering vertex in $G_{v_j^0}$, $j \neq h$. Note that, in this case, scattering vertices that have already been used to connect two disjoint paths in $G_{v_h^0}$ cannot be used to connect a potential terminal and a terminal of a disjoint path. (See Fig. 8.)

In order to explain in detail the above three cases, we define several technical terminologies related to $\widehat{G}$. By finding a Hamiltonian path or a disjoint path cover on $\widehat{G}$, we can tell which subgraph $\widehat{G}_i$ includes disjoint paths and which subgraph $\widehat{G}_j$ includes scattering vertices. Since disjoint paths on $\widehat{G}_{v_h^0}$ are constructed based on constructed paths on $\widehat{G}_{v_i^1}$, we call $\widehat{G}_{v_h^0}$ a $\widehat{P}$*-graph of* $\widehat{G}_{v_i^1}$ and denote it by $\widehat{G}^P_{v_i^1}$. (See Fig. 5 (a). In this case, $\widehat{G}_{v_1^0}$ is a $\widehat{P}$-graph $\widehat{G}^P_{v_i^1}$.) Moreover, a graph obtained by removing vertices and edges of $\widehat{G}^P_{v_i^1}$ from $\widehat{G}_{v_i^1}$, is induced only by scattering vertices. Then, we call such a graph an $\widehat{S}$*-graph of* $\widehat{G}_{v_i^1}$ and denote it by $\widehat{G}^S_{v_i^1}$. (See Fig. 5 (a). In this case, $\widehat{G}_{v_2^0}$ is an $\widehat{S}$-graph $\widehat{G}^S_{v_i^1}$.)

We next define, on a subgraph $G_{v_i^1}$ of an input graph G, a P-graph $G^P_{v_i^1}$ corresponding to a $\widehat{P}$-graph $\widehat{G}^P_{v_i^1}$ and an S-graph $G^S_{v_i^1}$ corresponding to an $\widehat{S}$-graph $\widehat{G}^S_{v_i^1}$ as follows. Note that each vertex of $\widehat{G}$ corresponds to a vertex of G.

Let v_k^0 be the root of $\widehat{T}^C_{v_k^0}$ corresponding to $\widehat{G}^P_{v_i^1}$. Vertex v_k^0 of cotree $\widehat{T}^C$ of $\widehat{G}$ corresponds to vertex v_k^0 of cotree T^C of G. On T^C, a subgraph $G_{v_k^0}$ corresponding to a subtree $T^C_{v_k^0}$ rooted at v_k^0 is a P*-graph* $G^P_{v_i^1}$ *of* $G_{v_i^1}$. (See Figs. 5 (a) (b). In this case, v_1^0 is the root of $T^C_{v_1^0}$ corresponding to $G^P_{v_i^1}$.) Moreover, a graph obtained by removing vertices and edges of $G^P_{v_i^1}$ from $G_{v_i^1}$, is an S*-graph* $G^S_{v_i^1}$ *of* $G_{v_i^1}$. (See Fig. 5 (b).) Whether potential terminals can be connected to distinct terminals of $p_1, \cdots, p_{a_l}$ or not depends on whether potential terminals are included in which of P-graph and S-graph. On $G_{v_i^1}$, each vertex of S-graph $G^S_{v_i^1}$ is adjacent to all vertices of P-graph $G^P_{v_i^1}$ by the definition of the cograph. Furthermore, each terminal of $p_1, \cdots, p_{a_l}$ on $G_{v_i^1}$ is included in P-graph $G^P_{v_i^1}$. Then, the above-mentioned three cases where a terminal of $p_1, \cdots, p_{a_l}$ is connected to a potential terminal is described formally as follows. Note that each vertex of $\widehat{G}$ corresponds to a vertex of G, then disjoint paths $p_1, \cdots, p_{a_l}$ on $\widehat{G}^P_{v_i^1}$ that have been found, exist on $G^P_{v_i^1}$.
Case 1: The case where, on paths $p_1, \cdots, p_{a_l}$ on $G^P_{v_i^1}$, either of terminals of paths $p_1, \cdots, p_{a_l}$ has been connected to a distinct potential terminal in $G^P_{v_i^1}$. (See Figs. 6 (a) (b).)

After finding an STNT on a child subgraph of $G^P_{v_i^1}$, either of terminals of paths on $G^P_{v_i^1}$ has been connected to a potential terminal in $G^P_{v_i^1}$.
Case 2: The case where an S-graph $G^S_{v_i^1}$ has potential terminals.

Since each vertex in $G^S_{v_i^1}$ is adjacent to all vertices in $G^P_{v_i^1}$, distinct vertices in $G^S_{v_i^1}$ can be connected to terminals of $p_1, \cdots, p_{a_l}$ not connected to a potential terminal in $G^P_{v_i^1}$. (See Fig. 7.)
Case 3: The case where potential terminals in $G^P_{v_i^1}$ that are not connected to a terminal of $p_1, \cdots, p_{a_l}$, exist.

Such vertex v_p is not adjacent to a terminal of $p_1, \cdots, p_{a_l}$, but it is adjacent to all non-terminal vertices in $G^S_{v_i^1}$. Let v_l in $G^P_{v_i^1}$ be a terminal not connected to a potential terminal and v_s in $G^S_{v_i^1}$ be a scattering vertex. v_p, v_s and v_l are connected indirectly via v_s. (See Fig. 8.)

We describe a method of determining whether an STNT exists or not on a subgraph $G_{v_i^1}$ with regard to the above-mentioned three cases. We construct a Hamiltonian path or a disjoint path cover on subgraphs by traversing cotree T^C in bottom-up order from leaves to the root. On each subgraph $G_{v_i^1}$, we first construct a Hamiltonian path or a disjoint path cover $q_1, \cdots, q_k$ by connecting disjoint paths

$p_1, \cdots, p_l$ on a child subgraph $G^P_{v^1_i}$ of $G_{v^1_i}$ and scattering vertices on a child subgraph $G^S_{v^1_i}$ of $G_{v^1_i}$. If the number of constructed disjoint paths on $G_{v^1_i}$ is one, that is, q_1 is a Hamiltonian path, then q_1 is a tree. (See Fig. 6 (a). In this case, $q_1 = p_1, s_1, p_2, s_2, p_{a_l}$ is constructed by aligning three disjoint paths p_1, p_2, p_{a_l} on $G^P_{v^1_i}$ and scattering vertices s_1, s_2 on $G^S_{v^1_i}$, alternately.) Otherwise, the set of $q_1, \cdots, q_k$ ($k > 1$) is a disjoint path cover. Before explaining the case where the number k of constructed disjoint paths on $G_{v^1_i}$ is more than one, we define several technical terminologies. As described later, we construct an STNT by connecting disjoint paths p_i's where either terminal of p_i has been connected to a distinct potential terminal. Therefore, we need to obtain the number of such disjoint paths p_i's. Thus we have obtained the number of disjoint paths p_i's constructed by non-terminal vertices where either terminal of p_i has been connected to a distinct potential terminal in $G^P_{v^1_i}$, because we have constructed disjoint paths connected to a distinct potential terminal in a child subgraph $G^P_{v^1_i}$ of $G_{v^1_i}$ by traversing cotree T^C in bottom-up order from leaves to the root. We call such a disjoint path p_i where either terminal of p_i has been connected to a distinct potential terminal a *satisfied path*. On the other hand, a *non-satisfied path* is a disjoint path p_i where neither terminal of p_i has been connected to a distinct potential terminal. In the case where q_1 is a Hamiltonian path, since q_1 is constructed by $p_1, \cdots, p_l$ and scattering vertices, if at least two satisfied paths exists among $p_1, \cdots, p_l$, then an STNT exists on $G_{v^1_i}$. (See Fig. 6 (a). p_1 and p_{a_l} are two satisfied paths. Then, an STNT exists on $G_{v^1_i}$.)

We show that when a Hamiltonian path does not exist on $G_{v^1_i}$, that is, $q_1, \cdots, q_k$, $k > 1$, holds, if the number of satisfied paths among $p_1, \cdots, p_l$ is k, then we can construct an STNT on $G_{v^1_i}$ as follows: On each subgraph $G_{v^1_i}$, we use the number of satisfied paths on its child subgraph $G^P_{v^1_i}$ and the number of constructed disjoint paths on $G_{v^1_i}$. Let $sp(G^P_{v^1_i})$ be the number of satisfied paths and $dp(G_{v^1_i})$ be the number of disjoint paths constructed by non-terminal vertex in $G_{v^1_i}$. (For example, $dp(G_{v^1_i}) = 1$, $sp(G^P_{v^1_i}) = 2$ hold in Fig. 6 (a) and $dp(G_{v^1_i}) = 2$, $sp(G^P_{v^1_i}) = 3$ hold in Fig. 6 (b).) The $dp(G_{v^1_i})$ disjoint paths $q_1, \cdots, q_{dp(G_{v^1_i})}$ are constructed by connecting $p_1, \cdots, p_{a_l}$ on $G^P_{v^1_i}$ and scattering vertices in $G^S_{v^1_i}$. We now assume that $p_1, \cdots, p_{sp(G^P_{v^1_i})}$ are satisfied paths and $p_{sp(G^P_{v^1_i})+1}, \cdots, p_{a_l}$ are non-satisfied paths. We first construct a disjoint path $q_1 = p_1, s_1, p_{sp(G^P_{v^1_i})+1}, s_2, \cdots, p_{a_l}, s_{a_l}, p_2$ by selecting two satisfied paths p_1, p_2 among $p_1, \cdots, p_{sp(G^P_{v^1_i})}$, as the starting path and the ending path, and by aligning scattering vertices $s_1, \cdots, s_{a_l}$ and non-satisfied paths $p_{sp(G^P_{v^1_i})+1}, \cdots, p_{a_l}$, alternately. (As illustrated in Fig. 6 (b), $q_1 = p_1, s_1, p_{sp(G^P_{v^1_i})+1}, s_2, p_2$.) A spanning tree $\mathcal{T}_{v^1_i}$ of $G_{v^1_i}$ can be constructed by connecting a terminal of surplus satisfied paths $p_3, \cdots, p_{sp(G^P_{v^1_i})}$ to a scattering vertex s_1 of q_1 in $G^S_{v^1_i}$. (As illustrated in Fig. 6 (b), a spanning tree $\mathcal{T}_{v^1_i}$ can be constructed by connecting a terminal of surplus satisfied path p_3 to a scattering vertex s_1 of q_1.) Two terminals of q_1 and one terminal for each of $p_3, \cdots, p_{sp(G^P_{v^1_i})}$ are leaves of $\mathcal{T}_{v^1_i}$. Therefore, since two terminals of q_1 and one terminal for each of $p_3, \cdots, p_{sp(G^P_{v^1_i})}$ are connected to a distinct potential terminal, then the constructed spanning tree $\mathcal{T}_{v^1_i}$ is an STNT on $G_{v^1_i}$.

In Case 1, if $(dp(G_{v^1_i}) + 1) - sp(G^P_{v^1_i}) \leq 0$ holds, then one terminal for each of disjoint paths can be connected to a distinct potential terminal, that is, we can construct an STNT on $G_{v^1_i}$. (As illustrated in Fig. 6 (a), $dp(G_{v^1_i}) = 1$ and $sp(G^P_{v^1_i}) = 2$, then $(dp(G_{v^1_i}) + 1) - sp(G^P_{v^1_i}) \leq 0$ holds. In Fig. 6 (b), $dp(G_{v^1_i}) = 2$ and $sp(G^P_{v^1_i}) = 3$, then $(dp(G_{v^1_i}) + 1) - sp(G^P_{v^1_i}) \leq 0$ holds.)

In Case 2, when l non-satisfied paths in $G^P_{v^1_i}$ exist after executing the process in Case 1, if l or more potential terminals in $G^S_{v^1_i}$ exist, we can connect such terminals of disjoint paths to distinct potential terminals. Then, we can construct an STNT on $G_{v^1_i}$. (As illustrated in Fig. 7, three non-satisfied paths in $G^P_{v^1_i}$ exist. However, there are three potential terminals in $G^S_{v^1_i}$. Then an STNT on $G_{v^1_i}$ can be constructed by connecting one terminal of each of three non-satisfied paths in $G^P_{v^1_i}$ to vertex s_1, which is only one vertex in the scattering set, and also connecting the other terminals to three potential terminals in $G^S_{v^1_i}$.)

When non-satisfied paths exist after executing processes in Cases 1 and 2, and if potential terminals not yet connected to a terminal of disjoint paths exist in $G^P_{v^1_i}$ as in Case 3, each of such vertices can be connected to a terminal by using non-terminal vertices in $G^S_{v^1_i}$. However, we need to use non-terminal vertices in $G^S_{v^1_i}$ for connecting disjoint paths on $G^P_{v^1_i}$. If the number $dp(G_{v^1_i})$ of disjoint paths constructed by non-terminal vertices on $G_{v^1_i}$ is a_l, we need $a_l - 1$ scattering vertices to construct disjoint paths. Let $nt(G^S_{v^1_i})$ be the number of non-terminal vertices in $G^S_{v^1_i}$. When $nt(G^S_{v^1_i}) - (a_l - 1) > 0$ holds, $nt(G^S_{v^1_i}) - (a_l - 1)$ scattering vertices are not used to construct disjoint paths. Such $nt(G^S_{v^1_i}) - (a_l - 1)$ scattering vertices are called *surplus scattering vertices* and are denoted by $r(G^S_{v^1_i})$. (Note: The role of surplus scattering vertices in this paper is different from that in [1] as follows. As described in Sect. 4.1, the algorithm [1] by Corneil et al. constructs a disjoint path cover by connecting disjoint paths and scattering vertices. If the number of disjoint paths is q, we need $q - 1$ scattering vertices for constructing a new disjoint path. If surplus scattering verities exist, then these are included in inner vertices of disjoint paths. On the other hand, our algorithm uses surplus scattering vertices for connecting disjoint paths and non-terminal vertices.) In Case 3, we can use surplus scattering vertices to connect a terminal of disjoint paths and a potential terminal in $G^P_{v^1_i}$. (As illustrated in Fig. 8, a new path P is constructed by aligning three paths p_1, p_2, p_{a_l} and two scattering vertices, respectively. Two terminals of p_1, p_2 and p_{a_l} are non-satisfied paths, but surplus scattering vertices $r(G^S_{v^1_i}) = 2$ and two potential terminals exist in $G^P_{v^1_i}$. Then an STNT on $G_{v^1_i}$ can be constructed by connecting each terminal of P to a distinct potential vertex indirectly via a surplus scattering vertex.)

To summarize the above, we examine whether distinct potential terminals can be connected to each terminal of non-satisfied paths by the following processing. Let $po(G^S_{v^1_i})$ be the number of potential terminals in $G^S_{v^1_i}$ and $po^-(G^P_{v^1_i})$ be the number of potential terminals in $G^P_{v^1_i}$ not connected to a terminal of disjoint paths.

(1) If $S1(G_{v^1_i}) = (dp(G_{v^1_i}) + 1) - sp(G^P_{v^1_i}) \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal.

(2) The case where $S1(G_{v^1_i}) > 0$ holds.

$S1(G_{v^1_i})$ terminals of disjoint paths not connected to distinct potential terminals exist. If $S2(G_{v^1_i}) = S1(G_{v^1_i}) - po(G^S_{v^1_i}) \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal in $G^S_{v^1_i}$.

(3) The case where $S2(G_{v^1_i}) > 0$ holds.

$S2(G_{v^1_i})$ terminals of disjoint paths not connected to distinct potential terminals exist. If $S3(G_{v^1_i}) = S2(G_{v^1_i}) - \min\{po^-(G^P_{v^1_i}), r(G^S_{v^1_i})\} \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal indirectly via a surplus scattering vertex.

Lemma 1: On a subgraph $G_{v^1_i}$ corresponding to subtree $T^C_{v^1_i}$ rooted at a label-1 vertex v^1_i, if at least one of $S1(G_{v^1_i})$, $S2(G_{v^1_i})$ and $S3(G_{v^1_i})$ is equal to or less than 0, then an STNT exists on $G_{v^1_i}$, otherwise it does not.

Proof. We find an STNT on a subgraph $G_{v^1_i}$ by applying bottom-up traversal for a cotree $T^C_{v^1_i}$ from leaves to root v^1_i. Let $v^0_1, \cdots, v^0_k$ be label-0 vertices that are children of v^1_i. On each of subgraphs $G_{v^0_1}, \cdots, G_{v^0_k}$ corresponding to $v^0_1, \cdots, v^0_k$, a Hamiltonian path or a disjoint path cover has been found and $sp(G^P_{v^1_i})$, $nt(G^S_{v^1_i})$, $dp(G^P_{v^1_i})$, $po(G^S_{v^1_i})$ and $po^-(G^P_{v^1_i})$ have been calculated. A subgraph $G_{v^0_h}$ that has a_l disjoint paths p_1, p_2, $\cdots$, p_{a_l} is selected among $G_{v^0_1}, \cdots, G_{v^0_k}$ by applying formula (1). Vertices of the other subgraphs $G_{v^0_j}$, $j \neq h$ are scattering vertices $s_1, s_2, \cdots, s_q$. Each vertex in $G_{v^0_j}$, $j \neq h$, is adjacent to all vertices in $G_{v^0_h}$ and potential terminals belong to either P-graph $G^P_{v^1_i}$ or S-graph $G^S_{v^1_i}$.

If $S1(G_{v^1_i}) = (dp(G_{v^1_i})+1) - sp(G^P_{v^1_i}) \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal.

When $S1(G_{v^1_i}) > 0$ holds, $S1(G_{v^1_i})$ terminals of disjoint paths not connected to distinct potential terminals exist. If $S2(G_{v^1_i}) = S1(G_{v^1_i}) - po(G^S_{v^1_i}) \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal in $G^S_{v^1_i}$.

When $S2(G_{v^1_i}) > 0$ holds, $S2(G_{v^1_i})$ terminals of disjoint paths not connected to distinct potential terminals exist. If $S3(G_{v^1_i}) = S2(G_{v^1_i}) - \min\{po^-(G^P_{v^1_i}), r(G^S_{v^1_i})\} \leq 0$ holds, then we can construct an STNT on $G_{v^1_i}$, since one terminal for each of non-satisfied paths is connected to a distinct potential terminal indirectly via a surplus scattering vertex.

We next show that if $S1(G_{v^1_i})$, $S2(G_{v^1_i})$ and $S3(G_{v^1_i})$ are greater than 0, then no STNT exists on $G_{v^1_i}$. We consider the case where $S1(G_{v^1_i}) > 0$ holds, that is, a terminal of disjoint paths not connected to a potential terminal exists. A spanning tree $\widehat{\mathcal{T}}_{v^1_i}$ with non-terminal vertices is constructed by connecting obtained disjoint paths $p_1, \cdots, p_{a_l}$ on $G^P_{v^1_i}$. We assume that P $(= p_1, s_1, p_2, s_2 \cdots, s_{a_l}, p_{a_l})$ is a path not connected to a potential terminal. (As illustrated in Fig. 9, P $(= p_1, s, p_{a_l})$.) In this case, we show that regardless of the construction of P, we cannot connect a potential terminal to a terminal of P.

Let $T^C_{v^1_P}$ be a cotree rooted at v^1_P corresponding to a P-graph $G^P_{v^1_i}$ of $G_{v^1_i}$. There is one-to-one correspondence between leaves of $T^C_{v^1_P}$ and vertices of G. On $T^C_{v^1_P}$, if there is a label-1 vertex adjacent to leaves $v_1, \cdots, v_k$ and at least one leaf among $v_1, \cdots, v_k$ is a potential terminal, then a terminal of P can be connected to a potential terminal, contradicting that P is a path not connected to a potential terminal.

Therefore, each leaf of $T^C_{v^1_P}$ adjacent to label-1 vertices is a non-terminal vertex. However, the leaves of $T^C_{v^1_P}$ adjacent to label-0 vertices may be potential terminals. (See Fig. 9. In

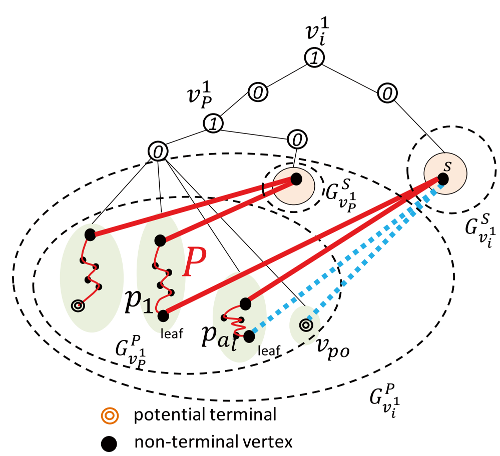


Fig. 9 Connection of a potential terminal and a terminal of a disjoint path using a scattering vertex.

this case, such potential terminal is v_{po}.) Since $S2(G_{v_i^1}) > 0$ holds, no potential terminal adjacent to a terminal of P exists in $G^S_{v_i^1}$. Then, potential terminals belong to P-graph $G^P_{v_P^1}$. To make a potential terminal v_{po} and a terminal of P connected, we need to use a scattering vertex s in $G^S_{v_i^1}$, i.e., as well as by connecting v_{po} and P with s by edges. However, since $S3(G_{v_i^1}) > 0$ holds, no surplus scattering vertex exists in $G^S_{v_i^1}$. (See Fig. 9. Scattering vertex s is used to construct to p_1 and p_{a_l}.) If we use scattering vertex s in $G^S_{v_i^1}$ to connect a potential terminal in $G^P_{v_P^1}$ to a terminal of P, then P is divided into two paths. The number of leaves of a spanning tree increases by increasing the number of paths. Therefore, on $G_{v_i^1}$, we cannot connect a potential terminal to a terminal of P no matter how we construct a P.

Consequently, when $S1(G_{v_i^1})$, $S2(G_{v_i^1})$ and $S3(G_{v_i^1})$ are greater than 0, no STNT exists on $G_{v_i^1}$. □

On each subgraph $G_{v_i^1}$, we find $S1(G_{v_i^1})$, $S2(G_{v_i^1})$ and $S3(G_{v_i^1})$ by applying bottom-up traversal for a cotree T^C from leaves to the root. If at least one of $S1(G_{v_i^1})$, $S2(G_{v_i^1})$ and $S3(G_{v_i^1})$ is equal to or less than 0, then $G_{v_i^1}$ has an STNT. This means that each terminal of disjoint paths connects to a distinct terminal on $G_{v_i^1}$. Otherwise, $S3(G_{v_i^1})$ terminals of disjoint paths do not connect to distinct terminals on $G_{v_i^1}$. Finally, if at least one of $S1(G)$, $S2(G)$ and $S3(G)$ is equal to or less than 0, then G has an STNT, otherwise it does not have an STNT. In summary, our algorithm is as follows.

Procedure Find_Spanning_Tree_with_Non-terminal_Set

Step 1. Find two cotrees T^C of G and $\widehat{T}^C$ of $\widehat{G}$.

Step 2. By traversing cotree $\widehat{T}^C$ in bottom-up order from leaves to the root, we execute the following instructions to each subgraph $\widehat{G}_{v_i^1}$ of $\widehat{G}$, corresponding to inner vertices v_i^1 of $\widehat{T}^C$.
begin
Find the scattering number $s(\widehat{G}_{v_i^1})$, a $\widehat{P}$-graph $\widehat{G}^P_{v_i^1}$ and an $\widehat{S}$-graph $\widehat{G}^S_{v_i^1}$.
end

Step 3. By traversing cotree T^C in bottom-up order from leaves to the root, we execute the following instructions to each subgraph $G_{v_i^1}$ of G corresponding to inner vertices v_i^1 of T^C.
begin
Find $nt(G^S_{v_i^1})$, $po(G^S_{v_i^1})$ and $po^-(G^P_{v_i^1})$.
end

Step 4. By traversing cotree T^C in bottom-up order from leaves to the root, we execute the following instructions to each subgraph $G_{v_i^1}$ of G corresponding to inner vertices v_i^1 of T^C.
begin
Find $S1(G_{v_i^1})$, $S2(G_{v_i^1})$ and $S3(G_{v_i^1})$.
end

Step 5. If at least one of $S1(G)$, $S2(G)$ and $S3(G)$ is equal to or less than 0, then G has an STNT, otherwise it does not have an STNT. □

Theorem 3: Procedure Find_Spanning_Tree_with_Non-terminal_Set finds an STNT of G in $O(n+m)$ time.

Proof. We first describe the correctness of the procedure. We find each STNT on $G_{v_i^1}$ by traversing a cotree T^C in bottom-up order from leaves to the root. When reaching the root of T^C, we obtain $S1(G)$, $S2(G)$ and $S3(G)$. By Lemma 1, if at least one of $S1(G)$, $S2(G)$ and $S3(G)$ is equal to or less than 0, then G has an STNT, otherwise it does not.

We next analyze the complexity of the procedure. In step 1, a cotree of a cograph can be constructed in $O(n+m)$ time [4], [5]. In step 2, the algorithm for finding a Hamiltonian path, if it exists, or disjoint paths on cograph can be done in $O(n+m)$ time [1]. In step 3, since the number of the disjoint paths and that of scattering vertices on each subgraph of $\widehat{G}$ have been found in step 2 [1], $nt(G^S_{v_i^1})$, $po(G^S_{v_i^1})$ and $po^-(G^P_{v_i^1})$ are calculated in $O(1)$ time on each internal vertex v_i^1 of T^C. Then, step 3 can be done in $O(n+m)$ time by applying bottom-up traversal for a cotree T^C. Step 4 can be done in $O(n+m)$ time by applying bottom-up traversal for a cotree T^C. Therefore, Procedure Find_Spanning_Tree_with_Non-terminal_Set finds an STNT in $O(n+m)$ time. □

6. The Case Where $\widehat{G}$ Is Not Connected

We finally consider the case where $\widehat{G}$ is not connected. Since an input graph G is connected, the root of cotree T^C is a label-1 vertex. Let $v_1^0, \cdots, v_k^0$ be label-0 vertices that are children of the root and $G_{v_1^0}, \cdots, G_{v_k^0}$ be corresponding subgraphs on G, respectively. As each vertex of $G_{v_i^0}$ is adjacent to all vertices of each of the other subgraphs $G_{v_j^0}$, $j \neq i$, if two or more subgraphs among $G_{v_1^0}, \cdots, G_{v_k^0}$ have non-terminal vertices, then $\widehat{G}$ is connected. Therefore, when $\widehat{G}$ is not connected, all non-terminal vertices are included in a subgraph among $G_{v_1^0}, \cdots, G_{v_k^0}$ on G. We assume that $G_{v_h^0}$ includes all non-terminal vertices and the other subgraphs include only potential terminals. (See Fig. 10.) As $\widehat{G}$ is not connected, all non-terminal vertices are included in subgraphs $G_{v_1^1}, \cdots,$

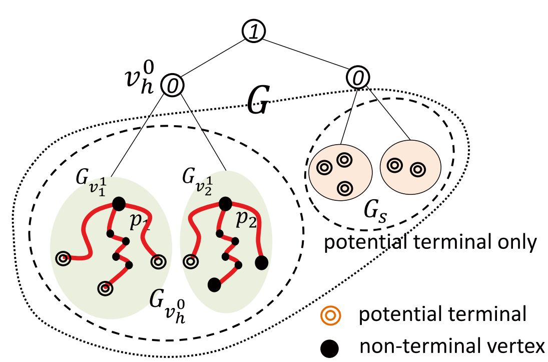


Fig. 10 The case where $\widehat{G}$ is not connected.

$G_{v_l^1}$ corresponding to label-1 vertices $v_1^1, \cdots, v_l^1$ whose parent is v_h^0. Subgraphs obtained by removing $G_{v_h^0}$ from G are induced only by potential terminals. Then, we apply Procedure Find_Spanning_Tree_with_Non-terminal_Set to each of $G_{v_1^1}, \cdots, G_{v_l^1}$. For each of $G_{v_1^1}, \cdots, G_{v_l^1}$, we find the number of terminals of disjoint paths that are not connected to a potential terminal. Vertices of the subgraph G_s obtained by removing $G_{v_h^0}$ from G are only potential terminals and each of these vertices is adjacent to all vertices in $G_{v_h^0}$. (See Fig. 10.) Then, each distinct potential terminal can be connected to a leaf in $G_{v_h^0}$. If the number of potential terminals in G_s is equal to or more than that of terminals in $G_{v_h^0}$ not connected to a potential terminal, then G has an STNT, otherwise it does not.

7. Concluding Remarks

In this paper, we first have shown that the problem for finding an STNT is NP-hard even if each edge has the weight of one on general graphs. We also have shown that if G is a cograph then finding an STNT of G is linearly solvable when each edge has the weight of one. We are interested in finding other classes of graphs in which this problem is polynomially solvable.

Acknowledgments

We thank anonymous referees whose comments were greatly beneficial to improving this paper. This work was partially supported by the Grant-in-Aid (C26330359) from Japan Society For The Promotion Of Science.

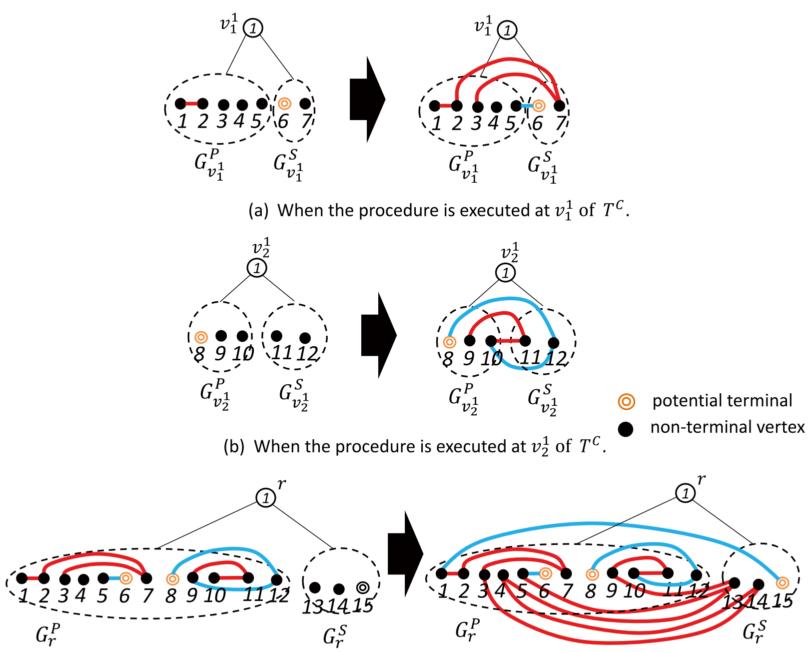
References

[1] D.G. Corneil, H. Lerchs, and L.S. Burlingham, “Complement reducible graphs,” Discrete Applied Mathematics, vol.3, no.3, pp.163–174, 1981.
[2] D.G. Corneil and Y. Perl, “Clustering and domination in perfect graphs,” Discrete Applied Mathematics, vol.9, no.1, pp.27–39, 1984.
[3] D.G. Corneil, Y. Perl, and L.K. Stewart, “Cographs: Recognition, applications and algorithms,” Proc. Fifteenth Southeastern Conference on Combinatorics, Graph Theory and Computing, pp.249–258, 1984.
[4] D.G. Corneil, Y. Perl, and L. Stewart, “A linear recognition algorithm for cographs,” SIAM J. Comput., vol.14, no.4, pp.926–934, 1985.
[5] M. Habib and C. Paul, “A simple linear time algorithm for cograph recognition,” Discrete Applied Mathematics, vol.145, no.2, pp.183–197, 2005.
[6] H.A. Jung, “On a class of posets and the corresponding comparability graphs,” Journal of Combinatorial Theory, Series B, vol.24, no.2, pp.125–133, 1978.
[7] D. Seinsche, “On a property of the class of n-colorable graphs,” Journal of Combinatorial Theory, Series B, vol.16, no.2, pp.191–193, 1974.
[8] T. Zhang and Y. Yin, “The minimum spanning tree problem with non-terminal set,” Information Processing Letters vol.112, no.17-18, pp.688–690, 2012.

Appendix

We explain in detail how to find an STNT of G by applying Procedure Find_Spanning_Tree_with_Non-terminal_Set to the graph as shown in Fig. 1 (a).

When the procedure is executed at v_1^1 of T^C as shown in Fig. 2 (a), a P-graph $G^P_{v_1^1}$ of $G_{v_1^1}$ is $G_{v_2^0}$ that includes 4-disjoint paths $p_1 = 1, 2$, $p_2 = 3$, $p_3 = 4$ and $p_4 = 5$,



(a) When the procedure is executed at v_1^1 of T^C.

(b) When the procedure is executed at v_2^1 of T^C.

(c) When the procedure is executed at root r of T^C.

Fig. A· 1 An example.

by applying formula (1). (See Fig. A·1 (a).) An S-graph $G^S_{v^1_1}$ is $G_{v^0_3}$ that includes a scattering vertex 7. A new path $1, 2, 7, 3$ is constructed and this subgraph $G_{v^1_1}$ is applicable to Case 2. Then, a potential terminal 6 is connected to a non-terminal vertex 5 that is one terminal of p_4. $S1(G_{v^1_1}) = (dp(G_{v^1_1}) + 1) - sp(G^P_{v^1_1}) = (3 + 1) - 0 = 4 \geq 0$, $S2(G_{v^1_1}) = S1(G_{v^1_1}) - po(G^S_{v^1_1}) = 4 - 1 = 3 \geq 0$ and $S3(G_{v^1_1}) = S2(G_{v^1_1}) - \min\{po^-(G^P_{v^1_1}), r(G^S_{v^1_1})\} = 3 - \min\{0, 0\} = 3 \geq 0$. Therefore, $G_{v^1_1}$ has no STNT, since $S1(G_{v^1_1})$, $S2(G_{v^1_1})$ and $S3(G_{v^1_1})$ are greater than 0.

When the procedure is executed at v^1_2 of T^C, a P-graph $G^P_{v^1_2}$ is $G_{v^0_4}$ that includes 2-disjoint paths $p_1 = 9$ and $p_2 = 10$, by applying formula (1). (See Fig. A·1 (b).) An S-graph $G^S_{v^1_2}$ is $G_{v^0_5}$ that includes two scattering vertices 11 and 12. A new path $9, 11, 10$ is constructed and this subgraph $G_{v^1_2}$ is applicable to Case 3 such that a non-terminal vertex 12 is a surplus scattering vertex. Then, a potential terminal 8 is connected to a non-terminal vertex 10 by using the surplus scattering vertex 12. $S1(G_{v^1_2}) = (dp(G_{v^1_2}) + 1) - sp(G^P_{v^1_2}) = (1 + 1) - 0 = 2 \geq 0$, $S2(G_{v^1_2}) = S1(G_{v^1_2}) - po(G^S_{v^1_2}) = 2 - 0 = 2 \geq 0$ and $S3(G_{v^1_2}) = S2(G_{v^1_2}) - \min\{po^-(G^P_{v^1_2}), r(G^S_{v^1_2})\} = 2 - \min\{1, 1\} = 1 \geq 0$. Therefore, $G_{v^1_2}$ has no STNT, since $S1(G_{v^1_2})$, $S2(G_{v^1_2})$ and $S3(G_{v^1_2})$ are greater than 0.

When the procedure is executed at root r of T^C, a P-graph G^P_r is $G_{v^0_1}$ that includes 4-disjoint paths $p_1 = 1, 2, 7, 3$, $p_2 = 4$, $p_3 = 5, 6$ and $p_4 = 9, 11, 10, 12, 8$, by applying formula (1). (See Fig. A·1 (c).) An S-graph G^S_r is $G_{v^0_6}$ that includes two scattering vertices 13 and 14. $S1(G_r) = (dp(G_r) + 1) - sp(G^P_r) = (2 + 1) - 2 = 1 \geq 0$, $S2(G_r) = S1(G_r) - po(G^S_r) = 1 - 1 = 0 \leq 0$. Therefore, G_r has no STNT, since $S2(G_r)$ is less than 0. A new path $1, 2, 7, 3, 14, 4, 13, 5, 6$ is constructed and this subgraph $G_{v^1_2}$ is applicable to Case 2. Then, a potential terminal 15 is connected to a non-terminal vertex 1 that is one terminal of the new path. G has two disjoint paths $p_1 = 15, 1, 2, 7, 3, 14, 4, 13, 5, 6$ and $p_2 = 9, 11, 10, 12, 8$, and two terminals of p_1 are potential terminals and one terminal of p_2 is a potential terminal. We can construct an STNT on G by connecting another terminal 9 of p_2 to a vertex 13 of p_1.

Shin-ichi Nakayama is an Associate Professor at the Department of Mathematical Sciences, Tokushima University. He received B.E., M.E. and Dr. of Engineering from Toyohashi University of Technology, in 1991, 1993 and 1996, respectively. His research interests include graph theory, network theory and parallel algorithms.

Shigeru Masuyama was born in Kyoto, Japan on April 9, 1952. He received the B.S., M.S. and Ph.D. degrees in Engineering (Applied Mathematics and Physics) from Kyoto University, Kyoto, in 1977, 1979 and 1983, respectively. He was with the Department of Applied Mathematics and Physics, Faculty of Engineering, Kyoto University from 1984 to 1989 as an Assistant Professor. He joined the Department of Knowledge-Based Information Engineering, Toyohashi University of Technology in 1989 and now is a Professor. His research interest includes computational graph theory, combinatorial optimization, parallel algorithms and natural language processing. Dr. Masuyama is a member of the OR society of Japan, the Information Processing Society of Japan, the Institute of Systems, Control, Information Engineers of Japan and the Association for Natural Language Processing.