Fairness concept in terms of the utilization

Hakyong Kim, Yongtak Lee, and Kiseon Kim

Indexing terms: fairness, fairness index

In this Letter, we discuss the concept of the fairness and define a fairness index in terms of the utilization. The fairness concept based on the utilization not only guarantees the fairness among entities but also culminates the switch throughput.

Introduction: Fairness is an important property that a good scheduling algorithm must hold with itself [1]. Normally, the fairness can be evaluated by calculating the ratio of the served traffic to the arrived traffic, where the ratio is the throughput of a connection. The fairness based on the throughput was defined to specify the fairness among connections in an output of the output-queued switch. For the reason, it could cause the problem of degradation in the switch utilization when we apply the fairness concept directly to the input-queued switch.

The example presented in Fig. 2 shows the problem that the conventional fairness definition reduces the switch utilization. In the example, we assume that input 1 has two connections, C_1 and C_2 whose constant arrival rate is 100 Mbps and 10 Mbps, respectively. Assuming that both outputs have the maximum service rate of 10 Mbps and that output A serves the traffic of connection C_1 at its maximum service rate, then output B must serve the traffic of connection C_2 at only 1 Mbps to guarantee fairness between the two connections. In this example, output B should waste its capacity of 9 Mbps or 90% of its maximum service rate to balance fairness between two connections. The phenomenon of under-utilization deteriorates as the difference among the traffic rates gets larger.

In this Letter, therefore, we define the fairness in terms of the utilization to solve the under-utilization problem occurred in maintaining the fairness. We also modify the fairness index by Jain [2] to compare quantitatively the fairness of scheduling algorithms for the multiple input-queued (MIQ) switch.

Fairness and fairness index in terms of the utilization: In this Letter, we define the fairness in terms of the utilization. That is, we evaluate the fairness by obtaining the ratio of the average service rate to the maximum service rate. In the output-queued switch, it is easy to find that the fairness defined in terms of the utilization is equal to its counterpart based on the connection's throughput. The fairness definition is applicable to the input-queued switch and the switch using multiple queues as well as the output-queued switch, and its acceptability will be demonstrated through two examples after defining the fairness index.

In order to compare the fairness quantitatively, we can think of the fairness index having its value in a bounded range, possibly 0 to 1. Thus far, Jain's fairness index porposed in reference [2] has been widely used as a representative fairness index. In this Letter, we also use Jain's index by modifying a parameter in terms of the utilization. The modified fairness index is as follows:

s_k : average service rate of the kth entity

m_k : maximum service rate or capacity of the kth entity

$x_k = s_k/m_k$

$$\text{Fairness index} = \frac{(\sum x_k)^2}{n \times \sum (x_k)^2}, \tag{1}$$

where n implies the number of entities with the non-zero load and x_k denotes the utilization or the normalized service rate of the kth entity. Supposing the MIQ switch, the entity is either queues, inputs, or outputs in the MIQ switch. The difference of the fairness index in eq. (1) from Jain's is that m_k denotes the maximum service rate or capacity of the kth entity, not the allocated or arrival rate of the kth connection. The fairness index is bounded between 0 and 1 and the fairness index of 1.0 represents the highest degree of fairness. If the fairness index is equal to 0.9, for example, it means that 10% of entities might be starved.

The fairness index of eq. (1) can be expressed in terms of variance by dividing both numerator and denominator by n^2. Then it becomes immediately

$$\begin{aligned}\text{Fairness index} &= \left(\frac{1}{n}\sum x_k\right)^2 \Big/ \left(\frac{1}{n}\sum x_k^2\right) \\ &= \frac{(E[X])^2}{var(X) + (E[X])^2}, \qquad (2)\end{aligned}$$

where $E[X]$ and $var(X)$ are the expectation and variance of the random variable X. Eq. (2) implies that the switching system can serve entities more fairly as the variance for the entity's utilization is small.

Evaluation of fairness in the MIQ switch: In order to assess the fairness in the input-queued switch including the MIQ switch, we have to use the fairness concept based on the utilization as mentioned previously, rather than the fairness based on the connection's throughput. In the MIQ switch, we can think of three types of fairness: inter-queue fairness, inter-input fairness, and inter-output fairness. Inter-queue fairness is the fairness among queues in the same input so that it is also referred to as the intra-input fairness. Inter-input and inter-output fairnesses are the fairness among the inputs and outputs, respectively.

In order to compare these fairnesses quantitatively, we consider two cases as shown in Fig. 2 and Fig. 3. In the examples, λ_{ij} and μ_{ij} designate the mean arrival rate at input i for output j and the service amount of λ_{ij} at output j, respectively. Generally, it is meaningless to evaluate and compare the fairness in the under-loaded cases, since almost traffic is served under the circumstance. Thus, above two examples assume the over-loaded and over-subscribed cases. Table 1 tabulates the fairness indices obtained by eq. (2) for the three types of fairness shown in Fig. 2

and Fig. 3.

In terms of inter-queue fairness, firstly, the SLIP algorithm proposed by McKeown is known to be fairest among the scheduling algorithms for the MIQ switch, since SLIP uses an unsynchornized round-robin matching approach [1]. Table 1 also proves this fact. On the other hand, the Chessboard (CB) algorithm is superior in terms of inter-input and inter-output fairnesses, since CB is designed to maximize the switch throughput [4]. Finally, parallel iterative matching (PIM) is in the middle for all types of fairness, since it is conceived to reduce the operation speed of the scheduling algorithm rather than to ensure fairness. Notice here that the result for the output-fairness is the same as when we use the fairness concept based on the connection's throughput, and this fact shows the validity for our fairness based on the utilization as a general definition on the fairness.

Conclusions: We have defined the fairness and fairness index in terms of the utilization to evaluate the fairness between entities, such as queues, inputs, or outputs. The fairness concept guarantees the higher switch utilization as well as the fairness between entities.

References

1. MCKEOWN, N.: 'The *i*SLIP scheduling algorithm for input-queued switches', *IEEE/ACM Tr. Networking*, April 1999, 7, (2), pp. 188-201.

2. JAIN, R.: 'Fairness: How to measure quantitatively?', ATM Forum/94-0881, Sept. 1994.

3. KIM, H., LEE, Y.T., OH, C.H., YOON, H.H., and KIM, K.S.: 'Cell selection algorithm for the multiple input-queued ATM switch: Chessboard and Random cell selections', *Proc. IEEE ATM Workshop'99*, Kochi, Japan, 1999. pp.259-264.

4. ANDERSON, T.E., OWICHI, S.S., SAXE, J.B., and THACKER, C.P.: 'High-speed switch scheduling for local-area networks', *ACM Tr. Computer Systems*, Nov. 1993, 11, (4), pp.319-352.

Authors' affiliations:

Hakyong Kim, Yongtak Lee and Kiseon Kim (Department of Information and Communications, Kwang-Ju Institute of Science and Technology (K-JIST), 1 Oryong-dong, Puk-gu, Kwang-Ju, 500-712, S. Korea)

e-mail: hykim@ieee.org or hykim@charly.kjist.ac.kr

Figure captions:

Fig. 1 Fairness based on the connection's throughput could cause under-utilization in the input-queued switch

Fig. 2 Fairness comparison in the over-loaded and over-subscribed case (2×2 switch)

Fig. 3 Fairness comparison in the over-loaded and over-subscribed case (4×4 switch)

Table captions:

Table 1 Fairness comparison of PIM, SLIP, and CB scheduling algorithms. Fairness indices are calculated for the examples of Fig. 2 and Fig. 3. Inter-queue fairness is calculated for the queues of input 1 and inter-output fairness is calculated for output 1 and output 2.

Figure 1

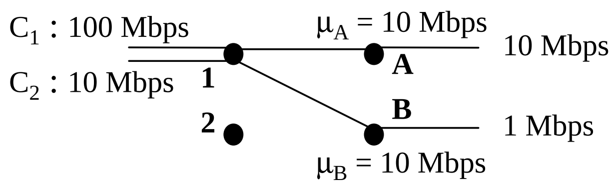
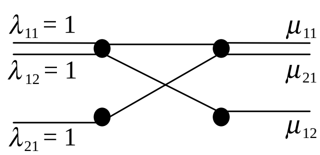


Figure 2

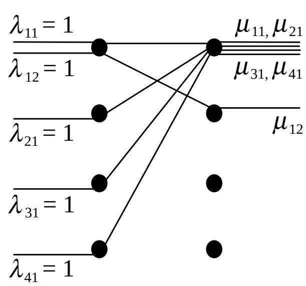


(a)

	PIM	SLIP	CB
μ_{11}	0.25	0.5	0.0
μ_{12}	0.75	0.5	1.0
μ_{21}	0.75	0.5	1.0
Σ	1.75	1.5	2.0

(b)

Figure 3



(a)

	PIM	SLIP	CB
μ_{11}	0.25	0.25	0.00
μ_{12}	0.75	0.75	1.00
μ_{21}	0.25	0.25	0.33
μ_{31}	0.25	0.25	0.33
μ_{41}	0.25	0.25	0.33
Σ	1.75	1.75	2.00

(b)

Table 1

	inter-queue fairness		inter-input fairness		inter-output fairness	
	Fig.2	Fig.3	Fig.2	Fig.3	Fig.2	Fig.3
PIM	0.80	0.80	0.98	0.64	0.98	0.98
SLIP	1.00	0.80	0.90	0.64	0.90	0.98
CB	0.50	0.50	1.00	0.75	1.00	1.00