韓國軍事運營分析學會誌
第18卷, 第2號, 1992. 12. 31.

# Analysis of Packet Transmission Probability under Flooding Routing

Jung Sik Hong*
Chang Hoon Lie**
Hae Sang Lee**

## Abstract

In this paper, the computational problems of packet transmission probability (PTP) in a computer communication network(CCN) under flooding routing are investigated. To avoid a congestion under this routing, two control methods are considerd, i.e., copy storage control and hop count control. Problems of PTP under flooding routings with these two control methods are respectively shown to be equivalent to those of source-to-terminal reliability(STR) with an exception for a case of hop count control where the hop count is less than the length of the longest path. For this exceptional case, an efficient computational algorithm for PTP is developed. This algorithm is proposed as an efficient tool for the determination of hop count which satisfies a given reliability constraint. A numerical example illustrates a proposed algorithm.

* Seoul National Polytechnic University
** Seoul National University

# 1. Introduction

Routing is a decision-making process in which a given node in a packet-switched computer communication network (CCN) selects one or more of its outgoing lines on which to forward a packet, making its way to an ultimate destination[4]. Flooding is one of the simplest routing techniques, which requires no network information (nonadaptive routing). Under this technique, an incoming packet in a specific node is transmitted to all adjacent nodes except the one from which it was received. In this way, this technique ensures us to have high packet transmission probability (PTP) and, hence, is used to send high priority messages. However, since a network is quickly loaded with an infinite number of packets, heavy traffic load is easily resulted, which is called "congestion". The following two control methods [8] are proposed to avoid such a congestion :

1. At each node, a list of packets that have already passed through is maintained. When a packet passes through this specific node for the second time, it is discarded (called as flooding with copy storage control)..
2. A hop count is included in the header of each packet and is decremented by one as a packet is transmitted. When the count reaches zero, the packet is discarded (called as flooding with hop count control).

In this paper, we investigate the computation problems of packet transmission probability (PTP) which is the probability that a packet is successfully transmmitted from source to destination in a CCN under flooding routing with two congestion controls.

In specific, we show that the problems of PTP under flooding routing with copy storage control and hop count control where the hop count is not less than the length of the longest path are respectively equivalent to those of source-to-terminal reliability (STR).

When the hop count is less than the length of the longest path in a flooding routing with hop count control, an efficient computational algorithm for PTP is developed, which utilizes a factoring algorithm as a subroutine. The proposed algorithm can efficiently be used to determine the hop count which satisfies a given reliability constraint. A numerical example illustrates a proposed algorithm.

Notation

V : set of vertices denoted as $\{v_1, \ldots, v_m\}$

E : set of edges denoted as $\{e_1, \ldots, e_n\}$

G(V, E) : an undirected graph G with vertex set V and edge set E

$p_i$ : the probability that ei ∈ E is working

$q_i$ : $1 - p_i$

G∗i : G with $e_i$ contracted

G-i : G with $e_i$ deleted

N=〔G(V, E), {s, t}, $p_i$〕: network N with edge reliability pi and source and terminal are attached to the graph G

PTP(G) : packet transmission probability in an undirected network N with copy storage control

$PTP^H$(G) : packet transmission probability in an undirected network N with hop count control where initial hop count is set to H

L(G) : length of the longest path in G, that is, number of edges in the longest path

S(G) : length of the shortest path in G, that is, number of edges in the shortest path

R(G) : STR, that is, a probability that source and terminal are connected by working edges

# 2. Formulation of Problem of PTP Under Flooding Routing

A packet-switched CCN with link failures is typically modeled as a probabilistic graph. Bidirectional links and switching nodes of CCN respectively correspond to undirected edges and vertices of the graph. In this setting, PTP of CCN is the probability that packets are successfully transmitted from source to destination by working edges. Edge failures are assumed to be s-independent. So, problems of PTP under two congestion controls are expressed as follows :

1. Problem of PTP under flooding routing with copy storage control〔PC〕;

   Input : N = 〔 G(V, E), {s, t}, pi 〕

   Output : PTP (G)

2. Problem of PTP under flooding routing with hop count control〔PH〕;

   Input : N = 〔 G(V, E), {s, t}, $p_i$ , H 〕

   Output : $PTP^H$ (G)

# 3. Analysis of PTP Under Flooding Routing

## 3.1 Relationship Between PTP and STR

As all routes between source and destination are used in flooding routing under two congestion controls (except for the hop count control where the initial hop count is less than the length of the longest path between source and destination ), the problem of PTP seems to correspond to that of STR. Computation of STR is known to be NP-hard〔8〕. However for some specifically structured networks, the linear time algorithm is developed〔7〕, known as a factoring algorithm. In this respect, once a relationship between the problem of PTP and that of STR could be established, the problem of PTP can be solved efficiently by using the algorithms developed for STR.

The following proposition shows the relationship between the problem of PTP under flooding routing with two congestion controls and that of STR.

Proposition 1.

(a) Problem of STR is equivalent to 〔PC〕.

(b) Problem of STR is equivalent to 〔PH〕 when $H \geq L(G)$.

Proof

(a) To prove the equivalence, it is enough to show that the success state in problem of STR implies that of 〔PC〕 and the failure state in problem of STR implies that of 〔PC〕, vice versa.

Consider an arbitrary state of a network system composed of working edges and failed edges. Then, this system state will indicate either of the following two cases;

(i) the success state in STR problem

(ii) the failed state in STR problem

Suppose case (i) is true. Then, there exists at least one connection path between source and terminal by working edge. In [PC], therefore, a packet has at least one outgoing edges. So, a packet is successfully transmitted to destination. Conversely, suppose a packet is successfully transmitted to destination in [PC]. This implies that there exists at least one connection path between source and terminal by working edges. So, it corresponds to case (i).

Suppose case (ii) is true. Then, there exists at least one cut set $(X, \overline{X})$ between source and terminal. In [PC], therefore, a packet transmitted from source is circulated only in vertices containing X and is not transmitted to destination.

Conversely, suppose a packet is not transmitted to destination. All vertices are divided into two classes $(X, \overline{X})$, where X is a set of vertices which contain a packet and $\overline{X}$ is a set of vertices which do not contain a packet.

Obviously, a set of edges $(\overline{X}, X)$ indicates failure. So, it corresponds to case (ii).

(b) When $H \geq L(G)$, all routes between source and destination are used. Therefore, proof is similar to that of (a).

## 3.2 Computational Algorithm of [PH] when $H < L(G)$

In case of $H < L(G)$ in [PH], some routes (paths) from source to terminal are not used in the process of packet transmission under flooding routing. The value of PTP is, therefore, less than that of STR. This problem is related to that of STR in the network with delay constraints, which is also called path-length feasibility problem[6]. An algorithm in [8] is based on sum-of-disjoint products.

In this paper, an efficient computational algorithm of [PH] when $H < L(G)$ is presented. The proposed algorithm utilizes a factoring algorithm which is based on the following factoring theorem of network reliability.

$$R(G) = p_i \; R(G*i) + (1-p_i)R(G-i) \qquad (1)$$

A packet transmission process in a CCN under flooding routing with hop count control proceeds as a source transmits packets to all outgoing links incident to source. Then, the remaining hop count of arrived packets becomes H-1. If the number of links incident to source is a, then the number of all possible states caused by link failures is 2a. So, in [PH], (1) is extended as

$$PTP^{H}(G) = \sum_{i=1}^{2a} P(F_i)PTP^{H-1}(G/F_i)$$

$$\sum_{i=1}^{2a} P(F_i)PTP^{H-1}(G^{1,i}) \qquad (2)$$

where Fi denotes the state of links {1, 2, ..., a} due to s-independent link failures

$G^{1,i}$ denotes the induced subgraph by deleting the failed edges and contracting the working edges in G

Note that the network diameter of $G^{1,i}$, $L(G^{1,i})$, is less than or equal to L(G)-1. In order to compute $PTP^{H-1}(G^{1,i})$, the following three check criteria are used :

1. If the induced subgraph $G^{1,i}$ is disconnected, then, obviously

$$PTP^{H-1}(G^{1,i}) = 0$$

2. If H-1 < $S(G^{1,i})$, then all paths from source to destination cannot be used. Hence,

$$PTP^{H-1}(G^{1,i}) = 0.$$

3. If H-1 ≥ $L(G^{1,i})$, then all paths from source to destination are used. Hence, from proposition 1,

$$PTP^{H-1}(G^{1,i}) = STR(G^{1,i}) .$$

If $G^{1,i}$ does not satisfy the above three criteria, i.e. it is connected and $S(G^{1,i}) \le H-1 < L(G^{1,i})$, the above recursive process is repeated for $G^{1,i}$ .

So, at step i, the induced subgraph, $G^{i,j}$, is in one of the following 4 classes :

1. $G^{i,j}$ is contained in class 1 if $G^{i,j}$ is disconnected

2. $G^{i,j}$ is contained in class 2 if $G^{i,j}$ is connected and H-i $<$ S $(G^{i,j})$

3. $G^{i,j}$ is contained in class 3 if $G^{i,j}$ is connected and H-i $\geq$ L $(G^{i,j})$

4. $G^{i,j}$ is contained in class 4 if Gi, j is connected and $S(G^{i,j}) \leq$ H-i $<$ $L(G^{i,j})$

As was seen for $G^{1,i}$ , $PTP^{H-i}(G^{i,j})$ = 0 for classes 1 and 2, $PTP^{H-i}(G^{i,j})$ = $STR(G^{i,j})$ for class 3 and the recursive process is repeated for $G^{i,j}$ for class 4.

Now, a computational algorithm for PTP in [PH] is as follows, where REL(G) denotes that of STR in network G(V,E) :

Function $PTP^H(G)$

Input N = [ G(V,E), {s,t}, $p_i$ , H ]

Output $PTP^H(G)$

Begin

If G is disconnected, then return(0).

If s=t, then return(1).

Do parallel reduction.

Compute S(G) and L(G).

If H $<$ S(G), then return(0).

If H $\geq$ L(G), then go to REL(G).

Otherwise, Select all edges incident to source.

These edges are denoted as {1, . . . , a}.

then PTP $\leftarrow \sum_{i=1}^{2a} P(F_i) PTP^{H-1}(G^{1,i})$

End

Parallel reduction is used in the algorithm because parallel reductions do not alter the length of all paths.

# 4. Numerical Example

Consider an example network in figure 1. The longest and the shortest path lengths of the graph are 5 and 3 respectively. $PTP^5(G)$ is, therefore, equal to STR in this example. To illustrate the iteration processes of an algorithm, we take H= 4 and obtain $PTP^4(G)$. At first, parallel reduction is carried out on edges {3, 4} and multi-pivoting is applied on edges {1, 2} which are incident to S. As a result, four subgraphs are obtained, whose related informations are illustrated in table 1.

More specifically, $PTP^4(G)$ is expressed as

$$PTP^4(G) = {}_{p1p2}PTP^3(G^{1,1}) + {}_{p1q2}PTP^3(G^{1,2}) + {}_{q1q2}PTP^3(G^{1,3}) .$$

Calculation of $PTP^3(G^{1,1})$ is efficiently carried out using the relationship that $PTP^3(G^{1,1}) = STR(G^{1,1})$ . Calculation processes are repeated for both $G^{1,2}$ and $G^{1,3}$ . Table 2 shows these processes for $G^{1,2}$ . Following these procedures, values of $PTP^4(G)$ for various p are obtained. Table3 illustrates values of $PTP^H(G)$ for various p and H=3, 4, 5. These are graphically illustrated in figure 2.

From a theoretical viewpoint, when H is close to S(G), $PTP^H(G)$ is easily obtained by generating feasible paths and using either the inclusion-exclusion formula or sum-of-disjoint product formula. In practice, however, H is taken as sufficiently large for the robustness.

# 5. Conclusions

In this paper, the relationship between the computational problem of PTP in a CCN under flooding routing and that of STR is investigated. Problem of PTP under flooding routing with two control methods are respectively shown to be equivalent to those of STR with an exception for a case of hop count control where the hop count is less than the length of the longest path. For this exceptional case, an efficient computational algorithm for PTP, which utilizes a factoring algorithm as a subroutine, is developed. The proposed algorithm is efficient in determining the hop count which satisfies a given reliability constraint.

## REFERENCES

1. J. A. Abraham, "An improved algorithm for network reliability," *IEEE Trans. Reliability*, vol. R-28, 1979, pp. 58-61.

2. M. O. Ball, "Computing network reliability," *Operation Research*, vol 27, 1979, pp. 823-838.

3. M. O. Ball, "Complexity of network reliability computations," *Networks*, vol. 10, 1979, pp. 153-165.
4. P. R. Bell and K. Jabbous, "Review of point-to-point network routing algorithms," *IEEE Comm.* Magazine, vol. 24, 1986, pp. 34-38.
5. M. K. Chang, "A graph theoretic appraisal of the complexity of network reliability algorithms," Ph. D. Thesis, OR Center, Univ. of California, Berkeley, 1981.
6. J. N. Hagstrom, "Directed network reliability : domination and computing coefficients of the success-marginal expansion," *Networks*, vol. 20, 1990, pp. 65-78.
7. J. N. Hagstrom and A. Satyanarayana, "Combinatorial properties of directed graphs useful in computers network reliability," *Networks*, vol. 11, 1981, pp. 357-366.
8. Y. J. Park and H. Tanaka, "Reliability evaluation of a network with delay," *IEEE Trans. Reliability*, vol. R-28, 1979, pp. 320-324.
9. T. Politof and A. Satayanarayana, "Network reliability and inner-four-cycle-free graphs," *Math. of OR*, vol. 1986, pp. 484-505.
10. W. Stalling, *Data and Computer Communications*, Macmillan Publishing Company, 1985.
11. R. K. Wood, "Factoring algorithm for computing k-terminal network reliability," *IEEE Trans. Reliability*, vol. R-35, 1986, pp. 269-278.

TABLE 1.

Induced subgraphs at 1st iteration with related informations

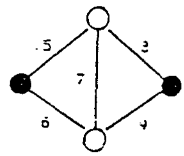
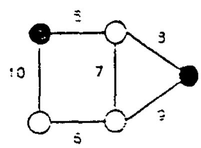
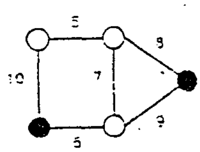
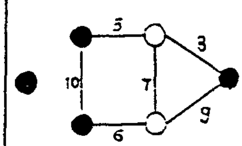
| state of edges | induced subgraphs | informations of subgraphs | classes |
|---|---|---|---|
| $F_1 = \{ 1\ 2 \}$ |  | S(G) = 2<br>H-1 = 3<br>L(G) = 3 | class 3 |
| $F_2 = \{ 1\ \bar{2} \}$ |  | S(G) = 2<br>H-1 = 3<br>L(G) = 4 | class 4 |
| $F_3 = \{ \bar{1}\ 2 \}$ |  | S(G) = 2<br>H-1 = 3<br>L(G) = 4 | class 4 |
| $F_4 = \{ \bar{1}\ \bar{2} \}$ |  | disconnected | class 1 |

TABLE 2.

Induced subgraphs at 2nd iteration with related informations

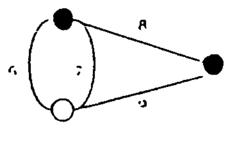
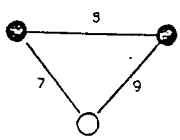
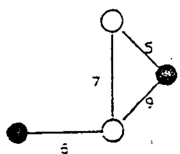
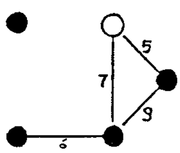
| state of edges | induced subgraphs | informations of subgraphs | classes |
|---|---|---|---|
| $F_1 = \{ 5 \; 10 \}$ |  | S(G) = 1<br>H-1 = 2<br>L(G) = 2 | class 3 |
| $F_2 = \{ 5 \; \overline{10} \}$ |  | S(G) = 2<br>H-1 = 2<br>L(G) = 3 | class 4 |
| $F_3 = \{ \overline{5} \; 10 \}$ |  | S(G) = 1<br>H-1 = 2<br>L(G) = 2 | class 3 |
| $F_4 = \{ \overline{5} \; \overline{10} \}$ |  | disconnected | class 1 |

TABLE 3.

Packet transmission probability for various values of pi = p, i = 1,2,...,9 and H = 3,4,5

| p \ H | $PTP^3(G)$ | $PTP^4(G)$ | $PTP^5(G)$ |
|---|---|---|---|
| 0.1 | 0.001999 | 0.002513 | 0.002540 |
| 0.2 | 0.015936 | 0.022670 | 0.023260 |
| 0.3 | 0.053271 | 0.079919 | 0.082753 |
| 0.4 | 0.123904 | 0.186278 | 0.193356 |
| 0.5 | 0.234375 | 0.339844 | 0.351563 |
| 0.6 | 0.385344 | 0.523695 | 0.537629 |
| 0.7 | 0.568351 | 0.709804 | 0.721602 |
| 0.8 | 0.761856 | 0.866451 | 0.872743 |
| 0.9 | 0.926559 | 0.967224 | 0.968523 |
| 0.91 | 0.939273 | 0.973592 | 0.974584 |
| 0.92 | 0.951021 | 0.979259 | 0.979988 |
| 0.93 | 0.961724 | 0.984227 | 0.984738 |
| 0.94 | 0.971298 | 0.988498 | 0.988835 |
| 0.95 | 0.979658 | 0.992079 | 0.992282 |
| 0.96 | 0.986714 | 0.994976 | 0.995085 |
| 0.97 | 0.992374 | 0.997202 | 0.997250 |
| 0.98 | 0.996542 | 0.998770 | 0.998785 |
| 0.99 | 0.999118 | 0.999696 | 0.999698 |

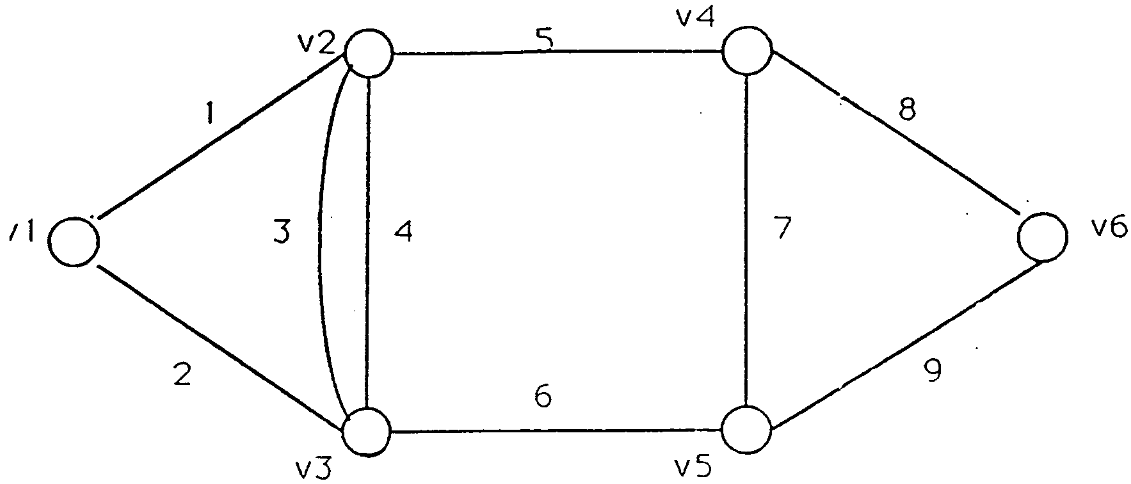



Fig. 1. Example Network

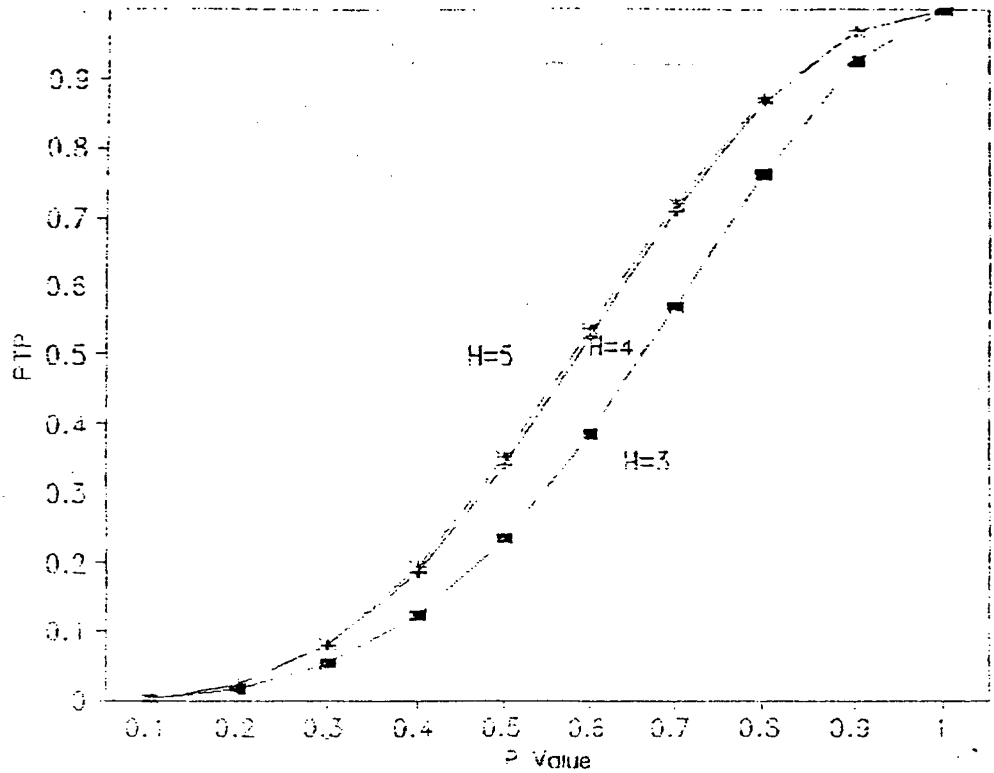



Fig. 2. Packet transmission probability for variovs values of pi=p, i=1, 2, 3 and H=3, 4, 5