Vol.2011-AL-134 No.16
2011/3/7

確率的な通信時間を持つネットワークにおけるブロードキャスト時間の計算手法

Ei Ando †1 Joseph Peters†2

本稿では，計算機ネットワークにおいて各計算機を互いに接続する通信路でのメッセージ伝達にかかる時間（通信時間）がランダムな時間がかかるものと考えた場合に，そのネットワーク内でのブロードキャストにかかる時間の分布関数を計算する方法を考える．各通信路の通信時間は互いに独立な確立変数とする．本稿ではまずこの問題が $\#P-$ 完全であることを証明し，その後ブロードキャスト時間の分布関数を計算するアルゴリズム BC-PDF について述べる．BC-PDF は一般のグラフにおいてはネットワーク規模について指数的な処理時間が必要であるものの、最大極小カットのサイズが定数 k で抑えられるようなネットワークで，各枝重みが分散 1 の指数分布に従う場合，グラフ規模の多項式時間で計算を完了する．

An Algorithm for Compung the Broadcast Time in Networks with Stochastic Transmission Time

Ei Ando †1 and Joseph Peters†2

We consider the problem of estimating the broadcast time in computer networks with random transmission times. Since the transmission times are random variables, the broadcast time is also a random variable; we are to compute the (cumulative) distribution function $F_B(x)$ of the broadcast time. We first show that the problem is $\#P$-complete. Then, we show an algorithm that computes $F_B(x)$. Although our algorithm may take exponential time in the network size for general network topology, our algorithm finishes in polynomial time with respect to network size in case the transmission time obey the exponential distribution with variance 1, and the network's maximum minimal cut size is bounded by a constant k.

1. Introduction

Consider a network where processors are connected by communication links, where the network topology is given by a connected graph $G = (V, E)$ with vertex set $V = \{v_0, \ldots, v_{n-1}\}$ and undirected edge set $E \subseteq \{\{u, v\} \mid u, v \in V\}$. The *broadcast* is a procedure in which a processor v_0, the broadcast source, send a short message to its neighbors and the other processors copy and send the message to their neighbors until every processor receives the message. In this paper, we consider the problem of estimating the time duration of the broadcast in networks where the transmission time of each communication link is given as a random variable. Since the transmission times are random variables, the broadcast time itself is also a random variable; we are to compute the (cumulative) distribution function $F_B(x)$ of the broadcast time.

If the transmission times are given as static values, there are efficient algorithms. We are to compute the shortest path tree by, for example, Dijkstra's algorithm considering the transmission time as the edge length. Then the distance from the source to the farthest vertex is the broadcast time.

However, the transmission time is not always static. For example, in the multi-commodity network like the Internet, the communication links are shared by many people. Since it is not realistic that every user knows the behavior of all the other users, the traffic of the network seem to have some randomness to the users.

If the transmission time is random, it seems hard to compute $F_B(x)$. It is partly because the farthest vertex from the source is not stable; the distance between two vertices may be varying; any simple path between two path may be the shortest path with some probability. Actually, we can prove the problem to be $\#P$-complete in the way that is very similar to Hagstrom's way[4] of proving that computing the distribution function of finishing time of all tasks in PERT networks with random project durations is $\#P$-complete.

Ando et al. have been working on the graph optimization problems with mutually

†1 Faculty of Computer and Information Science, Sojo University, 4-22-1, Ikeda, Kumamoto.
†2 School of Computing Science, Simon Fraser University, Burnaby, Canada

independent random edge lengths[1)–3)]. Although it seems hard to compute the distribution function of the longest path length in directed acyclic graphs with random edge weights in most of the case, they showed that the longest path length distribution function can be computed in polynomial time in the graph size if the incident graph's antichain size is bounded by a constant k and edge lengths' distributions satisfy some conditions[2)]. They also proposed some approximation algorithms for the graph optimization problems in[1),3)].

This paper is organized as follows. In Section 2, we explain the basic definitions. Then we prove that computing the broadcast time with random transmission times is $\#P$-complete, in Section 3. In Section 4, we show an algorithm that computes $F_B(x)$ analytically. Our algorithm BC-PDF computes the broadcast time distribution function in polynomial time in the network size if the maximum minimal cut size is bounded by a constant k and the transmission times obey the exponential distribution with variance 1. We conclude this paper in Section 5.

2. Preliminaries

2.1 Definitions Concerned with Network Models

In this paper, we consider the broadcast time in a computer network of n processors and m communication links that connects the processors through which the processors can send and receive short messages. Let the topology of the network is given by an undirected graph $G = (V, E)$; that is, each processors corresponds to a vertex, and each communication link between two processors u, v corresponds to an edge between the two corresponding vertices. We consider that a mutually independent random variable X_e, *transmission time*, is associated with an edge $e \in E$.

The *broadcast* is defined as follows. First, one processor v_0, the broadcast source, sends a (broadcast) message to each of its neighbors. Then the neighbors eventually receive the message. Immediately after that, the neighbors copy the message and send it to the adjacent processors that have not received the message. The entire procedure finishes when all processors receive the message. We call v as the *source* of the broadcast.

2.2 Definitions Concerned with Probabilities

It is well known that the distribution function of the sum of the two mutually independent random variables Y and Z is given by the following

$$P(Y + Z \le x) = \iint_{\mathbb{R}^2} P(y + z \le x) f_Y(y) f_Z(z) dydz, \tag{1}$$

where $f_Y(x)$ and $f_Z(x)$ are the density functions of Y and Z, respectively. Also, the distribution function of the max of two mutually independent random variables Y and Z is given by the following

$$\begin{aligned} P(\max\{Y, Z\} \le x) &= P(Y \le x \wedge Z \le x) \\ &= \iint_{\mathbb{R}^2} P(y \le x \wedge z \le x) f_Y(y) f_Z(z) dydz. \end{aligned} \tag{2}$$

This equals to the product $F_Y(x)F_Z(x)$ of the distribution functions $F_Y(x)$ and $F_Z(x)$ of Y and Z.

The above basic computations suggests that, as long as the broadcast is performed in a network with a tree topology, we can compute the broadcast time distribution function. The running time of the computation depends on the definitions of the transmission time distribution functions.

2.3 Probability Distribution of Static Values

A static value c may be considered as a random variable that has the step function $H(x - c)$ as its distribution function, where $H(x)$ is defined by

$$H(x) = \begin{cases} 0 & x \le 0 \\ 1 & x > 0. \end{cases} \tag{3}$$

We define $H(0) = 0$ to keep the later calculations simpler. Although $H(x)$ is not differentiable at $x = 0$, we consider the derivative of $H(x)$ is delta function $\delta(x)$ satisfying the following properties.

$$\delta(x) = 0 \text{ if } x \ne 0 \tag{4}$$

$$\int_a^b \delta(x)dx = \begin{cases} 1 & \text{if } a < 0 \text{ and } b > 0 \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

and

$$\int_a^b f(x)\delta(x - c)dx = \begin{cases} f(c) & \text{if } a < c \text{ and } b > c \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Vol.2011-AL-134 No.16
2011/3/7

By using $H(x)$, we write

$$\int_a^b f(x)\delta(x-c)dx = H(c-a)H(b-c)f(c). \tag{7}$$

The function $H(x)$ can be understood as an exponential distribution function with very small variance, that is,

$$H(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ 1-\exp(-x/\varepsilon) & \text{if } x > 0. \end{cases} \tag{8}$$

Then the derivative of $H(x)$ is an exponential density function with small variance. By assuming that ε is sufficiently small, we consider that $H(x)$ and $\delta(x)$ approximately satisfy (3 – 7).

3. $\#P$-Completeness

Here we prove that the problem of computing the distribution function of the broadcast time in networks with random transmission time is $\#P$-complete. By the similar way in which Hagstrom proved that computing the distribution function of the stochastic longest path length in DAGs is $\#P$-complete, we show that computing the distribution function of the broadcast time can be used to solve the reliability problem, which is proved to be a $\#P$-complete problem by Ball and Provan[5]. We prove that if we can compute the distribution function of the broadcast time in one class of undirected graph, we can solve the reliability function problem. We then show that the broadcast time distribution function can be computed by a counting, which shows that the problem is $\#P$-complete.

Let us see the definition of reliability problem. We consider a *transportation graph* G_t that is defined in the following. We first consider a directed bipartite graph G_d whose vertex set is $V_d = A \cup B$ and edge set is $E_d \subseteq A \times B$. The vertex set of G_t is given by adding two vertices s, t to V_d; as for the edge set of G_t, we add, to E_d, the edges from s to every vertex of A and the edges from every vertex of B to t. Then, given the probability that each edge is available, the reliability problem is to compute the probability that there is at least one path available from s to t, assuming that the availability of the edges are mutually independent. Fig. 1(left) shows the example of a transportation graph.

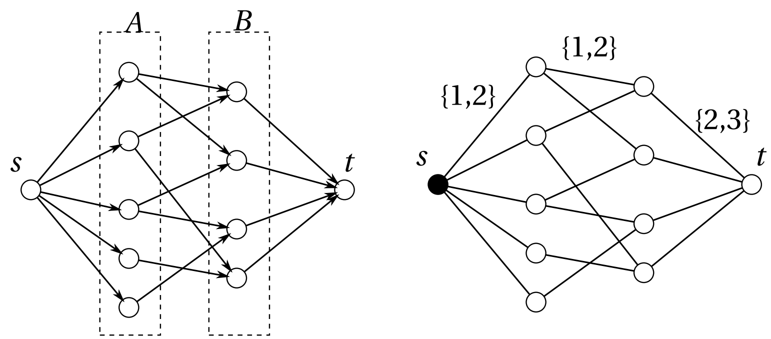

図 1 The example of a transportation graph (left) and its corresponding network that is used in the proof of Lemma 1 (right).

Here we have the following lemma.

Lemma1 If there is an oracle A_b that computes the broadcast time in undirected networks, then we can solve the reliability problem by using A_b.

Proof Given a transportation graph G_t, we consider G_t's underlying undirected graph G_u. We consider two kinds of transmission time for the edges in G_u: Each of the edges between vertices in B and t is associated with a mutually independent random variable that can take 2 or 3; the other edges are each associated with a mutually independent random variables that can take 1 or 2. Each edge takes the smaller value with probability that the corresponding edge is available in G_t. Note that t is the farthest vertex from s in any realization of the transmission times.

Now we can see that the broadcast time is 4 if and only if there is a path from s to t in G_t. Therefore, we can solve the reliability problem by computing the probability that the broadcast in G_u starting from s finishes in time 4. □

We can prove the following theorem.

Theorem1 Computing the distribution function $F_B(x)$ of the broadcast time is $\#P$-complete if the transmission time obeys the discrete distribution.

Proof *(sketch)* The problem can be solved as a counting problem by slightly changing the transformation that Hagstrom used to prove Theorem 1 in[4]. Since the problem is

Vol.2011-AL-134 No.16
2011/3/7

$\#P$-hard by Lemma 1, this theorem is proved. □

4. Algorithm for Continuously Distributed Transmission Times

4.1 Description of Algorithm BC-PDF

Here we show an algorithm for computing the broadcast time distribution function $F_B(x)$. The idea of our algorithm is computing the broadcast time distribution functions of the subgraph S of given graph G. We start from the source of the broadcast and we repeat computing the distribution function of the broadcast time in S and replacing S by new subgraph S' that is obtained by adding one edge or one vertex. When we have S is equal to G, we have the broadcast time distribution function $F_B(x)$. This procedure finishes in polynomial time if each repetition of computing the broadcast time distribution function in S can be completed in polynomial time.

Before proceeding to our algorithm, we need some definitions. Given an undirected graph $G = (V, E)$ with vertex set $V = \{v_0, \ldots, v_{n-1}\}$ and edge set $E \subseteq \{\{u, v\} \mid u, v \in V\}$. We consider a graph $G_J = (V_J, E_J)$ with vertex set $V_J = V \cup \{v_e \mid e \in E\}$ and edge set $E_J = J_1 \cup J_2$, where $J_1 = \{\{v, v_e\} \mid v \in V, e \in E\}$ and $J_2 = \{\{u_e, v_e\} \mid u, v \in V, e \in E\}$: We add two more vertices u_e and v_e for each edge $e = \{u, v\}$ in the original edge set E, then we replace the edge e by a path $\{u, u_e\}, \{u_e, v_e\}, \{v_e, v\}$. We assign each edge in J_1 a static transmission time 0. We assign each edge $\{u_e, v_e\} \in J_2$ the transmission time of $\{u, v\} \in E$. We call the vertices $v_e \in V_J \setminus V$ as the *joint vertices*. Fig.2 shows an example of G_J and $\mathcal{G}$ generated from a graph G.

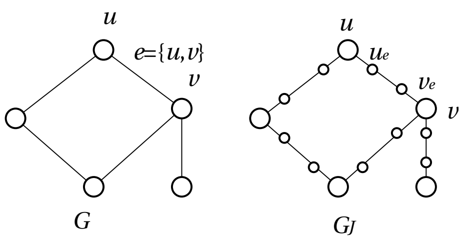

図 2 An example of a graph G (left) and G_J (right). The joint vertices are shown as smaller circles.

We proceed to the description of our algorithm BC-PDF. Let us denote by x_v the time when the broadcast message reaches the vertex $v \in V_J$, *the broadcast arrival time* at v. We denote by X the correspondence of each joint vertices v and a variable x_v for all $v \in V_J \setminus V$. We can define X as the set of pairs $X = \{(v, x_v) \mid v \in V_J \setminus V\}$. Then we have the following algorithm for analytically computing the broadcast time distribution function $F_B(x)$, the probability that the broadcast finishes before time x. We start with computing the broadcast time distribution function $B(S, X, x)$ of a subgraph S of G_J that has vertex set V_J and edge set J_1.

Algorithm BC-PDF(G)

1. Construct G_J from G;
2. Compute $B(S, X, x)$; (Explicit form of initial $B(S, X, x)$ is in Lemma 2.)
3. For each $e \in J_2$ in BFS order from the broadcast source v_0, do
4. Let S' be a graph that is obtained by adding e to S;
5. Let X' be obtained by removing the entry including x_{v_e} and x_{u_e} from X;
6. Compute $B'(S', X', x) = P_1(S, X, x; e) + P_2(S, X, x; e) + P_3(S, X, x; e)$ where $P_1(S, X, x; e)$, $P_2(S, X, x; e)$ and $P_3(S, X, x; e)$ are given by

$$P_1(S, X, x; e) = \iint_{\mathbb{R}^2} \left(\frac{d}{dx_{v_e}} B(S, X, x) \right) f_e(x_{u_e} - x_{v_e}) dx_{u_e} dx_{v_e}, \quad (9)$$

$$P_2(S, X, x; e) = \iint_{\mathbb{R}^2} \left(\frac{d}{dx_{u_e}} B(S, X, x) \right) f_e(x_{v_e} - x_{u_e}) dx_{v_e} dx_{u_e}, \quad (10)$$

$$P_3(S, X, x; e) = \iint_{\mathbb{R}^2} \left(\frac{d}{dx_{u_e}} \frac{d}{dx_{v_e}} B(S, X, x) \right) (1 - F_e(|x_{v_e} - x_{u_e}|)) dx_{u_e} dx_{v_e}, \quad (11)$$

where u_e and v_e are joint vertices in S, x_{u_e} and x_{v_e} are the broadcast arrival time at u_e and v_e respectively;
7. Update S, X and $B(S, X, x)$ by S', X' and $B'(S', X', x)$ respectively;
8. Done;
9. Output $B(S, X, x)$ as $F_B(x)$, where $S = G_J$ and $X = \emptyset$ at this point.

At Step 2., the order of the edges has to be the breadth first search (BFS) order from the source. We use this BFS order to bound the running time.

Here we note how we compute the integrals at step 6.

Let $A_i(S, X_i, x)$ be the broadcast distribution function of the connected component D_i of S in the i-th execution of the loop of algorithm BC-PDF, where X_i is the set of the broadcast arrival times at the joint vertices of D_i. We expand $A_i(S, X_i, x)$ in the integrand into a sum of products when we compute P_1, P_2 and P_3.

To describe the computation concerned with the step functions in the integral, we need some more notations. When $A_i(S, X_i, x)$ in the integrand is expanded into a sum of products at step 6, all terms in the expanded integrand are in the form of the product of a constant C, some elementary functions, at most one delta function and one or more step functions each of which has two variables in the argument. Since the term is 0 if one of the step function factor does not have the positive argument, the upper limit and the lower limit of the definite integral are given for each term by the step function factors that have the dummy function in the argument. Suppose that x_e is the dummy variable that we are going to execute an integral with respect to. Let $\mathcal{H}^+(X_i)$ (resp. $\mathcal{H}^-(X_i)$) be the product of step function factors of a term in the expanded integrand, where the coefficient of x_e in the argument is positive (resp. negative). Let X_i^+ and X_i^- be the disjoint subset of X_i. We have that $\mathcal{H}^+(X_i) = \prod_{(v_e,w_e)\in X_i^+} H(x_e - w_e)$, where w_e is a variable that gives the broadcast arrival time of a joint vertex v_e. Also we have $\mathcal{H}^-(X_i) = \prod_{(v_e,w_e)\in X_i^-} H(w_e - x_e)$.

Now we are ready to describe how we execute the integral in case the integrand consists of some step functions. Since $\mathcal{H}^+(X_i)$ is nonzero if and only if $x_e > \max_{(v_e,w_e)\in X_i^+}\{w_e\}$, a part of the resulting form of executing the definite integral is obtained by replacing x_e in the antiderivative of the term by $\max_{(v_e,w_e)\in X_i^+}\{w_e\}$. Suppose that there are ℓ variables w_e such that $(v_e, w_e) \in X_i^+$ for joint vertices v_e. To remove max operation, we make, from a resulting form's term, $\ell!$ terms for the all permutations of w_e's such that $(v_e, w_e) \in X_i^+$. Note that we put a condition that $w_1, w_2, \ldots, w_\ell$ are ordered $w_1 > w_2 >, \ldots, w_\ell$ otherwise the term is 0, by multiplying $\ell - 1$ step functions $H(w_1 - w_2)H(w_2 - w_3)\cdots H(w_{\ell-1} - w_\ell)$. To complete executing the definite integral, we symmetrically consider $\mathcal{H}^-(X_i)$, which creates min operations. The min operations, again, are removed by making at most $k!$ terms from the resulting form of a term.

For example, if a term consists of some step function factors $H(x_e - w_1)H(x_e - w_2)H(w_3 - x_e)H(w_4 - x_e)$, we have $\mathcal{H}^+(X_i) = H(x_e - w_1)H(x_e - w_2)$ and $\mathcal{H}^-(X_i) = H(w_3 - x_e)H(w_4 - x_e)$. The lower limit of the definite integral of the term is $\max\{w_1, w_2\}$; the upper limit is the $\min\{w_3, w_4\}$. Then we consider four terms of the resulting form: Two lower limit originating terms that is obtained by replacing x_e in the antiderivative by w_1 and w_2, and two upper limit originating terms that is obtained by replacing x_e by w_3 or w_4. Since replacing $\max\{w_1, w_2\}$ by w_1 must be conditioned by $w_1 > w_2$, the first lower limit originating terms is multiplied by $H(w_1 - w_2)$. The other terms are multiplied by $H(w_2 - w_1)$, $H(w_4 - w_3)$ and $H(w_3 - w_4)$, respectively.

Here we see the definition of $B(S, X, x)$ at step 2.

Lemma2 Suppose that $S_v = (V_v, E_v)$ is given by vertex set V_v and edge set $E_v = J_1$. The broadcast time distribution function $B(S, X, x)$ of S is given by,

$$B(S,X,x) = \prod_{v\in V}\sum_{v'_e\in V_v} H(x - x_{v'_e}) \prod_{v_e\in V_v, v_e\neq v'_e} (H(x_e - x_{v'_e})H(x - x_{v_e})), \qquad (12)$$

where we put $x_{v_e} = 0$ for all V_{v_0} if v is the broadcast source v_0.

Before showing the proof of Lemma 2, we explain the reason of defining $B(S, X, x)$ as in Lemma 2. Let C_i be a connected component of initial S; C_i includes the vertex $v_i \in V$. If the edges in J_1 had sufficiently small (but positive) transmission time, C_i ($i = 1, \ldots, n-1$) may have exactly one joint vertex v'_e that receive the broadcast message from the outside of C_i via an edge $e \in J_2$ that will be added to S in the later execution of the loop of step 3–8. The distribution function of C_i can be given as the probability that, on condition that v'_e receives the broadcast message via $e \in J_2$ at time $x_{v'_e}$, the broadcast in C_i finishes before time x, and all vertices but u_e receive the broadcast message from v_i before the times that are given by the corresponding variables in X respectively. Then, $B(S, X, x)$ at step 2 is the sum of the broadcast distribution functions of all possible scenarios in terms of the input joint vertex v of C_i for all $i = 1, \ldots, n-1$. In the above arguments, we consider that they have the distribution function $H(x)$ since the transmission time of edges in J_1 is sufficiently close to 0.

Proof At the beginning of algorithm BC-PDF, every connected component C_i of S is a star graph that has vertex set $V_{v_i} = \{v_i\} \cup \{v_e \mid \{v_i, v_e\} \in J_1\}$ and edge set $E_{v_i} = \{\{v_i, v_e\} \in J_1\}$. Suppose that $v'_e \in E_{v_i}$ is the joint vertex that receives the

Vol.2011-AL-134 No.16
2011/3/7

broadcast message from the outside of C_i. Let us consider the input side. The distribution function of the transmission time of edge $\{v'_e, v\}$ is given by $H(x_v - v'_e)$; the probability that the broadcast time is less than x and the broadcast arrival time at x_{v_i} is in the small interval $[x_{v_i}, x_{v_i} + dx_{v_i}]$ is

$$H(x - x_{v_i})\delta(x_{v_i} - x_{v'_e})dx_{v_i}. \tag{13}$$

As for the output side, the probability that the broadcast time is less than x and the other joint vertices x_{v_e} receive the broadcast message from v_i before time x_{v_e} for all v_e s.t. $\{v_i, v_e\} \in J_1$, on condition that the broadcast arrival time of v_i is x_{v_i}, is

$$\prod_{v_e \in V_{v_i} \setminus \{v_i\}} H(x_{v_e} - x_{v_i})H(x - x_{v_i}). \tag{14}$$

Then we aggregate all possible value of x_{v_i} by integrating the product of (13) and (14), we get

$$H(x - x_{v'_e}) \prod_{v_e \in V_v, v_e \neq v'_e} H(x_{v_e} - x_{v'_e})H(x - x_{v_e}). \tag{15}$$

$B(S, X, x)$ is the sum of the possibilities for all combinations of all $v_i \in V$ for $(i = 1, \ldots, n-1)$ and $v'_e \in V_v$, which can be factorized into (12). □

Inside the loop of BC-PDF is adding an edge to S and computing the corresponding broadcast time distribution function. We have the following lemma.

Lemma3 Let S' be the graph that is given by adding an edge $\{u_e, v_e\}$ to S. Then, the broadcast time distribution function $B'(S', X', x)$ is given by

$$B(S', X', x) = P_1(S, X, x; e) + P_2(S, X, x; e) + P_3(S, X, x; e), \tag{16}$$

where $P_1(S, X, x; e), P_2(S, X, x; e)$ and $P_3(S, X, x; e)$ are given by (9, 10, 11).

Proof We show that $P_1(S, X, x; e)$ is the probability that the broadcast message first reaches joint vertex v_e and then the message is transmitted to u_e via edge $\{u_e, v_e\}$. The probability that the time when the broadcast message reaches v_e from $v \in V$ is in a small interval $[x_{v_e}, x_{v_e} + dx_{v_e}]$ is given by

$$\frac{d}{dx_{v_e}} B(S, X, x)dx_{v_e}. \tag{17}$$

Then the probability that the time when the broadcast message reaches u_e via edge $\{v_e, u_e\}$ is in a small interval $[x_{u_e}, x_{u_e} + dx_{u_e}]$ is given by

$$\frac{d}{dx_{u_e}} B(S, X, x))f_e(x_{u_e} - x_{v_e})dx_{u_e}dx_{v_e}. \tag{18}$$

By integrating with respect to x_{u_e} and x_{v_e} for $x_{u_e} > 0$ and $x_{v_e} > 0$, we have $P_1(S, X, x; e)$ as the probability that u_e receives the the broadcast message from v_e via edge $\{u_e, v_e\}$ before time x. Symmetrically, we have that $P_2(S, X, x; e)$ is the probability that v_e receives the broadcast message from u_e via edge $\{u_e, v_e\}$ before time x.

Now it remains to show that $P_3(S, X, x; e)$ is the probability that u_e and v_e receive the broadcast message from $u, v \in V$, respectively. The probability that the time when the broadcast message reaches v_e is in a small interval $[x_{v_e}, x_{v_e} + dx_{v_e}]$ is given by (17). Then, the probability that the time when the broadcast message reaches u_e from $u \in V$ is in a small interval $[x_{u_e}, x_{u_e} + dx_{u_e}]$ is

$$\left(\frac{d}{dx_{v_e}} \frac{d}{dx_{u_e}} B(S, X, x)\right) dx_{u_e}dx_{v_e}. \tag{19}$$

Since this case does not occur when the transmission time of $\{u_e, v_e\}$ is less than $|x_{v_e} - x_{u_e}|$, we have

$$\left(\frac{d}{dx_{v_e}} \frac{d}{dx_{u_e}} B(S, X, x)\right) (1 - F_e(|x_{v_e} - x_{u_e}|))dx_{u_e}dx_{v_e}. \tag{20}$$

By integrating with respect to x_{u_e} and x_{v_e} for $x_{u_e} > 0$ and $x_{v_e} > 0$, we have $P_3(S, X, x; e)$ as the probability that the both joint vertices u_e and v_e receive the broadcast message from u and v, respectively. □

By using Lemma 3, it is clear that we have the broadcast time distribution function $F_B(x)$ after adding all edges in J_2 to S.

Theorem2 The output of algorithm BC-PDF is the broadcast time distribution function of a network whose topology is given by undirected graph G.

4.2 Running Time in Case Transmission Times Obey the Exponential Distribution with Variance 1

We say that a random variable X_e obeys the exponential distribution with variance 1 and mean $c + 1$ if the distribution function of X_e is given by $P(X_e \leq x) = H(x - r)(1 - \exp(-x + c)))$. In this paper we need to have $c \geq 0$ so that there can be no negative transmission time. We have the following theorem.

Theorem3 Consider a network whose topology is given by an undirected graph G whose maximum minimal cut size is bounded by a constant k. Let the transmission time of each communication links be mutually independent and obey the exponential distribution with variance 1. Then the broadcast time distribution function $F_B(x)$ in the network can be analytically computed in time $O(k^4m^22^{4(k+1)^2}(4m+1)^{2k})$ by algorithm BC-PDF(G).

Proof We prove the running time by bounding the number of terms in the description of $B(S,X,x)$.

Let us define some necessary symbols and words. We assume that the edges $e_1,e_2,\dots,e_m \in E$ is ordered in the breadth first search order from the broadcast source v_0. Let $A_i(S,X_i,x)$ be the broadcast distribution function of the connected component C_i of S in the ith execution of the loop of algorithm BC-PDF, where X_i is the set of the broadcast arrival times at the joint vertices of C_i. Remember that we expand $A_i(S,X_i,x)$ into a sum of the products after computing the integrals at step 6. The factors of each term in $A_i(S,X_i,x)$ can be separated into the step functions and the others; we call the earlier *the step function part* and the latter *the elementary function part*. In the following we bound the number of possible step function parts and elementary function parts.

We prove that the running time of processing the step function part is $O(k^32^{4(k+1)^2})$.

Let us first see that the coefficients of the variables in the step functions' arguments may be 1, -1 and 0. At the beginning, all the coefficients of the variables that appears in the step functions of $B(S,X,x)$ are one of 1, -1 or 0. Then, it is easy to see that neither multiplying the step functions nor differentiating the step functions does not make any change to none of the coefficients of the variables in the step functions' arguments. Then, since executing an integral of a term with respect to a variable x_v in BC-PDF causes only the replacement of x_v by another single variable, we can see that execution of the integral does make coefficients of the variables other than 1, -1 or 0. Therefore, at any point of the loop of the BC-PDF, the variables in the arguments of step functions are one of 1, -1 or 0.

We next prove that there are exactly two variables in the step functions' arguments. It is clear that there are exactly two variables in the arguments of every step functions of $B(S,X,x)$ at the beginning of BC-PDF. Then, again, neither multiplying nor differentiating the step functions does not change the number of the variables in the step functions' arguments. Let us assume that there all step functions' arguments in $A_i(S,X_i,x)$ have two variables each at a i-th execution of the loop of BC-PDF. Suppose that we are going to integrate a term with respect to a variable x_v. Since the step function part of a term may define the upper limit or the lower limit of the definite integral of x_v, executing an integral replaces x_v by another single variable, which means that we still have that all arguments consists of exactly two variable at the $(i+1)$-th execution of the loop of BC-PDF. Therefore, there are exactly two variables in any step function's argument.

Now we are ready to see that there are at most $2^{4(k+1)^2}$ step function part. Since there can be at most $k+1$ variables (k joint vertices and x) in a step function, the above arguments amount to that there are at most $(2(k+1))^2$ step function factors. Then the number of possible combinations of the step function factors is bounded by $2^{4(k+1)^2}$, which amounts to that we need $O(k^32^{4(k+1)^2})$ running time for processing the step function part.

We proceed to bounding the number of elementary function part of $A_i(S,X_i,x)$. Let S have ℓ joints at the i-th execution of the loop of BC-PDF. Since the transmission time distribution is the exponential distribution with variance 1, the elementary function part of the terms of $A_i(S,X_i,x)$ can be expanded into a sum of the following form

$$Cx_1^{a_1}x_2^{a_2}\dots x_\ell^{a_\ell}\exp(b_1x_1+b_2x_2+\dots b_\ell x_\ell+c'), \quad (21)$$

where C,c' are constants, $x_1,x_2,\dots,x_\ell$ are the broadcast arrival times at ℓ joints, $a_1,\dots,a_\ell$ and $b_1,\dots,b_\ell$ are integers. Since a_i and $|b_i|$ $(i=1,\dots,\ell)$ can increase by at most 1 in executing an integral, a_i and $|b_i|$ for any $i=1,\dots,\ell$ are at most twice the number of edges in S. It amounts to that there are at most $(2m)^\ell(4m+1)^\ell = O((4m+1)^{2k})$ elementary function parts.

It remains to show the running time per one term. When we execute an integrate a term in the form of (21) with respect to x_1, we compute the term by partial integration. Then, the partial integration generates a_1 terms from the term. Since there are $O(\ell)=O(k)$ factors in one term and there are $a_1=O(m)$ resulting form, we need $O(km)$ running time per one term.

Now multiplying the number of possible step function parts and the number of possible elementary function part proves the theorem. □

Then we have the following corollary.

Corollary1 If the network topology's maximum minimal cut is bounded by a constant k, the edge weights are mutually independent and obey the exponential distribution with variance 1, BC-PDF computes the broadcast time distribution function $F_B(x)$ in polynomial time in the network size.

5. Conclusion

In this paper, we considered the broadcast time in computer networks in which the transmission time of the communication links are given as mutually independent random variables. We proved that the problem of computing the distribution function of the broadcast time is $\#P$-complete. We showed an algorithm BC-PDF that computes the distribution function of the broadcast time in a network. When BC-PDF is applied to the network whose topology has maximum minimal cut with size at most a constant k, and the transmission times are mutually independent and obey the exponential distribution with variance 1, BC-PDF finishes in the polynomial time in the size of the network. For future work, we think that there may be an algorithm that computes a polynomial that approximates $F_B(x)$ with some reasonable conditions.

参 考 文 献

1) E. Ando, T. Nakata, M. Yamashita, Approximating the longest path length of a stochastic DAG by a normal distribution in linear time, *Journal of Discrete Algorithms*, 7 pp. 420–438 (2009)
2) E. Ando, H. Ono, K. Sadakane, M. Yamashita, Computing the Exact Distribution Function of the Stochastic Longest Path Length in a DAG, Proceedings of the 6th conference on Theory and Application of the Models of Computation (TAMC2009), LNCS, 5532, pp. 98–107 (2009)
3) E. Ando, H. Ono, K. Sadakane, M. Yamashita, A Generic Algorithm for Approximately Solving Stochastic Graph Optimization Problems, Proceedings of the 5th Symposium on Stochastic Algorithms, Foundations and Applications (SAGA 2009), LNCS, 5792, pp. 89–103, 2009
4) J. N. Hagstrom, Computational Complexity of PERT Problems, *NETWORKS*, Vol. 18, 1988, pp. 139–147.
5) J. S. Provan and M. O. Ball, The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM J. Comput*, Vol. 12, 1983, pp. 777–788.