CONGESTION CONTROL FOR COMPUTER NETWORKS WITH NODAL BUFFERS MANAGEMENT STRATEGY BASED ON TRAFFIC PRIORITIES

Ruttikorn VARAKULSIRIPUNTH, Norio SHIRATORI, and Shoichi NOGUCHI

(Research Institute of Electrical Communication, Tohoku University, Sendai, 980 Japan)

Abstract

In this paper a distributed congestion control policy, called PF-C, with nodal buffers management strategy based on traffic priorities is proposed and analyzed. The priority of a packet is depended upon the number of hops it has traveled and the number of hops it has to travel to reach its destination. Then, a packet that arrived at a given node is rejected if the number of allocated buffers exceeds a limit value corresponding to its priority.

This policy is analyzed in the context of symmetrical networks. Numerical applications to Loop networks clearly demonstrate the good behavior of this policy to achieve the maximum network throughput. As the results, PF-C scheme shows more effective than another similar policies which have been proposed.

1. INTRODUCTION

In a Store-and-Forward (S/F) computer network, if users' demands (i.e.,offered traffic) are allowed to exceed the system capacity without control, unpleasant congestion effects will occur, and its effective throughput is degraded as shown in Fig.1 (curve 4). Numerous flow control procedures have been proposed in order to prevent network congestion as well as to avoid network throughput degradation[5]. In this paper, we considered our flow control policy with a finite nodal buffers management based on the priorities of packets. Briefly, at first we assign different priorities to all packets which arrived at a node according to both the number of hops they have traveled, say i, and the number of hops from this node to reach their destinations, say j. The priority assignment is based on our main purpose in order to avoid the throughput degradation and also to maximize the throughput, as well. Intuitively, we assign increasing priority to any packet when i becomes larger and together with j becomes smaller. Next we divide all packets into classes in accordance with the rank of their priorities, by defining a packet of priority r as a class-r packet. Recall that the larger r is, the higher priority is indicated. Finally, we assign the limited number of buffers, say L_r, as a limit value to each class,

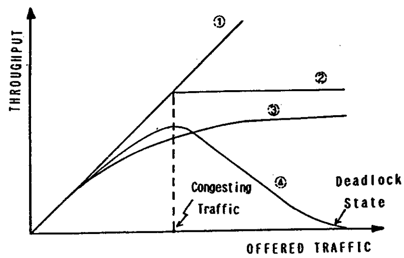

Fig.1 Offered traffic-Throughput relationship, ①:Infinite storage buffers condition, ②. ③. ④:Finite storage buffers condition, (②:ideal flow control, ③:effective flow control, ④:no flow control).

say r, such that $L_0 < L_1 < \cdots < L_m$. Then the concept of our control policy is that a class-r packet is accepted by a node if and only if the number of occupied buffers in this node is less than L_r, otherwise it is rejected. Rejected packets are dropped from the networks and considered as lost. Due to our consideration, we call our control policy as "Past-and-Future-hops-Count (PF-C)" flow control.

A number of similar policies has been proposed and some analyzed[1],[2],[3]. Geissler et al.[1] made a simulation studies of assigning increasing priority to a packet according to the number of hops it has traveled only. Lam and Reiser[2], and Kamoun[3] made a simplified analysis by dividing packets only into two classes, call "Input" and "Transit" packets. Packets that traveled over one or more hops are considered as transit packets and packets that are candidates to enter the communication network are considered as new packets. Their idea is to favor transit packets over input packets.

A similar policy in case of infinite nodal buffers is also proposed by Tezuka and Sanada et al.[4]. Their main purpose is to minimize system delay. However, the analysis of blocking probability suffered by packets in a finite storage system is more difficult than one in an infinite storge system.

In this paper we present and analyze our PF-C scheme for a family of symmetrical networks. A node model and analytic conditions are the same as one introduced by Kamoun[(3),(6)]. A queueing model of a node is developed whereby network throughput is evaluated in terms of offered traffic, the number of buffers in a node and the limit value L_r. It is clear and easy to show that our analytic results of PF-C scheme obtained in section 3 cover both schemes proposed by Giesler et al. and Kamoun.

In section 4, we present a decision protocol for determining the priorities of packets based on PF-C scheme, in accordance with the difference between i and j. The numerical applications to Loop networks show that a group of optimal limit value improves the maximum throughput obtained in a no control environment and succeeds in maintaining a very good throughput behavior under heavy traffic conditions. As the results of Loop network with uniform traffic assumption, the PF-C scheme shows more effective than another policies.

2. PF-C FLOW CONTROL POLICY

PF-C scheme is a mechanism that make a decision to accept or reject a packet upon its arrival at a node, in compliance with the number of occupied buffers in this node. Therefore, we have to decide that, 1) which packet should be accepted or rejected, and 2) in what condition, in order to maximize network throughput.

2.1 Packet Priorities and Classification

Upon arrival of a packet at a given node, let
i be the number of hops it has traveled,
j be the number of hops in the path from this node to its destination node,

then we define $P_f(i,j)$ as a priority function of this packet, where i and j are two positive integer variables ($i \geqslant 0$ and $j \geqslant 0$).

Now, let r_{ij} be the positive integer that denotes the ranking of packet priorities corresponds to the value of $P_f(i,j)$. Then we formulate the priority function in terms of r_{ij} as follow,

$$P_f(i,j) = r_{ij}. \quad (1)$$

And let us assume that there are at most m+1 different priorities rankings in the system. Then $r_{ij} = 0,\ldots,m$ whereby 0 denotes the lowest priority and m denotes the highest priority.

Finally, we define the packet of priority r_{ij} as the class-r_{ij} packet. Hence, there are m+1 classes of packets in the entire system.

2.2 Buffers Management Strategy

Now, let B be the total number of S/F buffers available at a given node. Then define $L_{r_{ij}}$ as a "limit value" (i.e., a limited number of buffers) corresponds to class-r_{ij} packet, such that, $L_0 < L_1 < \cdots < L_m$ and $L_m = B$.

We need to assume that $B \geqslant m+1$ in order to define m+1 different limit values correspond to each class of packets. Then the constraint of PF-C scheme is that, a packet that arrived at a given node and is of class-r_{ij} packet will be accepted by this node if and only if the number of occupied buffers, immediately prior to its arrival, is less than $L_{r_{ij}}$, otherwise it is rejected. Rejected packets are dropped from the networks. Therefore, PF-C algorithm for any packet is defined as follow.

PF-C Algorithm

Upon arrival of a packet belongs to class-r_{ij} at a given node, let n be the number of occupied buffers immediately prior to its arrival.

Then, the decision rule is as follow.

(1) IF $n = B$ reject this packet.
(2) IF $n < L_0$ accept this packet.
(3) IF $L_{k-1} \leqslant n < L_k$ accept this packet only if $r_{ij} \geqslant k$, where k=1,..,m.

3. ANALYSIS OF THE PF-C SCHEME IN A SYMMETRICAL COMPUTER NETWORK ENVIRONMENT

3.1 A Symmetrical Network System

A symmetrical network is one such that

1) all nodes are equivalent with respect to the topology of the network,
2) all channels are of equal capacity, say C (b/s),
3) all external offered traffic rates to each node are equal.

The above properties imply that all nodes are identical and also perform identical functions. All links between nodes are full-duplex lines with capacity C in each direction. Therefore, if R is the number of links attached to a node, then there are R incoming and R outgoing channels at any given node. Let N be the number of nodes, then there are N×R channels in the network. As examples, Loop and Torus networks fall into this category(6).

Let γ_{ab} (packets/sec.) be the average offered external traffic rate from source node a to destination node b, then

$$\gamma_{ab} = \gamma \qquad \forall a,b \text{ network nodes and } a \neq b.$$

We assume that a node does not generate traffic to itself, i.e., $\gamma_{aa} = 0$ for all a.

Let Γ be the total offered external traffic rate, then

$$\Gamma = \sum_{a,b} \gamma_{ab} = N(N-1)\cdot\gamma$$

where N(N-1) is the number of source-destination node pairs, in each direction, in the network.

All nodes and channels are assumed to be perfectly reliable. It is then obvious that with this particular topology structure, the routing decision is assumed to be a shortest path routing policy in order to achieve the optimal flow assignment(6). The selection of the particular shortest paths, in case that more than one exists, must result in perfectly balanced flow.

Let $\lambda^{(s)}_{r_{ij}}$ be the average offered traffic rate of class-r_{ij} packets to channel s, immediately prior to the acceptance or rejection by a node. Then, due to the network symmetry all $\lambda^{(s)}_{r_{ij}}$'s are equal,i.e.,

$$\lambda^{(s)}_{r_{ij}} = \lambda_{r_{ij}} \qquad s=1,\dots,N\times R.$$

Moreover, all nodes contain the same number of S/F buffers B and use the same PF-C algorithm.

As a result, the probabilities of blocking suffered by class-r_{ij} packets are equal at all nodes. And because of the possibility of the loss of packets, the offered external traffic rate Γ is no longer equal to network throughput.

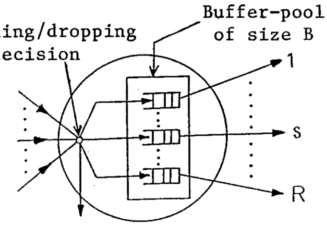

Fig.2 A S/F node model.

3.2 Blocking Probabilities

The model of S/F node is shown in Fig.2. Let s=1,....,R be the index of R outgoing channels. A given channel can only serve packets routed to that channel. We assume that one buffer size is equal to the maximum packet length. Then, let n be the total number of packets stored at a given node and let n_s be the number of packets waiting in the output queue of channel s, we get

$$n = \sum_{s=1}^{R} n_s \leqslant B \tag{2}$$

Moreover, let α_n be the average rate of packets that can access into any outgoing channel of a given node, which depends on n. Then, due to the PF-C algorithm, at the immediately posterior to acceptance or rejection we find

$$\alpha_n = \sum_{r_{ij}=k}^{m} \lambda_{r_{ij}} \qquad \text{for } L_{k-1} \leqslant n < L_k \tag{3}$$

where k=0,1,...,m and let $L_{-1}=0$.

In order to pursue the analysis we make the following classical local assumptions as is introduced by Kleinrock(8).

(1) α_n is a Poisson arrival for all $n=\sum_{s=1}^{R} n_s$.
(2) Distribution of packet lengths is negative exponential and average packet length is denoted by $1/\mu$ (bit).
(3) The independence assumption of Kleinrock is assumed valid for all nodes.

As a result, a node is equivalent to R single M/M/1 queueing systems sharing a waiting room of size B under the PF-C scheme. Finally, the entire system is equivalent to a birth-death process which α_n is a birth rate and μC is a death rate for each outgoing channel. Then the state of the system can be simply described by vector denoted as $(n_1,..,n_s,..,n_R)$.

Now, let Δ_n and δ_s are two Kronecker deltas with respect to n and n_s respectively, such that

$$\Delta_n = \begin{cases} 1 & \text{if } 0 \leqslant n < B \\ 0 & \text{if } n = B \end{cases} \quad (4)$$

$$\delta_s = \begin{cases} 1 & \text{if } 0 < n_s \leqslant B \\ 0 & \text{if } n_s = 0 \end{cases} \quad (5)$$

By using these two Kronecker deltas, we can write the state-transition-rate diagram of a node with respect to state $(n_1,..,n_s,..,n_R)$ as shown in Fig.3.

Let $P(n_1,..,n_s,..,n_R)$ be a state probability in steady state. According to the equilibrium distribution of traffic flow[8], and from the state-transition-rate diagram of Fig.3, we can write the "global" balance equation that describes the behavior of the system in the steady state for all feasible states such that $n=\sum_{s=1}^{R} n_s \leqslant B$ as below.

$$\left[\left\{\sum_{s=1}^{R}\delta_s \mu C\right\} + R\Delta_n \alpha_n\right] P(n_1,..,n_s,..,n_R)$$
$$= \alpha_{n-1}\left[\sum_{s=1}^{R}\delta_s P(n_1,..,n_s-1,..,n_R)\right]$$
$$+\Delta_n \mu C\left[\sum_{s=1}^{R} P(n_1,..,n_s+1,..,n_R)\right] \quad (6)$$

To solve the above global balance equation for a general solution, we apply the technique of local balance equation introduced by Chandy[9]. It is then obvious that from equation (6), we are able to obtain the following two local balance equations.

$$\left\{\sum_{s=1}^{R}\delta_s \mu C\right\} P(n_1,..,n_s,..,n_R)$$
$$= \alpha_{n-1}\left[\sum_{s=1}^{R}\delta_s P(n_1,..,n_s-1,..,n_R)\right] \quad (6a)$$

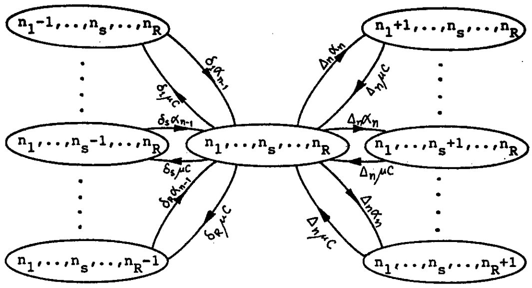

Fig.3 State-transition-rate diagram.

$$R\Delta_n\alpha_n\, P(n_1,..,n_s,..,n_R)$$
$$= \Delta_n \mu C\left[\sum_{s=1}^{R} P(n_1,..,n_s+1,..,n_R)\right] \quad (6b)$$

First, we derive the solution of both equation (6a) and (6b). Finally, it is exactly certified that the solution obtained from (6a) and (6b) is also satisfy the global balance equation (6). That is, after some algebra, the probability of state $(n_1,..,n_s,..,n_R)$ in steady state is as follow,

$$P(n_1,..,n_s,..,n_R) = K\left\{\prod_{x=1}^{n}\alpha_{x-1}\right\} \Big/ (\mu C)^n \quad (7)$$

where $K=P(0,...,0)$ is the probability of an empty system and is a constant. Then by the substitution of α_n for all $n=\sum n_s \leqslant B$ into (7), we obtain the final solution based on PF-C scheme as follow,

$$P(n_1,..,n_s,..,n_R) = \begin{cases} K\rho_0^n & \text{if } 0 \leqslant n \leqslant L_0 \\ K\left\{\prod_{t=0}^{k-1}(\rho_t/\rho_{t+1})^{L_t}\right\}\rho_k^n & \text{if } L_{k-1} < n \leqslant L_k \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where $k=1,...,m$ and $\rho_t = \left\{\sum_{r_{ij}=t}^{m} \lambda_{r_{ij}}\right\} \Big/ \mu C$ for $t=0,1,..,m$.

Moreover, let $Pr(n)$ be the probability denotes that there are n packets waiting for service in a given node in the steady state, then

$$Pr(n) = \sum_{(\sum_{s=1}^{R} n_s)=n} P(n_1,..n_s,..,n_R). \quad (9)$$

As a result, from (8) and (9) we obtain

$$\Pr(n) = \binom{n+R-1}{R-1} P(n_1,..,n_s,..,n_R) \quad (10)$$

and finally,

$$\Pr(n) = \begin{cases} \binom{n+R-1}{R-1} K \rho_0^n & \text{if } 0 \leqslant n \leqslant L_0 \\ \binom{n+R-1}{R-1} K \left\{ \prod_{t=0}^{k-1} (\rho_t/\rho_{t+1})^{L_t} \right\} \rho_k^n & \text{if } L_{k-1} < n \leqslant L_k \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

The constant K can be computed by using the normalization condition such that $\sum_{n=0}^{B} \Pr(n)=1$. Using (11) and after some algebra, we get

$$K^{-1} = \sum_{n=0}^{L_0} \binom{n+R-1}{R-1} \rho_0^n + \sum_{k=1}^{m} \left[\left\{ \prod_{t=0}^{k-1} (\rho_t/\rho_{t+1})^{L_t} \right\} \left\{ \sum_{n=L_{k-1}+1}^{L_k} \binom{n+R-1}{R-1} \rho_k^n \right\} \right] \quad (12)$$

Now, let us define $PB_{r_{ij}}$ as the probability of blocking suffered by the class-r_{ij} packets which the same at all nodes. Then we are able to derive $PB_{r_{ij}}$ obviously that,

$$PB_{r_{ij}} = \Pr(B \geqslant n \geqslant L_{r_{ij}}) \quad (13)$$

Moreover, because of L_m=B, then the probability of blocking suffered by class-m packets is as follow,

$$PB_m = \Pr(n=B). \quad (14)$$

Finally, using (11),(13) and (14) and after some algebra, we obtain

$$PB_{r_{ij}} = K \sum_{k=r_{ij}}^{m-1} \left[\left\{ \prod_{t=0}^{k} (\rho_t/\rho_{t+1})^{L_t} \right\} \left\{ \sum_{n=L_k}^{L_{k+1}-1} \binom{n+R-1}{R-1} \rho_{k+1}^n \right\} \right]$$

$$+ K \left\{ \prod_{t=0}^{m-1} (\rho_t/\rho_{t+1})^{L_t} \right\} \binom{B+R-1}{R-1} \rho_m^B \quad (15)$$

3.3 Network Throughput

Γ is now referred to as the offered load. Let us define Ps as the probability that in steady state, a packet transmitted over the network reachs its destination node successfully. And let Γ_s denotes network throughput (s for successful traffic). Then clearly that

$$Ps = \Gamma_s/\Gamma. \quad (16)$$

Now, let us define f_k as the fraction (or probability) of node-pairs at distance k hops. If the number of node pairs at distance k hops, as derived from the shortest path routing policy, is denoted by a_k, then obviously that

$$f_k = a_k/N(N-1). \quad (17)$$

Next, we define $Ps^{(k)}$ as the probability that in steady state, a packet transmitted over a node pair of distances k hops reachs its destination node successfully. Because of the uniform traffic assumption of our symmetrical network, we also get

$$Ps = \Gamma_s/\Gamma = \sum_{k \geqslant 1} Ps^{(k)} f_k \quad (18)$$

However, in order to obtain the throughput Γ_s corresponds to a given load Γ under the PF-C scheme, we have to express $Ps^{(k)}$ in terms of blocking probability $PB_{r_{ij}}$. And in order to derive $PB_{r_{ij}}$ we need to express $\lambda_{r_{ij}}$ in terms of offered load Γ. This can be done after the determination of priority function $P_f(i,j)$, as we will show concretely in the next section.

4. NUMERICAL APPLICATION

4.1 Priorities Decision and Network Throughput

We consider that it is more effective to give a higher priority to a packet when it has traveled over a large number of hops (i is large) and has very few hops to reach its destination (j is small), in comparison with another packets. Therefore, we have to find an appropriate rule for formulating $P_f(i,j)$ in terms of i and j, in order to improve the maximum network throughput.

In this section we present a priorities decision rule for PF-C scheme based on the difference of i and j, say i-j.

Let H_{max} be the longest path length in hops, as derived from the shortest path deterministic routing policy. Then the priority function of a class-r_{ij} packets is define as follow.

$$P_f(i,j) = r_{ij} = H_{max} + i - j \qquad (19)$$

In this decision rule, we assign increasing priorities to any class-r_{ij} packet as large as i-j. If the distance of any source-destination node pair is denoted by k, and $k=1,\ldots,H_{max}$; then we rewrite equation (19) in terms of k and i as follow,

$$P_f(i,j) = r_{ij} = H_{max} - k + 2i \qquad (20)$$

where $i=1,\ldots,k$ and $i+j=k$.

Moreover, because of $r_{ij}=0,\ldots,m$; then $m=2H_{max}$ and there are at most $1+2H_{max}$ different classes of packets in this system.

We are now able to derive the probability that a packet that traveled over a node pair of distance k will reach its destination successfully. This was denoted by $Ps^{(k)}$, then by the network symmetry, we have

$$Ps^{(k)} = \prod_{i=0}^{k}(1 - PB_{H_{max}-k+2i}) \qquad (21)$$

Hence, by replacing $Ps^{(k)}$ in equation (18) we obtain

$$Ps = \Gamma s/\Gamma = \sum_{k=1}^{H_{max}}\left[\prod_{i=0}^{k}(1 - PB_{H_{max}-k+2i})f_k\right] \qquad (22)$$

According to the uniform traffic assumption and the behavior of the decision rule, we can formulate the equation of $\lambda_{r_{ij}}$ in terms of γ and $PB_{r_{ij}}$ with respect to the value of r_{ij} and H_{max} as shown in (23).

$$\lambda_{r_{ij}} = \begin{cases} \left[\sum_{x=r_{ij}-H_{max}}^{[r_{ij}/2]}\left\{\prod_{y=1}^{x}(1-PB_{r_{ij}-2y})\right\}\right]\gamma & \text{when } r_{ij} > H_{max} \\ \left[\sum_{x=1}^{[r_{ij}/2]}\left\{\prod_{y=1}^{x}(1-PB_{r_{ij}-2y})\right\}\right]\gamma & \text{when } r_{ij} = H_{max} \\ \left[1+\sum_{x=1}^{[r_{ij}/2]}\left\{\prod_{y=1}^{x}(1-PB_{r_{ij}-2y})\right\}\right]\gamma & \text{when } 2 \leqslant r_{ij} < H_{max} \end{cases} \qquad (23)$$

where $[r_{ij}/2]$ is a Gauss notation, denotes the greatest integer not greater than $r_{ij}/2$. Furthermore, for $r_{ij}=0$ and $r_{ij}=1$ we always get $\lambda_0 = \gamma$ and $\lambda_1 = \gamma$.

Finally, solving the system of equations compose of (15) and (23) we obtain $PB_{r_{ij}}$ for a given load Γ. Replaceing $PB_{r_{ij}}$ in (22) we obtain network throughput Γs.

However, due to the complexity of equation (15) and (23), it is hard to obtain the exact computation. Because each equation forms a nonlinear function of several variables and of degree n $(n \geqslant 1)$. Whereby, we use the approximation computation introduced by Powell[10]. This we do for Loop networks.

Now, let us call the policies proposed by Giesler et al., and by Kamoun as P-C and N-C policy respectively. Here, we can show that the analytic results of PF-C scheme can apply to both policies as follow.

Consequently, the priority function of P-C is that

$$P_f(i,j) = r_{ij} = i \quad \text{for all } i=0,\ldots,H_{max}$$

and of N-C is that

$$P_f(i,j) = r_{ij} = \begin{cases} 0 & \text{when } i=0 \text{ ("new")} \\ 1 & \text{when } i \geqslant 1 \text{ ("transit")} \end{cases}$$

Similarly, as we do for PF-C scheme, the results for both policies are as follow:

In case of P-C scheme;

$$Ps = \Gamma s/\Gamma = \sum_{k\geqslant 1}\left[\left\{\prod_{i=0}^{k}(1 - PB_i)\right\}f_k\right] \qquad (24)$$

$$\lambda_0 = (N-1)\gamma/R$$

$$\lambda_r = \left\{\prod_{i=0}^{r-1}(1 - PB_i)\right\}\left[\left\{N-1-R(r-1)\right\}/R\right]\gamma \qquad (25)$$

In case of N-C scheme;

$$P_s = \Gamma s/\Gamma = (1 - PB_0)\left[\sum_{k\geqslant 1}(1 - PB_1)^k f_k\right] \qquad (26)$$

$$\lambda_0 = (N-1)\gamma/R$$

$$\lambda_1 = (1-PB_0)\left[\sum_{k\geqslant 1}(1-PB_1)^{k-1}\left\{\left\{N-1-R(k-1)\right\}/R\right\}\right]\gamma \qquad (27)$$

Note that, we derived the above equations by consideration of systems which all packets that reach their destinations are also need to store in the nodal buffers before delivered to their sink Host (i.e., for reassembly and resequence).

4.2 Numerical Results

We consider a Loop networks with N nodes. According to the shortest path deterministic routing policy, the maximum path length H_{max} is determined as follow:

$$H_{max} = \begin{cases} (N-1)/2 & \text{if N is an odd number} \\ N/2 & \text{if N is an even number} \end{cases}$$

More precisely, we define normalized throughput S and load G as follow[3].

$$S = \frac{h\Gamma_S}{\mu CNR} \quad ; \quad G = \frac{h\Gamma}{\mu CNR}$$

where h is the network average shortest path length when the network is assumed to be no-loss system (i.e.,infinite storage assumption). For Loop networks, h=(N+1)/4 if N is odd and $h=N^2/4(N-1)$ if N is even number.

The numerical results of PF-C, P-C and N-C are shown in Fig.4,5,6 respectively, in comparison with no control environment. The number of nodes N=5, the buffers size B=6 for each node and μC=10(packets/sec.).

In this system H_{max}=2, h=1.5, and R=2. Hence, we have $2H_{max}+1=5$ classes of packets correspond to our PF-C scheme, in the whole system. However, there are only 3 classes of packets on P-C scheme and 2 classes of packets on N-C scheme. Therefore, five limit values are assigned in PF-C scheme. Similarly, three and two limit values are assigned in P-C and N-C respectively.

Note that in a finite storage environment, G is no longer the channel utilization. Furthermore the maximum normalized throughput S is smaller than or equal to one.

The result of PF-C scheme in Fig.4 shows a very good behavior with all parameters (a set of limit values). The throughput increases when offered G increases, reashs a maximum value and never decreases even if G becomes large.

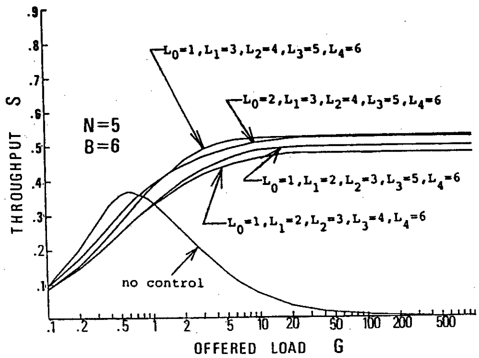

Fig.4 Throughput vs. offered load for 5-node Loop network based on PF-C scheme.

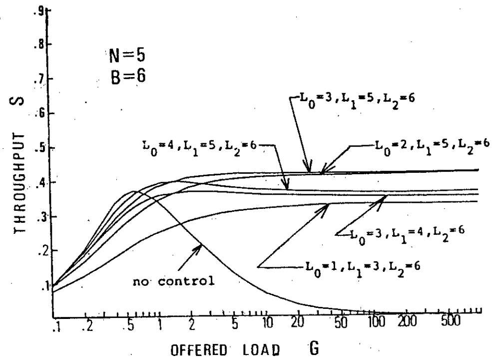

Fig.5 Throughput vs. offered load for 5-node Loop network based on P-C scheme.

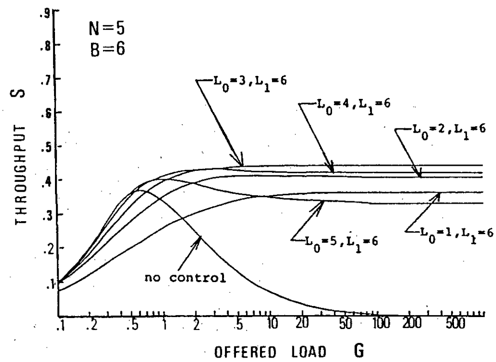

Fig.6 Throughput vs. offered load for 5-node Loop network based on N-C scheme.

But the results of P-C and N-C in Fig.5 and Fig.6 show that with an unsuitable limit value, S decreses toward an asymptotic value when G becomes greater.

However, all of them improve the network throughput. The maximum throughput curves of each policy are shown in Fig.7. And in Fig.8, we write the Ps curves which correspond to the maximum throughput curves in Fig.7. As the results, it is obviously to say that our PF-C scheme with packet priorities based on the difference of i and j (mentioned in section 4.1) shows a very good result than another two schemes.

5. CONCLUSION

In this paper we present a new approach of congestion control, namely, PF-C policy. We propose the ideas and describe the control protocol, then we analyze in the context of symmetrical networks. The priorities decision presented in this paper is one example based on PF-C policy. The numerical results of this example clearly demonstrate a good behavior in comparison with another two policies.

Further studies on packet priorities decision based on PF-C scheme are underway in order to achieve an optimal policy.

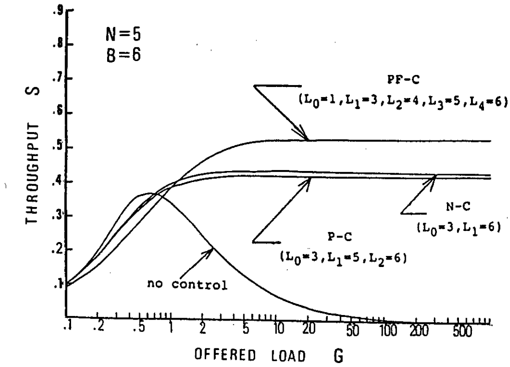

Fig.7 A comparison on the maximum throughput of each policy (5-node Loop network).

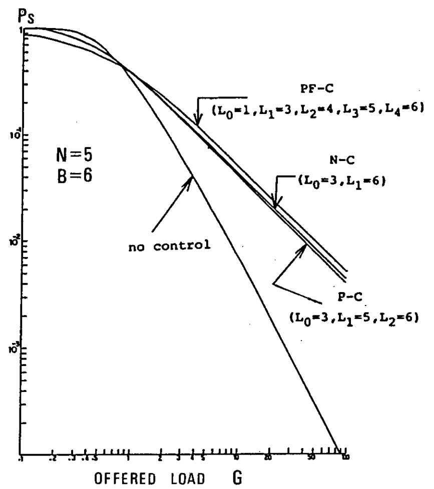

Fig.8 Probability of successful transmission correspond to max. throughput in Fig.7.

REFFERENCES

(1) A.Giesler,et.al:"Free Buffer Allocation-An Investigation by Simulation",Computer Networks, vol.1,July,1978,pp.191-208.

(2) S.S.Lam;M.Reiser:"Congestion Control of Store-and-Forward Networks by Input Buffer Limit :An Analysis",IEEE,COM-27,no.1, January,1979.

(3) F.Kamoun:"A Drop and Throttle Flow Control Policy for Computer Networks",IEEE,COM-29, no4,April,1981.

(4) Y.Tezuka and H.Sanada,et al.:"Utilization of Distance-Dependent Priority on Store-and-Forward Communication Network",Paper of Technical Group, TGSE75-11, IECE Japan (1975).

(5) M.Gerla and L.KLeinrock:"Flow Control: A Comparative Survey",IEEE,COM-28,no.4, April,1980.

(6) F.Kamoun:"Design Considerations for Large Computer Communication Networks",Ph.D. Dissertation, Dep. Computer Science, University of California, 1976.

(7) F.Kamoun and L.Kleinrock:"Analysis of Shared Finite Storage in a Computer Networks: Node Environment under General Traffic Condition",IEEE,COM_28,no.7,July, 1980.

(8) L.Kleinrock:"Queueing Systems" Vol.2 :Computer Applications, New York, Wiley-Inter.,1976.

(9) K.M.Chandy and C.H.Sauer,"Computer Systems Performance Modeling",Prentice-Hall,1981.

(10) M.J.D.Powell:"A Method for Minimizing a Sum of Squares of Non-Linear Functions without Calculating derivative", Computer Journal,vol.7,pp. 303-307 1965.