A Novel Flow Control Scheme for Best Effort Traffics in Network-on-Chip Based on Weighted Max-Min-Fairness

Fahimeh Jafari, Mohammad H. Yaghmaee
Ferdowsi University of Mashhad
Mashhad, Iran
fjafari@wali.um.ac.ir, hyaghmae@ferdowsi.um.ac.ir

Abstract- **Network on Chip (NoC) has been proposed as an attractive alternative to traditional dedicated busses in order to achieve modularity and high performance in the future System-on-Chip (SoC) designs. Recently, end to end flow control has gained popularity in the design process of network-on-chip based SoCs. Where flow control is employed, fairness issues need to be considered as well. In fact, one of most difficult aspects of flow control is that of treating all sources fairly when it is necessary to turn traffic away from the network. In this paper, we propose a flow control scheme which admits Max-Min fairness criterion for all sources. In fact, we formulate Weighted Max-Min fairness criterion for the NoC architecture and presente implementation to be used as flow control mechanism.**

Keywords: Network on Chip; flow control; Weighted Max-Min fairness.

I. Introduction

The high level of system integration characterizing Multi-Processor Systems on Chip (MPSoCs) is raising the scalability issue for communication architectures. The problems emanating from the scalability issue in the MPSoCs have been remedied by the emergence of Network-on Chip (NoC) architectures [1]. Due to the rapid growth of the number of processing elements in NoCs [2], employing efficient policies for flow control has become an inevitable subject in the design of NoCs to provide the required Quality of Service (QoS). A NoC must have network level flow control in order to avoid congestion in the bottleneck links.

Recently, QoS provisioning in NoC's environment has attracted many researchers and currently is the focus of many literatures in NoC research community. NoCs are expected to serve as multimedia servers and are required to carry both Best Effort (BE) and Guaranteed Service (GS) traffics. It's trivial that such a networked architecture with data services should have some policies to avoid congestion. Congestion Control in data networks is known as a widely-studied issue over the past two decades. However, it is still a novel issue in NoC and to the best of our knowledge only a few works have been carried out in this field.

Many strategies for flow control have been proposed for off-chip networks, e.g. data networks, etc. [3-5]. On-chip networks pose different challenges. For instance, in off-chip environments, to overcome congestion in links, packet dropping is allowed. On the contrary, reliability of on-chip wires makes NoCs a loss-less environment.

So far, several works have focused on this issue for NoC architectures. In [6], a prediction-based flow-control strategy for on-chip networks has been proposed in which each router predicts the buffer occupancy to sense congestion. In [7] link utilization is used as a congestion measure and a Model Prediction-based Controller (MPC), determines source rates. Dyad [8] controls the congestion by using adaptive routing when the NoC faces congestion.

Where flow control is employed, fairness issues need to be considered as well [9]. In fact, one of the most difficult aspects of flow control is to choose a policy to accommodate a fair rate allocation. All of the abovementioned studies only regarded the flow control by taking into account the constraints of the system. To the best of our knowledge no policy to maintain fairness among sources has been chosen.

Different flow control approaches can be classified with respect to the fairness criteria, in favor of which rate allocation is done. One of the famous forms of fairness criterion is Max-Min fairness, which has been discussed in earlier literature and described clearly in [10]. Our main contribution in this paper is to present a flow control scheme for Best Effort traffic in NoC which satisfies Weighted Max-Min fairness criterion through the analysis of mathematical model and simulation. Our framework is mainly adopted from the seminal work [10] which presents a basic Max-Min fairness. In this paper, we formulate Weighted Max-Min problem for the NoC architecture.

The organization of the paper is as follows. In Section II we present the system model and flow control problem. In section III we present an iterative algorithm as the solution to the flow control optimization problem. Section IV

978-1-4244-2751-2/08/$25.00 ©2008 IEEE

presents the simulation results and discussion about them. Finally, section IV concludes the paper.

II. System Model and Flow Control Problem

We consider a NoC architecture which is based on a two dimensional mesh topology and wormhole routing. We also assume that the NoC architecture is lossless, and packets traverse the network on a shortest path using a deadlock free XY routing [2].

We model the flow control in NoC as the solution to an optimization problem. For the sake of convenience, we turn the aforementioned NoC architecture into a mathematically modeled network. In this respect, we consider NoC as a network with a set of bidirectional links L and a set of sources S. Each source $s \in S$ consists of processing elements, routers and input/output ports. Each link $l \in L$ is a set of wires, busses and channels that are responsible for connecting different parts of the NoC and has a fixed capacity of c_l packets/sec. We denote the set of sources that share link l by $S(l)$. Similarly, the set of links that source s passes through, is denoted by $L(s)$. By definition, $l \in L(s)$ if and only if $s \in S(l)$.

As discussed in section 1, there are two types of traffic in a NoC: Guaranteed Service (GS) and Best Effort (BE) traffic. For notational convenience, we divide S into two parts, each one representing sources with the same kind of traffic. In this respect, we denote the set of sources with BE and GS traffic by S_{BE} and S_{GS}, respectively. Each link l is shared between the two aforementioned traffics. GS sources will obtain the required amount of the capacity of links and BE sources benefit from the remainder.

Our objective is to choose source rates with BE traffic so that to maximize the type of weighted α-Fair function in which $\alpha = \infty$. Weighted α-Fair function is define as below [11]:

$$U(x,\alpha,w) = \begin{cases} w\dfrac{x^{1-\alpha}}{1-\alpha} & \alpha \neq 1 \\ w \ln x & \alpha = 1 \end{cases} \tag{1}$$

where $\alpha > 0$ is a parameter. Therefore we define our flow control problem as below:

$$\lim_{\alpha\to\infty} \max_{x_s} \sum_s w_s \frac{x_s^{1-\alpha}}{1-\alpha} \tag{2}$$

subject to:

$$\sum_{s\in S_{BE}(l)} x_s + \sum_{s\in S_{GS}(l)} x_s \le c_l \qquad \forall l \in L \tag{3}$$

$$x_s > 0 \qquad \forall s \in S_{BE} \tag{4}$$

Optimization variables are BE source rates, i.e. $(x_s,\ s \in S_{BE})$. The constraint (3) states that the sum of BE source rates passing thorough link l cannot exceed its free capacity, i.e. the portion of c_l which has not been allocated to GS traffic.

Problem (2) is a convex optimization problem with linear constraints. Hence it admits a unique maximizer [12]. Treating problem (2) using such an extreme case is not disobedient. However, the following theorem states that it can be reduced to a well-known type of disciplined optimization problem known as Weighted Max-Min problem. The following definition states the formal definition of WMMF.

THEOREM 1**:** *α-Fair maximization problem for $\alpha = \infty$ reduces to weighted max-min optimization problem, as below* [11]:

$$\max_x \min_{s\in S} w_s x_s \tag{5}$$

subject to:

$$\sum_{s\in S_{BE}(l)} x_s + \sum_{s\in S_{GS}(l)} x_s \le c_l \qquad \forall l \in L \tag{6}$$

$$x_s \ge 0 \qquad \forall s \in S_{BE} \tag{7}$$

For notational convenience, we define:

$$u = \min_{s\in S} w_s x_s$$

$$\hat{c}_l = c_l - \sum_{s\in S_{GS}(l)} x_s$$

To solve the above problem, it should be converted so as to be in the form of disciplined optimization problems [13] as follows:

$$\max u \tag{8}$$

subject to:

$$u \le w_s x_s \quad \forall s \in S \tag{9}$$

$$\sum_{s\in S_{BE}(l)} x_s \le \hat{c}_l \qquad \forall l \in L \tag{10}$$

$$x_s > 0 \qquad \forall s \in S_{BE} \tag{11}$$

Weighted Max-Min optimization problem is a widely-studied problem formulation in the design of data networks. Weighted Max-Min problem has an important property which discriminates it from the others. The optimal solution to the weighted max-min problem exists, a specific type of fairness characteristic know as Weighted Max-Min Fairness (WMMF) is admitted which will formally be defined in the following.

DEFINITION 1: (Weighted Max-Min Fairness [14]). *Given some positive constants w_i (called the "weights"), $x = (x_s, s \in S)$ is "Weighted-Max-Min Fair", if and only if increasing one component x_s must be at the expense of decreasing some other component x_t such as $x_t/w_t \le x_s/w_s$.*

If we assume $w_s = 1 \ \forall s \in S$, WMMF will be known as Max-Min Fairness (MMF) which will formally be defined in the following.

DEFINITION 2: (Max-Min Fairness [14]). *A feasible rate allocation $x = (x_s, s \in S)$ is said to be "Max-Min Fair" (MMF) if and only if an increase of any rate within the* *domain of feasible allocations must be at the cost of a decrease of some already smaller rate. Formally, for any other feasible allocation y, if $y_s > x_s$ then there must exist some s' such that $x_{s'} \leq x_s$ and $y_{s'} < x_{s'}$.*

Depending on the network topology, a max-min fair allocation may or may not exist. However, if it exists, it is unique (see [14] for proof). In what follows the condition under which the Max-Min rate allocation exists will be stated. Before we proceed to this condition, we define the concept of bottleneck link.

DEFINITION 3: (Bottleneck Link [14]), *With our system model above, we say that link l is a bottleneck for source s if and only if*

1. *link l is saturated:* $\sum_{s \in S_{BE}(l)} x_s + \sum_{s \in S_{GS}(l)} x_s = c_l$
2. *source s on link l has the maximum rate among all sources using link l.*

Intuitively, a bottleneck link for source s is a link which limits x_s.

THEOREM 2: *A max-min fair rate allocation exists if and only if every source has a bottleneck link.*

Proof: See [14] for proof.

Any discussion of the performance of a rate allocation scheme must address the issue of fairness, since there exist situations where a given scheme might maximize network throughput, for example, while denying access for some users or sources. Max-Min fairness is one the significant fairness criteria. Crudely speaking, a set of rates is max-min fair if no rate can be increased without simultaneously decreasing another rate which is already smaller. In a network with a single bottleneck link, max-min fairness simply means that flows passing through the bottleneck link would have equal rates.

The most famous and simplest algorithm to solve the max-min problem is the well-known Progressive Filling Algorithm [10]. We would like to modify the progressive filling algorithm as an iterative solution to the weighted max-min problem (8) for our system model. We finally would like to utilize it as BE flow control mechanism in NoC. The modified version of the progressive filling as a BE flow control mechanism is listed below as algorithm 1.

Algorithm 1: Weighted Max-Min Fair (WMMF) Flow Control Algorithm for BE in NoC

Initialization:

1. Initialize $\hat{c}_l$ of all links.
2. Define:
   a. T as the set of sources not passing through any saturated link.
   b. B as the set of saturated links.
3. Set source rate vector to zero.
4. Initialize $T = S_{BE}$ and $B = \varnothing$.

Loop:

Do until ($T = \varnothing$)

1. $\Delta_s = \min_{l \in (L-B)} \left[\left(c_l - \sum_{s \in (S_{BE}-T)} R_{ls} x_s(t) \right) \Big/ \sum_{s \in T} w_s R_{ls} \right]$
2. $x_s(t+1) = x_s(t) + w_s \Delta_s \quad \forall s \in T$
3. Calculate new bottleneck links and update B.
4. $\forall s \in T$; if s passes through any saturated link then $T \Leftarrow T - \{s\}$

Output:

Communicate BE source rates to the corresponding nodes.

III. Weighted Max-Min-Fairness Algorithm

Theorem 2 is particularly useful in deriving a practical method for obtaining a max-min fair allocation, called “progressive filling”. The idea is as follows: rates of all flows are increased at the same pace, until one or more links are saturated. The rates of flows passing through saturated links are then frozen, and the other flows continue to increase rates. All the sources that are frozen have a bottleneck link. This is because they use a saturated link, and all other sources using the saturated link are frozen at the same time, or were frozen before, thus have a smaller or equal rate. The process is repeated until all rates are frozen. Lastly, when the process terminates, all sources have been frozen at some time and thus have a bottleneck link. Using Theorem 2, the allocation is max-min fair.

In the sequel, we modify the progressive filling algorithm as an iterative solution to the weighted max-min problem (8) for our system model and based on this algorithmic solution, we present a flow control scheme for BE traffic in NoC systems.

Thus, the aforementioned algorithm can be employed to control the flow of BE traffic in the NoC. The iterative algorithm can be addressed in distributed scenario. However, due to well-formed structure of the NoC, we focus on a centralized scheme; we use a controller like [7] to be mounted in the NoC to implement the above algorithm. The necessary requirement of such a controller is the ability to accommodate simple mathematical operations and the allocation of few wires to communicate flow control information to nodes with a light GS load.

IV. Simulation Results

In this section we examine the proposed flow control algorithms for a typical NoC architecture. In our scenario, we have used a NoC with 4×4 Mesh topology which consists of 16 nodes communicating using 24 shared bidirectional links; each one has a fixed capacity of 1 Gbps. In our scheme, packets traverse the network on a shortest path using a deadlock free XY routing. We also assume that each packet consists of 500 flits and each flit is 16 bit long.

In order to simulate our scheme, some nodes are considered to have a GS data, such as Multimedia, etc., to be sent to a destination while other nodes, which maybe in the set of nodes with GS traffic, have a BE traffic to be sent. As stated in section 2, GS sources will obtain the required amount of the capacity of links and the remainder should be allocated to BE traffics.

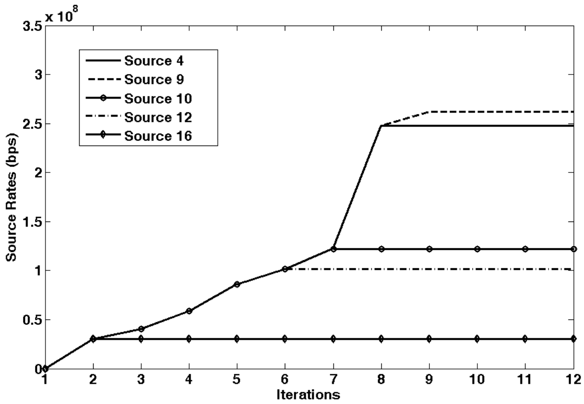

Fig. 1 Source Rates vs. Iteration Steps for Max-Min

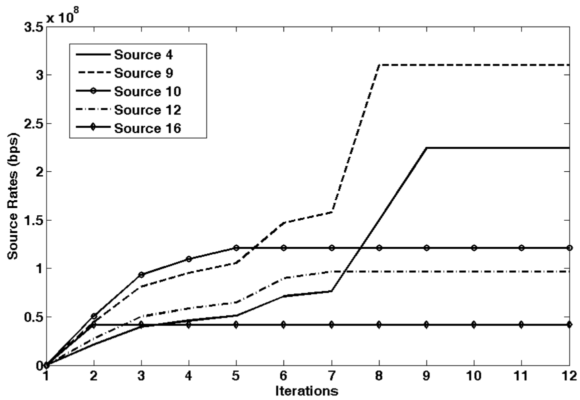

Fig. 2 Source Rates vs. Iterations for Weighted Max-Min with w_1

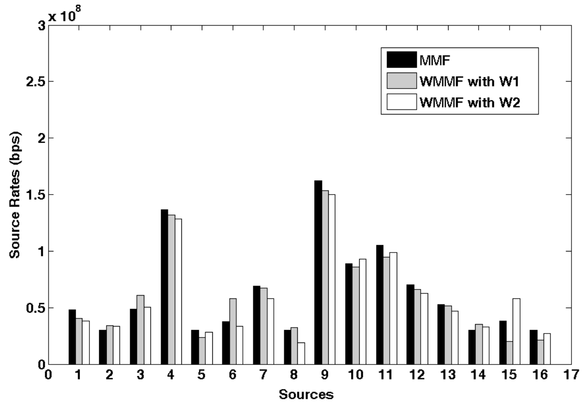

Fig. 3 Comparison of Max-Min, Weighted Max-Min with w_1 and Weighted Max-Min with w_2

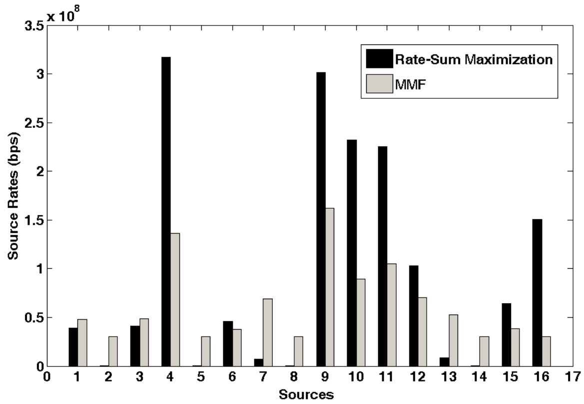

Fig. 4 Comparison between Rate-Sum and Max-Min

We are mainly interested in investigating the fairness properties among source rates.

A. WMMF with Various Weights

We obtained source rates using proposed algorithm in MATLAB. Rate variation versus iteration steps for both MMF and WMMF (with weight vector w_1) flow control schemes are shown in Fig. 1 and Fig. 2, respectively.

We solved problem (8) with different weight vectors such as w_2 and w_3 (due to space limit, values of weight vectors have been omitted) to control the priority of resource allocation. Such weighting factors can be appropriately derived so that to designate the network resources (link capacities) in favor of source priorities.

For the sake of convenience in comparing these schemes, steady state source rates for all sources are depicted in Fig. 3. It is clear from Fig. 3 that with different weight vectors, priorities of sources vary significantly and as a result, WMMF lead to great differences in rate allocations.

B. WMMF Fairness Metrics

In order to compare the results of the proposed Max-Min fair flow control with other fairness criteria, we have used rate allocation based on maximizing the sum of source rates, i.e. the so-called Rate-Sum Maximization. For comparing the two schemes, steady state source rates for all sources are depicted in Fig. 4. Comparing Rate-Sum and Max-Min in Fig. 4, it's evident that although Rate-Sum criterion aims to maximize the sum of source rates, there is no guarantee for the rates of *weak* sources, i.e. ones which achieve very small rate. Indeed, in many scenarios with Rate-Sum flow control, such sources will earn as small as zero. On the other hand, the weakest source in Max-Min scenario earns about 0.3 Gbps.

To compare the results of the above mentioned schemes in more detail, we have considered four parameters featuring the merit of the different schemes as following:

1. Least source rate
2. Variance of source rates with respect to mean value.
3. Jain's fairness Index (JFI) [15]
4. min-max ratio [15]

These parameters are presented in Fig. 5 and Fig 6. Jain's fairness Index and max-min ratio, are defined by (12) and (13), respectively.

$$\text{Jain's Fairness Index} = \frac{\left(\sum_{s=1}^{N} x_s\right)^2}{N\sum_{s=1}^{N} x_s^2} \tag{12}$$

$$\text{Min-Max Ratio} = \frac{\min_{s\in S} x_s}{\max_{s\in S} x_s} \tag{13}$$

Amongst the aforementioned parameters, Jain's Fairness Index, Min-Max Ratio and Variance of source rates for

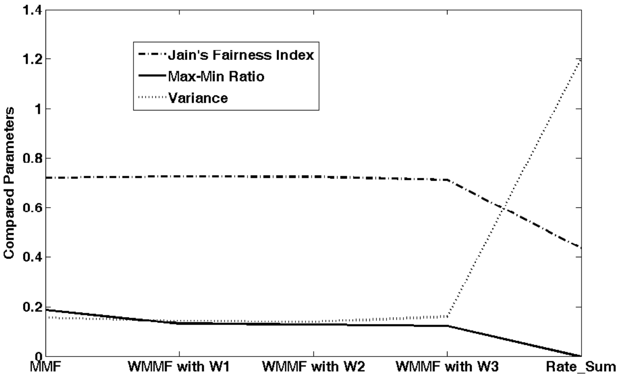

Fig. 5 Different parameters for Different scenarios

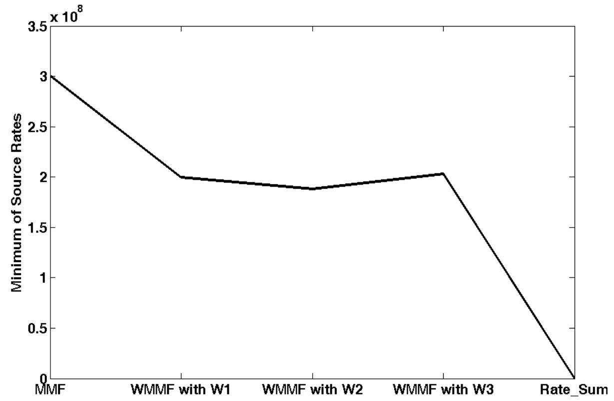

Fig. 6 Least source rate for Different scenarios

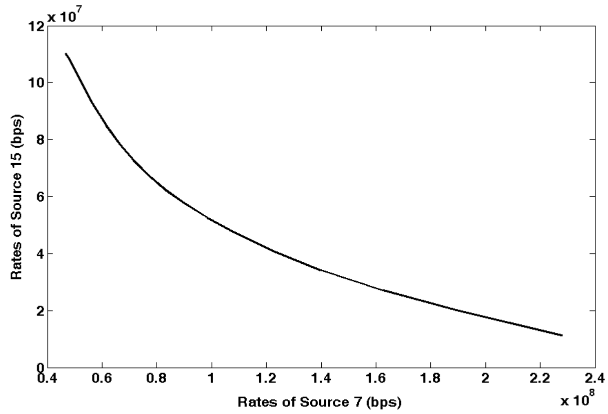

Fig. 7 Rate region for x_7 and x_{10}

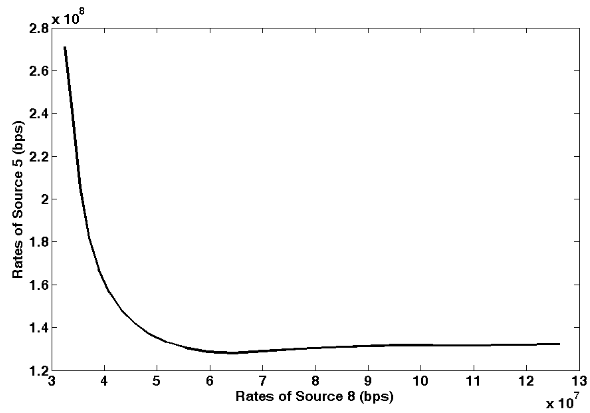

Fig. 8 Rate region for x_8 and x_5

MMF, WMMF (with three different weights) and Rate-Sum schemes are depicted in Fig. 5. It is apparent that using MMF and WMMF schemes, the variance of source rates are considerably less than Rate-Sum, which denotes the intrinsic fairness in these mechanisms with respect to Rate-Sum mechanism. Smaller variance results in the larger Min-Max Ratio and JFI; therefore MMF and WMMF schemes have greater Min-Max Ratio and JFI.

To have a better insight about the influence of weights on the MMF scheme, the rate of the weakest source for the aforementioned scenarios is shown in Fig. 6. It's apparent that with pure Max-Min scheme, the weakest user obtains the largest rate among other schemes. As the variance of weight elements increases, the weakest source's rate falls rapidly. Finally, in the Rate-Sum, the rate of the weakest source is approximately zero.

C. Rate-Region for WMMF

In order to analyze the effect of the weighting scheme in more detail, we introduce the concept of *Rate Region* for the flow control we considered in this paper. We think of a Rate Region as a region of all possible rate tuples $(x_1, x_2, ..., x_S)$ that satisfy link capacity constraints, i.e.

$$\left\{ (x_1, x_2, ..., x_S) \in \mathbb{R}^S \left| \sum_{s \in S(l)} x_s \le c_l ; \forall l \in L \right. \right\}$$

Rate region for two Weighted Max-Min scenarios are depicted in Fig. 7 and Fig. 8. We briefly discuss about some heuristic insights that can be obtained from these figures.

For the sake of simplicity in representing such a region, we fix the weight of all nodes to 1 and assume the weights of two nodes, say nodes *i* and *j*, are set to *w* and 2-*w*, respectively. Then, by sweeping *w* over [0,2] interval, we study the effect of the rate allocation on the rate of node *i* and *j*, e.g. x_i and x_j. In this respect, rate region can be depicted efficiently using a two-dimensional curve whose axes are x_i and x_j. In Fig. 7, we have *i*=7 and *j*=15. It is apparent that by varying the weight from 0 to 2, x_{15} would vary from 0 to 0.12 Gbps, however, for the source 7 such a variation is limited to 0.04 to 0.22 Gbps. This means that in the worst case source 7 would obtain a considerably larger weight with regard to source 15. In fact, source 15 is more sensitive to weight selection than source 7. Setting i=8 and j=5 yields the rate region depicted in Fig. 8. A similar discussion also holds and we conclude that source 8 is more sensitive to weighting, because the range over which its rates varies is much larger.

Another advantage of such rate regions, which is worth discussing, might be the selection of efficient weighting factors which suits the demands and constraints of the underlying system. Crudely speaking, we may study such a S-dimensional rate region by evaluating a number of simpler two-dimensional rate-regions, as with above, and then determine source pairs which are highly-sensitive to weight selection. Regarding such regions, based on the rate demands of sources, we can obtain the appropriate point of the region, thereby the corresponding weights.

V. CONCLUSION

In this paper we addressed the flow control problem for BE traffic in NoC systems. We considered two objectives. First, choosing source rates (IP loads) of BE traffics so that to accomplish flow control in response to demands at a reasonable level. Second, maintaining Weighted Max-Min fairness for all sources. Flow control was modeled as the solution to a simple algorithmic solution to an optimization problem. The algorithm can be implemented by a controller which admits a light communication and communication overhead.

REFRENCES

[1] L. Benini, and G. DeMicheli, ”Networks on Chips: A New SoC Paradigm.” *Computer*, 2002, vol. 35, no. 1, pp. 70-78.

[2] W. J. Dally, and B. Towles, ”Route Packets, Not Wires: On- Chip Interconnection Networks,” in *Proceedings of Design Automation Conference*, 2001, pp. 684-689.

[3] F. P. Kelly, A. Maulloo, and D. Tan, ”Rate control for communication networks: Shadow prices, proportional fairness, and stability.” *Operational Research Society*, 1998, vol. 49, no. 3, pp. 237-252.

[4] Y. Gu, H. O. Wang, and L. G. Yiguang Hong Bushnell, ”A predictive congestion control algorithm for high speed communication networks,” in *Proceedings of American Control Conference*, 2001, vol. 5, pp. 3779-3780.

[5] C. Yang, and A. V. S. Reddy, ”A taxonomy for congestion control algorithms in packet switching networks,” *IEEE Network*, 1995, vol. 9, no. 4, pp. 34-45.

[6] U. Y. Ogras, and R. Marculescu, ”Prediction-based flow control for network-on-chip traffic,” in *Proceedings of Design Automation Conference*, 2006.

[7] J. W. van den Brand, C. Ciordas, K. Goossens, and T. Basten, ”Congestion-Controlled Best-Effort Communication for Networks-on-Chip,” *Design, Automation and Test in Europe Conference*, 2007, pp. 948-953.

[8] J. Hu, and R. Marculescu, ”DyAD - smart routing for networks-onchip,” in *Proceedings of Design Automation Conference*, 2004, pp. 260-263.

[9] I. Cidon and I. Keidar, “Zooming in on Network on Chip Architectures.” *Technion Department of Electrical Engineering*, 2005.

[10] D. Bertsekas, and R. Gallager, *Data Networks*, Prentice-Hall, 1992.

[11] J. Mo, J. Walrand, ”Fair End-to-End Window-Based Congestion Control,” *IEEE/ACM Transactions on Networking*, 2000, vol. 8, no. 5, pp. 556-567.

[12] S. H. Low, and D. E. Lapsley, ”Optimization Flow Control, I: Basic Algorithm and Convergence,” *IEEE/ACM Transactions on Networking*, 1999, vol. 7, no. 6, pp. 861-874.

[13] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 1999.

[14] J. Y. Le Boudec, ”Rate adaptation, Congestion Control and Fairness: A Tutorial.” Ecole Polytechnique Federale de Lausanne (EPFL), 2001.

[15] R. Jain, D. Chiu, and W. Hawe, ”A Quantitative Measure of Fairness And Discrimination For Resource Allocation In Shared Computer Systems”, DEC Research Report TR-301, 1984.