Discrete Applied Mathematics 40 (1992) 411-421
North-Holland

Set to set broadcasting in communication networks

Hsun-Ming Lee and Gerard J. Chang

Institute of Applied Mathematics, National Chiao Tung University, Hsinchu 30050, Taiwan, ROC

Received 15 January 1990
Revised 1 December 1990

Abstract

Lee, H.-M. and G.J. Chang, Set to set broadcasting in communication networks, Discrete Applied Mathematics 40 (1992) 411–421.

Suppose $G = (V,E)$ is a graph whose vertices represent people and edges represent telephone lines between pairs of people. Each person knows a unique message and is ignorant of the messages of other people at the beginning. These messages are then spread by telephone calls. In each call, two people exchange all information they have so far in exactly one unit of time. Suppose A and B are two nonempty subsets of V. The main purpose of this paper is to study the minimum number $b(A,B,G)$ of telephone calls by which A broadcasts to B; and the minimum time $t(A,B,G)$ such that A broadcasts to B. In particular, we give an exact formula for $b(A,B,K_n)$ and linear-time algorithms for computing $b(A,B,T)$ and $t(A,B,T)$ of a tree T.

Keywords. Gossip, broadcast, complete graph, tree, algorithm.

1. Introduction

Gossiping and broadcasting problems have been extensively studied for several decades; see [11] for a survey. In these problems, there are n people. Each person knows a unique message and is ignorant of the messages of other people at the beginning. These messages are then spread by telephone calls. In each call, two people exchange all information they have so far in exactly one unit of time. Suppose each person can only call some of the people. We use a graph $G=(V,E)$ to represent these information as follows. $V=\{1,\dots,n\}$ is the set of these n people; and

Correspondence to: Professor G.J. Chang, Institute of Applied Mathematics, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu, Taiwan.

* Supported in part by the National Science Council of the Republic of China.

$(i,j) \in E$ if and only if i and j can call each other. A typical example is the complete graph K_n which represents that each pair of people can call each other.

The *gossip problem* is to determine the *gossip number* of a graph G, denoted by $g(G)$, which is the minimum number of telephone calls for all people to know all messages. For any connected graph G, it is well known that $g(G)=0$ for $n=1$, $g(G)=1$ for $n=2$, $g(G)=3$ for $n=3$, $g(G)=2n-4$ for $n \geq 4$ and G has a 4-cycle, $g(G)=2n-3$ for $n \geq 4$ and G has no 4-cycle; see [1–5, 7–9, 12, 14, 15, 18, 20].

The *gossip time problem* is similar to the gossip problem except that telephone calls are allowed to occur simultaneously. The problem is to determine the minimum time $\text{gt}(G)$ for completing gossip in a graph G. This problem is not well solved as the gossip problem. The only results, as we know, are the following. For the path P_n of n vertices, $\text{gt}(P_n)=n$ when n is odd and $\text{gt}(P_n)=n-1$ when n is even; see [6]. For the grid graph $G_{m,n}$ with $m \geq 2$ and $n \geq 2$, $\text{gt}(G_{m,n})$ is equal to the diameter of $G_{m,n}$ except $\text{gt}(G_{3,3})=5$; see [6]. For the complete graph K_n, $\text{gt}(K_n)=\lceil \log_2 n \rceil$ if n is even and $\text{gt}(K_n)=\lceil \log_2 n \rceil + 1$ if n is odd; see [2, 13, 15].

A third problem of this type is the *broadcast* (*time*) *problem*. Again, telephone calls are allowed to occur simultaneously. The problem is to find the minimum time $\text{bt}(x, G)$ for person x to send his/her message to everyone else. The *broadcast time* $\text{bt}(G)$ of a graph $G=(V,E)$ is $\min\{\text{bt}(x,G): x \in V\}$; and the *broadcast center* $\text{BC}(G)$ is $\{x \in V: \text{bt}(x,G)=\text{bt}(G)\}$. [19] gave a linear-time algorithm for finding $\text{bt}(T)$ and $\text{BC}(T)$ of a tree T. $\text{BC}(T)$ is always a star as the algorithm shows. They also presented Johnson's proof that the problem of computing $\text{bt}(x,G)$ in general graphs is NP-complete.

As a generalization of the above three problems, [16] introduced the idea of set to set broadcasting. Suppose A and B are two nonempty sets of vertices in a connected graph $G=(V,E)$. The *set to set broadcast number from A to B*, denoted by $b(A,B,G)$, is the minimum number of telephone calls by which *A broadcasts to B*, i.e., all people in A send their messages to all people in B. The *set to set broadcast time from A to B*, denoted by $t(A,B,G)$, is the minimum time such that A broadcasts to B under the assumption that telephone calls are allowed to occur simultaneously. We will use $b(a,B,G)$ for $b(\{a\},B,G)$ and $t(a,B,G)$ for $t(\{a\},B,G)$. It is easy to see that $b(V,V,G)=g(G)$, $t(V,V,G)=\text{gt}(G)$ and $t(x,V,G)=\text{bt}(x,G)$ are precisely the previous three problems.

If Q is a sequence of telephone calls by which A broadcasts to B, then the inverse sequence of Q is such that B broadcasts to A. Consequently, both functions $b(A,B,G)$ and $t(A,B,G)$ are symmetric in A and B, i.e., $b(A,B,G)=b(B,A,G)$ and $t(A,B,G)=t(B,A,G)$. It is also clear that these two functions are monotone, i.e., $b(A,B,G) \geq b(A',B',G)$ and $t(A,B,G) \geq t(A',B',G)$ whenever $A' \subseteq A$ and $B' \subseteq B$.

The main purpose of this paper is to study $b(A,B,G)$ and $t(A,B,G)$. Section 2 gives an exact formula of $b(A,B,K_n)$. Section 3 gives a linear-time algorithm for computing $b(A,B,T)$ of a tree T and Section 4 gives a linear-time algorithm for $t(A,B,T)$.

2. Exact formula for $b(A, B, K_n)$

This section studies the set to set broadcast number $b(A,B,K_n)$ from A to B in the complete graph K_n. We shall give an exact formula of $b(A,B,K_n)$ in terms of $|A|$, $|B|$, $t=|A\cap B|$ and the gossip number $g(K_t)$. Note that $g(K_1)=0$, $g(K_2)=1$, $g(K_3)=3$ and $g(K_t)=2t-4$ for $t\geq 4$. For simplicity, let $g(K_0)=-1$.

Theorem 2.1. *If* $|A|=r$, $|B|=s$ *and* $|A\cap B|=t$, *then* $b(A,B,K_n)=r+s-2t+g(K_t)$.

Remark. [16] proved the theorem for $t\leq 3$ and $b(A,B,K_n)\leq r+s-2t+g(K_t)$ for $t\geq 4$.

Proof. For the case of $t\geq 1$, choose a vertex $x\in A\cap B$. In order that A broadcasts to B, first $A-B$ broadcasts to x by using $r-t$ calls. Secondly, vertices in $A\cap B$ make a complete gossip by $g(K_t)$ calls. At this moment, x knows the messages of all vertices in A. Finally, x sends these messages to all people in $B-A$ by $s-t$ calls. This gives $b(A,B,K_n)\leq r+s-2t+g(K_t)$. The case of $t=0$ is similar except that now we choose $x\in A$.

Next we shall prove that $b(A,B,K_n)\geq r+s-2t+g(K_t)$. Suppose $m=b(A,B,K_n)$ and $c_1,c_2,\dots,c_m$ is a sequence of telephone calls by which A broadcasts to B. Consider the graph G with $V(G)=A\cup B$ and $E(G)=\{c_1,\dots,c_m\}$. Note that G is connected. For the case of $t=0$, the connectivity of G gives $|E(G)|\geq|V(G)|-1$ and so

$$b(A,B,K_n)=m=|E(G)|\geq|V(G)|-1=r+s-1=r+s-2t+g(K_0).$$

For the case of $t\geq 1$, we consider the graph G^* obtained from G by shrinking $A\cap B$ to a vertex x^*. Note that $|V(G^*)|=r+s-2t+1$. Let T^* be a spanning tree of G^*. T^* corresponds to t disjoint trees $T_1,\dots,T_t$ in G; each T_i contains exactly one vertex x_i in $A\cap B$. For each call $c_i=(x,y)$ with $x\in T_j$ and $y\in T_k$, let $c_i^*=\text{null}$ if $j=k$ or equivalently $c_i\in E(T^*)$; $c_i^*=(x_j,x_k)$ if $j\neq k$ or $c_i\notin E(T^*)$. Replace c_i by c_i^* in the sequence $c_1,c_2,\dots,c_m$ to get a sequence of $m^*=m-|E(T^*)|$ calls among vertices in $A\cap B$.

Denotes Q_i the sequence of calls $c_1,\dots,c_i$ and Q_i^* the sequence $c_1^*,\dots,c_i^*$. For a sequence Q of calls and a vertex x, let $Y(x,Q)$ be the set of vertices in $A\cap B$ whose messages x learns after the calls in Q are made.

Claim. $Y(x_i,Q_j^*)\supseteq Y(v,Q_j)$ *for* $v\in V(T_i)$, $1\leq i\leq t$, $0\leq j\leq m$.

If the claim holds, then we have that $Y(x_i,Q_m^*)\supseteq Y(x_i,Q_m)=A\cap B$, i.e., the m^* calls Q_m^* in $A\cap B$ make each person in $A\cap B$ knows the message of everyone else. So

$$g(K_t)\leq m^*=m-|E(T^*)|=m-(|V(T^*)|-1)=m-(r+s-2t),$$

i.e., $b(A,B,K_n)=m\geq r+s-2t+g(K_t)$.

Now, we shall prove the Claim by induction on j. The case of $j=0$ is true since $Y(x_i, Q_0^*) = Y(x_i, Q_0) = \{x_i\}$ and $Y(v, Q_0) = \emptyset$ for $v \notin A \cap B$. Suppose the Claim holds for $j-1$. Now consider the sequence $Q_j = Q_{j-1} c_j$ where $c_j = (v_1, v_2)$ with $v_1 \in T_a$ and $v_2 \in T_b$. Note that $Y(x_i, Q_j^*) = Y(x_i, Q_{j-1}^*)$ for $x_i \notin \{x_a, x_b\}$; and $Y(v, Q_j) = Y(v, Q_{j-1})$ for $v \notin \{v_1, v_2\}$. By induction hypothesis, we only have to prove the Claim for $x_i \in \{x_a, x_b\}$ and $v \in \{v_1, v_2\}$. By induction hypothesis, $Y(x_a, Q_{j-1}^*) \supseteq Y(v_1, Q_{j-1})$ and $Y(x_b, Q_{j-1}^*) \supseteq Y(v_2, Q_{j-1})$. Hence,

$$\begin{aligned} Y(x_a, Q_j^*) = Y(x_b, Q_j^*) &= Y(x_a, Q_{j-1}^*) \cup Y(x_b, Q_{j-1}^*) \\ &\supseteq Y(v_1, Q_{j-1}) \cup Y(v_2, Q_{j-1}) = Y(v_1, Q_j) = Y(v_2, Q_j). \end{aligned}$$

The first two equalities follow from the fact that $Q_j^* = Q_{j-1}^*$ when $a=b$ and $Q_j^* = Q_{j-1}^* c_j^*$ with $c_j = (x_a, x_b)$ when $a \neq b$. The last two equalities follow from that $Q_j = Q_{j-1} c_j$ with $c_j = (v_1, v_2)$. These prove the Claim. □

3. A linear-time algorithm for $b(A, B, T)$ of a tree T

In this section, we study the set to set broadcast number $b(A, B, T)$ of a tree T, where A and B are two nonempty subsets of $V(T)$. We shall give a linear-time algorithm which finds $b(A, B, T)$ and gives a sequence of $b(A, B, T)$ calls by which A broadcasts to B. A *leaf* in a tree is a vertex which is adjacent to exactly one vertex.

Lemma 3.1. *If T' is a subtree of a tree T and $A, B \subseteq V(T')$, then $b(A, B, T) = b(A, B, T')$ and $t(A, B, T) = t(A, B, T')$.*

Proof. It suffices to prove the case that $V(T) - V(T')$ contains only one vertex x which is a leaf in T and is adjacent to y. The lemma follows from the fact that the deletion of the call (x, y) from a sequence of calls by which A broadcasts to B is still a sequence of calls by which A broadcasts to B. □

By Lemma 3.1, from now on we can assume that all leaves of T are in $A \cup B$. If it is not the case, we may repeatedly delete a leaf not in $A \cup B$ until all leaves are in $A \cup B$.

Let T^* be the only minimal subtree of T such that each component of $T - T^*$ contains vertices only in $V(T) - B$ or only in $V(T) - A$. We call components of $T - T^*$ *branches* of T (with respect to A and B). Each branch has a unique vertex r adjacent to a unique vertex r^* in T^*; r is called the *root* of the branch and r^* the *co-root*. A branch that contains vertices only in $V(T) - B$ (respectively $V(T) - A$) is called an *A-branch* (respectively *B-branch*).

Figure 1 shows an example of T and T^* in which $A = \{1, 2, 8, 9\}$ and $B = \{3, 8, 11, 13\}$. Vertices of $A - B$, $B - A$, $A \cap B$ are labeled by a, b, c respectively.

Roughly speaking, the algorithm first broadcasts the messages in A-branches to

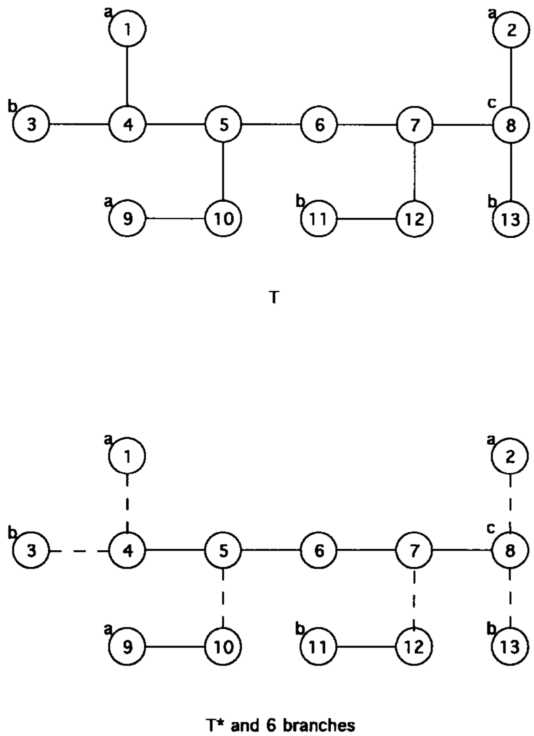

Fig. 1.

T^*. Secondly, the algorithm makes a complete gossip in T^*. Finally, it broadcasts messages from T^* to B-branches. For an A-branch, we broadcast messages from leaves toward to the root and then to the co-root. For a B-branch, we broadcast messages from its co-root in T^* to the root and then to other vertices. More precisely, we have the following linear-time algorithm.

Algorithm B1.
Input: A tree $T=(V,E)$ and two nonempty sets $A,B \subseteq V$.
Output: The set to set broadcast number $b(A,B,T)$ and a sequence Q of $b(A,B,T)$ calls by which A broadcasts to B.

Method:
$L(x) \leftarrow a$ for all $x \in A - B$;
$L(x) \leftarrow b$ for all $x \in B - A$;
$L(x) \leftarrow c$ for all $x \in A \cap B$;
$L(x) \leftarrow 0$ for all $x \notin A \cup B$;
{construct T^*}
$T^* \leftarrow T$; $Q_1 \leftarrow \emptyset$; $Q_2 \leftarrow \emptyset$;

while (there is a leaf x of T^* with $L(x)=a$ or $L(x)=b$) **do**
  let y be the unique vertex adjacent to x;
  if $L(x)=a$ **then** $Q_1 \leftarrow Q_1(x,y)$ **else** $Q_2 \leftarrow (x,y)Q_2$;
  if $L(y)=0$ **then** $L(y) \leftarrow L(x)$;
  if $L(y) \neq 0$ **and** $L(x) \neq L(y)$ **then** $L(y) \leftarrow c$; {y is a co-root}
  $T^* \leftarrow T^* - x$;
end;
{complete gossip in T^*}
$T' \leftarrow T^*$;
while (T' has at least two edges) **do**
  find a leaf x adjacent to y;
  $Q_1 \leftarrow Q_1(x,y)$; $Q_2 \leftarrow (x,y)Q_2$;
  $T' \leftarrow T' - x$;
end;
if T' has one edge (x,y) **then** $Q_1 \leftarrow Q_1(x,y)$;
{get answer}
$Q \leftarrow Q_1 Q_2$; $b(A,B,T) \leftarrow |Q|$.

For the example in Fig. 1, we can use Algorithm B1 to get an optimum sequence of calls by which A broadcasts to B: (1,4), (2,8), (9,10), (10,5), (4,5), (5,6), (6,7), (7,8), (6,7), (5,6), (4,5), (3,4), (13,8), (12,7), (11,12).

Theorem 3.2. *Algorithm* B1 *works. Moreover*, $b(A,B,T)=\alpha$ *where*

$$\alpha = \begin{cases} |E(T)|, & \text{if } |E(T^*)| \leq 1, \\ |E(T)| + |E(T^*)| - 1, & \text{if } |E(T^*)| \geq 2. \end{cases}$$

Proof. From the algorithm, it is easy to see that Q is a sequence of α calls by which A broadcasts to B. This shows that $b(A,B,T) \leq \alpha$.

Suppose Q^* is a sequence of $b(A,B,T)$ calls by which A broadcasts to B. We first observe that every edge of T must appear in Q^* at least once, for if an edge of T is not in Q^*, the forests induced by Q^* has a component containing a vertex $x \in A$ and another component containing a vertex $y \in B$. In this case, y cannot learn the information of x. This shows that $|Q^*| \geq \alpha$ for $|E(T^*)| \leq 1$. For the case of $|E(T^*)| \geq 2$, suppose $|Q^*| < \alpha = |E(T)| + |E(T^*)| - 1$. Since every edge of T appears in Q^* at least once, T^* has at least two edges e_1 and e_2 which appear in Q^* exactly once. The removal of e_1 and e_2 from T results in three subtrees, each of the two "extreme" subtrees must contain at least one vertex in A and one in B. Thus, there is no way for the vertex of B in one of the "extreme" subtrees to receive the information of the vertex of A in the other extreme subtree. Then, $|Q^*| \geq \alpha$ in any case and so the theorem holds. □

4. A linear-time algorithm for $t(A, B, T)$ of a tree T

In this section, we study the set to set broadcast time $t(A, B, T)$ for a tree T, where A and B are two nonempty subsets of $V(T)$. We shall give a linear-time algorithm for computing $t(A, B, T)$. Our method employs the algorithm in [19] for finding the broadcast center $\mathrm{BC}(T)$ of a tree T. Let us first review their results.

The *broadcast time of x in a graph G* is $\mathrm{bt}(x, G) \equiv t(x, V(G), G)$. The *broadcast time of G* is $\mathrm{bt}(G) = \min\{\mathrm{bt}(x, G): x \in V(G)\}$. The *broadcast center of G* is $\mathrm{BC}(G) = \{x \in V(G): \mathrm{bt}(x, G) = \mathrm{bt}(G)\}$.

If (u, v) is an edge of a tree T, then $T(u, v)$ and $T(v, u)$ will denote the subtrees of T consisting of the components of $T - (u, v)$ containing u and v, respectively. If $v_1, \ldots, v_k$ are the neighbors of a vertex w in a tree T, then we have the following formula.

$$\mathrm{bt}(w, T) = \min_{\pi \in S_k} \max_{1 \le i \le k} \{\mathrm{bt}(v_i, T_i) + \pi(i)\}, \tag{1}$$

where S_k is the set of all permutations of $\{1, \ldots, k\}$ and $T_i = T(v_i, w)$ for $1 \le i \le k$. The meaning of (1) is that w sends its message to v_i at the $\pi(i)$th time unit and then v_i broadcasts to $V(T_i)$ in $\mathrm{bt}(v_i, T_i)$ time units. Suppose $\mathrm{bt}(v_1, T_1) \ge \mathrm{bt}(v_2, T_2) \ge \cdots \ge \mathrm{bt}(v_k, T_k)$. It is clear that $\pi(i) = i$ minimizes (1), i.e.,

$$\mathrm{bt}(w, T) = \max_{1 \le i \le k} \{\mathrm{bt}(v_i, T_i) + i\}. \tag{2}$$

Based on (2), [19] gave the following linear-time algorithm for finding $\mathrm{bt}(T)$ and $\mathrm{BC}(T)$.

Algorithm BROADCAST.

let W be the set of all leaves of T;
for each $w \in W$ **do** $t(w) \leftarrow 0$;
$U \leftarrow \emptyset$; $T' \leftarrow T$;
while $|V(T')| \ge 2$ **do**
  let $w \in W$ satisfy $t(w) = \min\{t(w_i): w_i \in W\}$;
  let v be the vertex adjacent to w in T';
(*) $W \leftarrow W - \{w\}$; $U \leftarrow U \cup \{w\}$; $T' \leftarrow T' - \{w\}$;
  if v is a leaf of T' **then**
   [let v be adjacent to labeled vertices $u_1, u_2, \ldots, u_k$ in U ordered so that $t(u_1) \ge t(u_2) \ge \cdots \ge t(u_k)$;
   $t(v) \leftarrow \max\{t(u_i) + i: 1 \le i \le k\}$;
   $W \leftarrow W \cup \{v\}$;]
end;
(**) let v be the only vertex left in T';
let the neighbors of v in T be $u_1, u_2, \ldots, u_k$ with $t(u_1) \ge t(u_2) \ge \cdots \ge t(u_k)$;
let j be the smallest integer such that $t(u_j) + j = \max\{t(u_i) + i: 1 \le i \le k\}$;
$\mathrm{bt}(T) \leftarrow t(v)$; $\mathrm{BC}(T) \leftarrow \{v, u_1, u_2, \ldots, u_j\}$.

The following four results from [19] are useful in this paper.

(R1) Suppose in step ($*$) that Algorithm BROADCAST deletes vertex w from W and adds it to U. If v is adjacent to w in T', then $t(w)=\mathrm{bt}(w, T(w,v))$.

(R2) Let v be the single remaining vertex in step ($**$) of Algorithm BROADCAST, and let $u_1, u_2, \ldots, u_k$ be the neighbors of v in T ordered so that $t(u_1) \geq t(u_2) \geq \cdots \geq t(u_k)$. Let j be the smallest integer such that $t(u_j)+j=\max\{t(u_i)+i: 1 \leq i \leq k\}$. Then $\mathrm{BC}(T)=\{v, u_1, \ldots, u_j\}$ and $\mathrm{bt}(T)=t(v)=t(u_j)+j$. Note that $\mathrm{BC}(T)$ is a star whose center is v.

(R3) Let x be a vertex in a tree T and the shortest distance from x to a vertex y in $\mathrm{BC}(T)$ is k. Then $\mathrm{bt}(x, T)=k+\mathrm{bt}(y, T)=k+\mathrm{bt}(T)$.

(R4) Suppose v is the center of the broadcast center $\mathrm{BC}(T)=\{v, u_1, u_2, \ldots, u_j\}$ of a tree T. For $1 \leq r \leq j$, there is an optimum call sequence whose first call is (u_r, v), by which v (respectively u_r) broadcasts to $V(T)$ in $\mathrm{bt}(T)$ time units.

Now we are ready to explain our algorithm for computing $t(A, B, T)$. Again, we assume that all leaves of T are in $A \cup B$. Let T_a (respectively T_b) be the unique minimal subtree of T that contains A (respectively B). Note that all leaves of T_a (respectively T_b) are in A (respectively B). Roughly speaking, our method consists of three steps: (1) A broadcasts to a vertex $x \in \mathrm{BC}(T_a)$, (2) x broadcasts to a vertex $y \in \mathrm{BC}(T_b)$, (3) y broadcasts to B. More precisely, we have the following linear-time algorithm.

Algorithm B2.

$T_a \leftarrow T$; **while** (T_a has a leaf x not in A) **do** $T_a \leftarrow T_a - x$;
$T_b \leftarrow T$; **while** (T_b has a leaf x not in B) **do** $T_b \leftarrow T_b - x$;
Use Algorithm BROADCAST to find $\mathrm{BC}(T_a)=\{v_1=u_{10}, u_{11}, u_{12}, \ldots, u_{1r}\}$ and $\mathrm{BC}(T_b)=\{v_2=u_{20}, u_{21}, u_{22}, \ldots, u_{2s}\}$;
if $\mathrm{BC}(T_a) \cap \mathrm{BC}(T_b)$ contains two vertices which are $\{v_1, u_{1i}\}=\{v_2, u_{2j}\}$
then [$A \subseteq V(T_a)$ broadcasts to v_1 by an optimum call sequence of length $\mathrm{bt}(T_a)$, whose last call is (v_1, u_{1i}); {This and next steps make use of (R4).}
v_1 broadcasts to $B \subseteq V(T_b)$ by an optimum call sequence of length $\mathrm{bt}(T_b)$, whose first call is $(v_2, u_{2j})=(v_1, u_{1i})$, however the first call was made in the previous step and so can be saved;
$t(A, B, T) \leftarrow \mathrm{bt}(T_a)+\mathrm{bt}(T_b)-1$;]
else [let $P=u_{1i}, \ldots, u_{2j}$ be a shortest path from $\mathrm{BC}(T_a)$ to $\mathrm{BC}(T_b)$;
$A \subseteq V(T_a)$ broadcasts to u_{1i} in T_a by $\mathrm{bt}(T_a)$ calls;
u_{1i} broadcasts to u_{2j} along P by $|P|$ calls;
u_{2j} broadcasts to $B \subseteq V(T_b)$ in T_b by $\mathrm{bt}(T_b)$ calls;
$t(A, B, T) \leftarrow \mathrm{bt}(T_a)+\mathrm{bt}(T_b)+\mathrm{distance}(\mathrm{BC}(T_a), \mathrm{BC}(T_b))$.]

We need the following lemmas to verify the correctness of Algorithm B2, i.e., Theorem 4.5.

Lemma 4.1. *Suppose A and B are two nonempty sets of vertices in a tree T. If B^* is the set of all vertices in a path between any two vertices in B, then $t(A,B,T)=t(A,B^*,T)$.*

Proof. Suppose Q is a sequence of calls by which A broadcasts to B. For any x in A and y in B^*, there are two vertices $z, w \in B$ such that $y \in P(z, w)$ which is the unique $z-w$ path in T. Either $P(x,y) \subseteq P(x,z)$ or $P(x,y) \subseteq P(x,w)$. Since x broadcasts to z and w through Q, $P(x,z) \subseteq Q$ and $P(x,w) \subseteq Q$. Hence $P(x,y) \subseteq Q$, i.e., x broadcasts to y through Q. This shows that Q is also a sequence of calls by which A broadcasts to B^*. □

Lemma 4.2. *If $\mathrm{BC}(T)=\{v,u_1,u_2,\ldots,u_j\}$ is the broadcast center of a tree T, then $\mathrm{bt}(v, T(v,u_i))=\mathrm{bt}(T)-1$ for $1\le i\le j$.*

Proof. By the choice of j in (R2), $t(u_h)+h\le t(u_j)+j-1$ for $1\le h<j$. This and $t(u_{j-1})\ge t(u_j)$ imply that $t(u_{j-1})=t(u_j)$ whenever $j>1$. By formula (2),

$$\mathrm{bt}(v, T(v,u_i))=\max\left\{\max_{1\le h<i}\{t(u_h)+h\},\ \max_{i<h\le k}\{t(u_h)+h-1\}\right\}.$$

Note that the maximum achieves by $h=j$ when $i<j$ and by $h=j-1$ when $i=j>1$. In either case, $\mathrm{bt}(v, T(v,u_i))=t(u_j)+j-1=\mathrm{bt}(T)-1$. For the case of $i=j=1$, the above argument only gives $\mathrm{bt}(v, T(v,u_i))\le t(u_j)+j-1=\mathrm{bt}(T)-1$. However, at the moment when T' has only two vertices v and u_j in the while loop of Algorithm BROADCAST, $\mathrm{bt}(v, T(v,u_i))=t(v)\ge t(u_j)=\mathrm{bt}(G)-j=\mathrm{bt}(G)-1$. □

Lemma 4.3. *Suppose $\mathrm{BC}(T)=\{v,u_1,u_2,\ldots,u_j\}$ is the broadcast center of a tree T. Let $\bar{T}=\bigcup_{1\le i\le j}\{(v,u_i)+T(u_i,v)\}$. Then $\mathrm{BC}(\bar{T})=\mathrm{BC}(T)$, $\mathrm{bt}(\bar{T})=\mathrm{bt}(T)$ and $\mathrm{bt}(x,\bar{T})=\mathrm{bt}(x,T)$ for all $x\in V(\bar{T})$.*

Proof. The lemma follows from Algorithm BROADCAST and results (R1)–(R3). □

Lemma 4.4. *If T_1 and T_2 are two subtrees of T such that $V(T_1)\cap V(T_2)=\{x\}$, then $t(A,B,T)=t(A,x,T)+t(x,B,T)$.*

Proof. If Q_1 (respectively Q_2) is a sequence of calls by which A broadcasts to x (respectively x broadcasts to B), then Q_1Q_2 is a sequence of calls by which A broadcasts to B. So $t(A,B,T)\le t(A,x,T)+t(x,B,T)$. On the other hand, suppose $Q=c_1\ldots c_m$ is an optimal sequence of calls by which $A\subseteq V(T_1)$ broadcasts to $B\subseteq V(T_2)$. Permute the calls in Q to get a new sequence $Q^*=Q_1Q_2$, where each $Q_i=Q\cap E(T_i)$, such that calls in Q_i have the same order as in Q. Note that any subsequence of Q by which a vertex in A broadcasts its message to a vertex in B must have calls in $E(T_1)$ followed by calls in $E(T_2)$. Hence Q^* is also a sequence of calls by which A broadcasts to B. Thus

$$t(A,B,T)=m=|Q_1|+|Q_2|\geq t(A,x,T)+t(x,B,T). \qquad \square$$

Theorem 4.5. *Algorithm* B2 *gives an optimum sequence of calls by which A broadcasts to B. Furthermore,* $t(A,B,T)=\mathrm{bt}(T_a)+\mathrm{bt}(T_b)-1$ *when* $|\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)|\geq 2$, *and* $t(A,B,T)=\mathrm{bt}(T_a)+\mathrm{bt}(T_b)+\mathrm{distance}(\mathrm{BC}(T_a),\mathrm{BC}(T_b))$ *otherwise.*

Proof. Since $A\subseteq V(T_a)$ and $B\subseteq V(T_b)$, Algorithm B2 gives a sequence of calls by which A broadcasts to B. So $t(A,B,T)\leq \mathrm{bt}(T_a)+\mathrm{bt}(T_b)-1$ when $|\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)|\geq 2$, and $t(A,B,T)\leq \mathrm{bt}(T_a)+\mathrm{bt}(T_b)+\mathrm{distance}(\mathrm{BC}(T_a),\mathrm{BC}(T_b))$ otherwise.

Since all leaves of T_a (respectively T_b) are in A (respectively B), by Lemma 4.1, $t(A,B,T)=t(V(T_a),V(T_b),T)\geq t(V(\bar{T}_a),V(\bar{T}_b),T)$.

Case 1: $|\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)|\geq 2$. In this case, $\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)$ must contain v_1 and v_2; otherwise v_1 and v_2 together with two vertices in $\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)$ form a 4-cycle. We can assume $\{v_1,u_{1i}\}=\{v_2,u_{2j}\}\subseteq \mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)$, where $1\leq i\leq r$ and $1\leq j\leq s$.

Subcase 1.1: $v_1=v_2$. For each $1\leq h\leq s$, $T_{1h}=\bar{T}_a\cap T(v_1,u_{2h})$ and $T_{2h}=\bar{T}_b\cap \{(v_1,u_{2h})+T(u_{2h},v_1)\}$ are two subtrees of T which intersect only at v_1. Let $m=t(V(\bar{T}_a),V(\bar{T}_b),T)$ and $Q=c_1\dots c_m$ be a sequence of calls by which $V(\bar{T}_a)$ broadcasts to $V(\bar{T}_b)$. Since $V(T_{1h})\subseteq V(\bar{T}_a)$ and $V(T_{2h})\subseteq V(\bar{T}_b)$, Q is also a sequence of calls by which $V(T_{1h})$ broadcasts to $V(T_{2h})$. Note that $T_{1h}=\bar{T}_a$ when $u_{2h}\notin V(\bar{T}_a)$ or $T_{1h}=\bar{T}_a(v_1,u_{2h})$ when $u_{2h}\in V(\bar{T}_a)$. So $\mathrm{bt}(T_{1h})=\mathrm{bt}(\bar{T}_a)$ or $\mathrm{bt}(T_{1h})=\mathrm{bt}(\bar{T}_a)-1$ by Lemma 4.2; in any case $\mathrm{bt}(T_{1h})\geq \mathrm{bt}(\bar{T}_a)-1$. From Lemma 4.4, $Q^*=c_{\mathrm{bt}(\bar{T}_a)}\dots c_m$ is a sequence of calls by which $v_2=v_1$ broadcasts to T_{2h} for $1\leq h\leq s$. Hence Q^* is also a sequence of calls by which v_2 broadcasts to $V(\bar{T}_b)$. Then $m-\mathrm{bt}(\bar{T}_a)+1=|Q^*|\geq \mathrm{bt}(\bar{T}_b)$; i.e., $t(A,B,T)\geq m\geq \mathrm{bt}(\bar{T}_a)+\mathrm{bt}(\bar{T}_b)-1=\mathrm{bt}(T_a)+\mathrm{bt}(T_b)-1$ by Lemma 4.3.

Subcase 1.2: $v_1=u_{2j}$ *and* $v_2=u_{1i}$. In this case, $\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)=\{v_1,u_{1i}\}=\{v_2,u_{2j}\}$ otherwise there is a 3-cycle. $T_1=\bar{T}_a(v_1,v_2)$ and $T_2=(v_2,v_1)+\bar{T}_b(v_2,v_1)$ are two subtrees of T which intersect at v_1. Hence

$$\begin{aligned} t(A,B,T) &\geq t(V(\bar{T}_a),V(\bar{T}_b),T)\\ &\geq t(V(T_1),V(T_2),T)\\ &= t(V(T_1),v_1,T)+t(v_1,V(T_2),T) && \text{(by Lemma 4.4)}\\ &= \mathrm{bt}(\bar{T}_a)-1+\mathrm{bt}(\bar{T}_b) && \text{(by Lemmas 3.1 and 4.2)}\\ &= \mathrm{bt}(T_a)+\mathrm{bt}(T_b)-1 && \text{(by Lemma 4.3).} \end{aligned}$$

Case 2: $|\mathrm{BC}(T_a)\cap \mathrm{BC}(T_b)|\leq 1$. Suppose the shortest distance from $\mathrm{BC}(T_a)$ to $\mathrm{BC}(T_b)$ is along a u_{1i}-u_{2j} path P where $0\leq i\leq r$ and $0\leq j\leq s$. Let $T_1=\bar{T}_a+P$ if $i=0$, and $T_1=\bar{T}_a(u_{10},u_{1i})+(u_{10},u_{1i})+P$ otherwise. Let $T_2=\bar{T}_b$ if $j=0$, and $T_2=\bar{T}_b(u_{20},u_{2j})+(u_{20},u_{2j})$ otherwise. Then

$$t(A,B,T)\geq t(V(\bar{T}_a),V(\bar{T}_b),T)$$

$$\begin{aligned}&\geq t(V(\bar{T}_a)\cap V(T_1), V(\bar{T}_b)\cap V(T_2), T)\\&= t(V(\bar{T}_a)\cap V(T_1), u_{2j}, T) + t(u_{2j}, V(\bar{T}_b)\cap V(T_2), T)\\&\quad \text{(by Lemma 4.4)}\\&= \mathrm{bt}(\bar{T}_a) + |P| + \mathrm{bt}(\bar{T}_b)\\&\quad \text{(by Lemmas 3.1 and 4.2)}\\&= \mathrm{bt}(T_a) + \mathrm{bt}(T_b) + |P|\\&\quad \text{(by Lemma 4.3).} \qquad \square\end{aligned}$$

Acknowledgement

We are grateful to the referees for many suggestions about the revision of this paper.

References

[1] B. Baker and R. Shostak, Gossips and telephones, Discrete Math. 2 (1972) 191-193.
[2] A. Bavelas, Communication patterns in task-oriented groups, J. Acoust. Soc. Amer. 22 (1950) 725-730.
[3] G. Berman, The gossip problem, Discrete Math. 4 (1973) 91.
[4] R. Bumby, A problem with telephones, SIAM J. Algebraic Discrete Methods 2 (1981) 13-19.
[5] N. Cot, Extensions of the telephone problem, in: Proceedings Seventh SE Conference on Combinatorics, Graph Theory and Computing (Utilitas Mathematics, Winnipeg, 1976) 239-256.
[6] A. Farley and A. Proskurowski, Gossiping in grid graphs, J. Combin. Inform. System Sci. 5 (1980) 161-172.
[7] M.C. Golumbic, The general gossip problem, Tech. Rept., IBM Res. Rept. RC 4977, IBM (1974).
[8] A. Hajnal, E.C. Milner and E. Szemeredi, A cure for the telephone disease, Canad. Math. Bull. 15 (1972) 447-450.
[9] F. Harary and A.J. Schwenk, The communication problem on graphs and digraphs, J. Franklin Inst. 297 (1974) 491-495.
[10] F. Harary and A.J. Schwenk, Efficiency of dissemination of information in one-way and two-way communication networks, Behavioral Sci. 19 (1974) 133-135.
[11] S.M. Hedetniemi, S.T. Hedetniemi and A.L. Liestman, A survey of gossiping and broadcasting in communication networks, Networks 18 (1988) 319-349.
[12] D.J. Kleitman and J.B. Shearer, Further gossip problems, Discrete Math. 30 (1980) 151-156.
[13] W. Knodel, New gossips and telephones, Discrete Math. 13 (1975) 95.
[14] R. Labahn, The telephone problem for trees, Elektron. Informationsverarb. Kybernet. 22 (1986) 475-485.
[15] H.G. Landau, The distribution of completion times for random communication in a task-oriented group, Bull. Math. Biophys. 16 (1954) 187-201.
[16] D. Richards and A.L. Liestman, Generalizations of broadcasting and gossiping, Networks 18 (1988) 125-138.
[17] P. Schmitt, Spreading information by conferences, Discrete Math. 15 (1976) 305-306.
[18] A. Seress, Quick gossiping without duplicate transmissions, Graphs Combin. 2 (1986) 363-383.
[19] P.J. Slater, E. Cockayne and S.T. Hedetniemi, Information dissemination in trees, SIAM J. Comput. 10 (1981) 692-701.
[20] R. Tijdeman, On a telephone problem, Nieuw Arch. Wisk. 19 (1971) 188-192.