An Approach for Optimal Bandwidth Allocation in Packet Processing Systems

Mahmood Ahmadi and Stephan Wong
Computer Engineering Laboratory
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
mahmadi, stephan@ce.et.tudelft.nl

Abstract

The increasing demand for more bandwidth and the increased application variety fuel the need for high performance network processors. A simple but highly repetitive task performed by such processors is packet processing. Typically, a network processor consists of a parallel processor core with a number of memory interfaces and special co-processors. Recently, distributed architectures are being utilized in the design of network processors. In such environments, a challenging problem is to allocate optimal bandwidth between different network processors (NPs) to achieve more performance. In this paper, the formulation and solution of an optimal bandwidth allocation strategy using queuing network for NP-based architectures at system level is proposed. The solution allocates optimal bandwidth between network processors in a grid-oriented environment. It encompasses a new formula based on the optimal capacity allocation concept in queuing network. our simulation results show that the proposed solution is able to enhance the response time in NP-based architectures when compared to a same NP-based architectures without optimal bandwidth allocation.

Keywords: Network processors, queuing theory, optimal bandwidth allocation.

1 Introduction

The bandwidth growth of networks increased almost exponentially in the past couple of years and is expected to continue for years to come. This has been fueled by new emerging technologies that are capable of achieving higher bandwidths. Consequently, new applications are being developed that take advantage of the new capabilities. In turn, more consumers are starting to use these applications and thereby further increasing the demand for higher bandwidth. The bandwidth growth and applications variety fueled the need for high performance network processors (NPs). Network processors combine the flexibility of general-purpose processors with the high performance of application-specific integrated circuits (ASICs). The type of processing in network processors is different from processors found in servers and workstations. Typically, a network processor comprises a parallel programmable processor core with a number of memory interfaces and special co-processors that are optimized for packet processing [1][2][4]. The packet processing tasks have specific requirements in term of response time and throughput. The traditional NP consumes many cycles when it needs to communicate with other networking elements. Therefore, the utilization of more powerful network processors should improve the communication between NPs and boost the overall performance within the network. A valuable tool to analyze such networks is the queuing network model. The queuing network model can be utilized to derive a model for network processors for packet processing system in a grid-oriented environment. In this model, it is important to be able to determine how to best allocate the arrival rate (bandwidth) in such a manner as to optimize various performance measures, such as the response time and the number of items in the network. In this paper, we propose a solution to optimize the arrival rate allocation between network processing elements to minimize their response time. The solution utilizes queuing network models and an optimal capacity allocation concept. Subsequently, we derive a formula to optimally allocate the arrival rate between NPs. Using this formula, the optimal arrival rate for different NPs can be evaluated to optimize response times. Subsequently, the solution is applied to a grid-oriented network processor model. The results show that the optimal arrival allocation enhances the response time when compared to the same NP-based architecture without optimal arrival allocation.

This paper is organized as follows. Section 2 presents related work. Section 3 describes a summary of queuing network models and the Jackson theorem. Section 4 describes our NP-based architecture model and optimal arrival

rate allocation solution. Section 5 presents simulation results of the solution. In Section 6, we draw the overall conclusions.

2 Related work

In this section, we take a brief look at previous work regarding optimal capacity allocation and performance modeling using queuing network models in the network processing field. In [8], the optimal capacity allocation is considered in a clustered web system environment. It formulates the problem as a nonlinear program to minimize a convex separable function of the capacity assignment vector. The solution can be applied in e-commerce service environment that involves multiple clusters of machines and each cluster handles a particular set of functions. An approximation method to solve the problem was developed. In [10], the assignment of the service capacity in a queuing network is considered. The author studies systems with several types of incoming items, general service time distributions, stochastic or deterministic routing, and a variety of service regimes. The residual-life approximation technique for the distribution of queuing times was utilized. In [7][11], analytical modeling using mean value analysis (MVA) has been used in shared memory multi-processor systems. This technique is shown to be efficient and reasonably accurate for large systems. It used the closed queuing model and an MVA analysis. In [9], J. Lu, et. al., proposed a performance analysis of network processor-based application design using the closed queuing model and an MVA algorithm. This approach investigated the internal network processor behavior at component level as a closed model for specific application similar to traditional processor systems. In our work, we utilize open queuing network models, the optimal arrival rate allocation approach is extracted and is applied for bandwidth allocation for NP-based architectures in a grid-oriented environment.

3 Queuing Models and Jackson Theorem

In this section, we briefly present the concepts of network of queues, and the Jackson network model.

3.1 Networks of Queues and Jackson Theorem

Queuing network analysis is a valuable tool in determining the performance and operating characteristics of real-world networked systems. A queuing network is a collection of two or more nodes where items are being serviced. Items arriving at the network request service from one or more of the nodes and then may leave the network [3]. A fundamental and simple characteristic of queuing networks is whether they are open or closed. An open network allows items to enter and leave the network. In a closed network, items are "trapped" and circulate among the various nodes in the network. A Jackson network consists of M nodes that satisfy the following conditions:

1. Each node consists of c_i identical exponential servers where the service rate of the i_{th} node is μ_i.
2. Items arrive from outside the system to the i_{th} node according to a Poisson process with rate s_i. Items may also arrive from other nodes within the network.
3. Items from node i are routed to node j with probability p_{ij} or leave the network with probability $1-\sum_{j=1}^{M} p_{ij}$.

The arrival rate λ_i to each node i from all sources (external and internal) is

$$\lambda_i = s_i + \sum_{j=1}^{M} p_{ji}\lambda_j, i = 1, ..., M \quad (1)$$

In this equation, s_i is the external arrival rate in each node, p_{ji} is the routing probability between node j and i, λ_j is arrival rate to node j. For each network, we have M arrival equations and these equations form a linear system that can be solved. For networks that satisfy the above conditions, Jackson proved that nodes can be treated as a $M/M/c_i$ queuing model with arrival rate λ_i and service rate μ_i. In the Jackson network [5][12], important parameters are: the mean number of items (mean queue length) and mean resident time in network (response time). The mean number of items in each node i (N_i) with utilization ρ_i is: $N_i = \frac{\rho_i}{1-\rho_i}$. Therefore, the total mean number of items in the network is:

$$\bar{N} = \sum_{i=1}^{M} N_i = \sum_{i=1}^{M} \frac{\rho_i}{1-\rho_i} \quad (2)$$

The mean resident time (response time) in the network of an item is:

$$T_s = \frac{\bar{N}}{\lambda} = \frac{1}{\lambda}\sum_{i=1}^{M} \frac{\rho_i}{1-\rho_i} = \frac{1}{\lambda}\sum_{i=1}^{M} \frac{\lambda_i}{\mu_i - \lambda_i} \quad (3)$$

In the Jackson network, we assume to have control over the service rates $\mu_1, ..., \mu_M$ with the constraint that the total service capability is fixed to a constant value c as follows: $c = \sum_{i=1}^{M} \mu_i$. For a given set of arrival rates λ_i, the optimal set μ_i that minimizes the average number of items in the network $\bar{N} = \sum_{i=1}^{M} N_i$ is [5]:

$$\mu_i = \lambda_i + \frac{\sqrt{\lambda_i}}{\sum_{i=1}^{M}\sqrt{\lambda_i}}(c - \sum_{j=1}^{M}\lambda_j) \quad (4)$$

4 NP-Based Architecture Model and Optimal Arrival Rate Allocation

In this section, we present the simple abstract NP model and an NP-based architecture model. Subsequently, we present the optimal arrival rate allocation concept.

4.1 Simple Abstract NP Model

The handling of incoming packets by a network processor can be separated into two planes, i.e., the data plane and the control plane, that differ in speed and the manner in which packets are handled. In the data plane, simple and highly repetitive tasks are performed. Most packets pass through this high-speed plane of an NP. In the control plane, exceptional packets and complex routines are handled. This model is depicted in Figure 1(A).

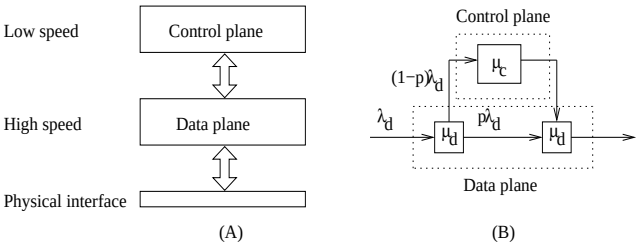

Figure 1. (A) Simple abstract NP model. (B) Simple abstract NP queuing model.

Based on the abstract NP model, we can derive a queuing model with the mapping of each plane on a separate queue. The related queuing model is depicted in Figure 1(B). We call this model the Abstract NP Queuing (ANPQ) model. In this figure, the λ_d and μ_d are the arrival rate and the service rate in the data plane, respectively, and λ_c and μ_c are the arrival rate and the service rate in the control plane, respectively, and λ is the arrival rate of the overal system. Using Eq. 3, the response time T_s in the ANPQ model is:

$$T_s = \frac{1}{\lambda}\left(\frac{\lambda_d}{\mu_d - \lambda_d} + \frac{p\lambda_d}{\mu_d - p\lambda_d} + \frac{(1-p)\lambda_d}{\mu_c - (1-p)\lambda_d}\right) = \left(\frac{1}{(\mu_d - \lambda_d)} + \frac{p}{(\mu_d - p\lambda_d)} + \frac{(1-p)}{(\mu_C - (1-p)\lambda_d)}\right) \quad (5)$$

4.2 Model Overview of Grid-oriented NP Network

In this section, the grid-oriented NP architecture network model is presented. This model is not for specific NPs or their internal components such as buses and memories. This model investigates the role of NPs as processing elements to process incoming packets in a grid computing environment. In this model, one of the NPs operates as master-NP and others cooperate as slave-NPs. When the master-NP's load is saturated, it requests cooperation from other NPs that have a low load. After finding an NP as a slave, the master-NP defers part of data packets to it for processing. The functions of a master-NP include platform configuration and reconfiguration, load balancing, packet processing, scheduling, management, and accounting of the slave-NPs. In this platform, each NP can operate as a slave or a master at different times, it depends on the condition of the NPs. The master-NP segregates input packets between slave-NPs if it needs more processing power. When some packets cannot be handled by slave-NPs, these will be forwarded back to the master-NPs. The master-NP sends the packet stream to slave-NPs using direct path and receives control and non-handled packets from slave-NPs. In the modeling, the behavior of slave-NPs is evaluated based on the master-NP, therefore, the master-NP is represented using the ANPQ model with slave-NPs as simple processing elements.

4.2.1 Model Analysis of Network of NPs in Grid-oriented Environment

In grid computing, a large pool of heterogeneous computing resources is geographically dispersed over a large network,e.g., the Internet, collaborating in solving a single, large, and mostly complex scientific problem. The important part of network processing and grid computing resources are mainly routers and switches. These hardware resources are comprised mainly of network elements that are called network processors (NPs). The NPs in this platform are spread in different network environments and locations. The general architecture is depicted in Figure 2.

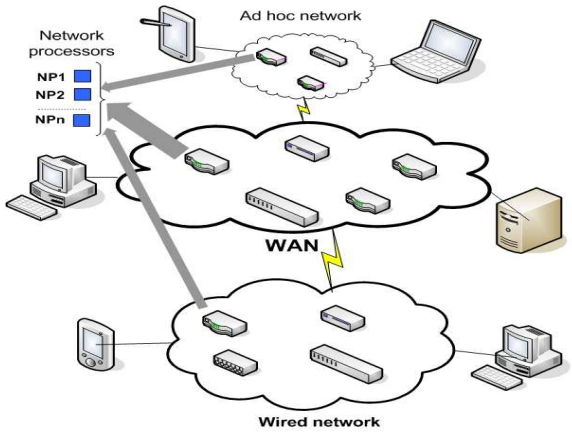

Figure 2. NPs distribution in grid environment.

In this architecture, the cooperative processing is the main ambition where the one of NPs can operate as master-NP and others can cooperate as slave-NPs, the concept is depicted in Figure 3.

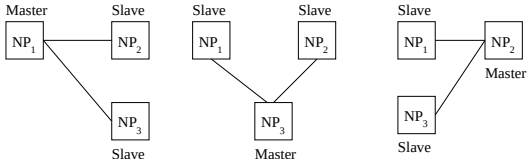

Figure 3. Different NPs configuration.

The NP architecture in grid environments is depicted in Figure 4. In this figure, NP1 receives a packet stream S_1, subdivides it to other streams, and sends those to slave-NPs.

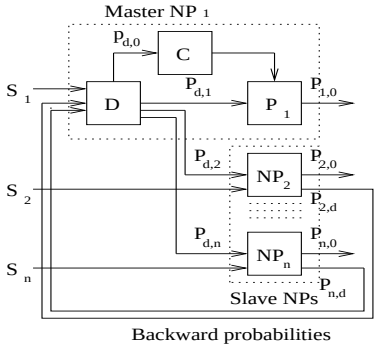

Figure 4. Model of a grid-oriented NP.

The model comprises two parts: a master-NP and a set of slave-NPs. The master-NP includes two data plane processing units D and P1 and a control plane processing unit C (based on the ANPQ model). We can observe that the data plane processing unit receives the packet stream S_1 and divides it between different slave-NPs. In this figure, the S_i (with $i \geq 2$) represents external arrival rate to different slave-NPs, P_{di} (with $i \geq 1$) represents the probabilities of internal arrival rate between master-NP and slave-NPs and called the *forward routing probability*. P_{id} (with $i \geq 2$) represents the probabilities of internal arrival rate among slave-NPs and master-NP and called the *backward routing probability*. The values of the backward routing probabilities is zero when the slave-NPs can handle all packet streams. P_{i0} represents the probability of the outgoing stream for each NP, λ_i and μ_i represent the arrival rate and service rate for different slave-NPs, respectively, λ_d represents the arrival rate for the data plane processing unit of master-NP, λ_c represents the arrival rate for control plane processing unit of master-NP, μ_d represents service rate for data plane processing unit, μ_c represents service rate for control plane processing unit, and μ_i represents service rate for different slave-NPs. Using Eq. 1, we can write the arrival rate equations for different NPs in Figure 4 as following:

$$\begin{aligned} &\lambda_d = \frac{s_1 + \sum_{i=2}^{n} s_i p_{id}}{1 - \sum_{i=2}^{n} p_{di} p_{id}} \quad \lambda_1 = p_{d1}\lambda_d \\ &\lambda_i = s_i + p_{di}\lambda_d, i = 2...n \quad \lambda_c = p_{d0}\lambda_d \end{aligned} \tag{6}$$

Using the Eq. 3, the mean response time for our model can be determined as follows:

$$\begin{aligned} &T_s = \frac{1}{\lambda}\left(\frac{\lambda_d}{\mu_d - \lambda_d} + \frac{\lambda_c}{\mu_c - \lambda_c} + \sum_{i=1}^{n} \frac{\lambda_i}{\mu_i - \lambda_i}\right) = \\ &\frac{1}{\lambda}\left(\frac{\lambda_d}{\mu_d - \lambda_d} + \frac{p_{d0}\lambda_d}{\mu_c - p_{d0}\lambda_d} + \frac{p_{d1}\lambda_d}{\mu_d - p_{d_1}\lambda_d} + \right. \\ &\left. \sum_{i=2}^{n} \frac{s_i + p_{di}\lambda_d}{\mu_i - s_i - p_{di}\lambda_d}\right) \end{aligned} \tag{7}$$

In Eq. 7, we can observe that for some values the denominator of different terms can be zero therefore, the response time T_s will be increased. Since some NPs are busy and can not handle the incoming packets. Consequently, the slave-NPs with acceptable load and response time should be selected. Using Eqs. 6 and 7, the model response time equation is extended as follows:

$$\begin{aligned} &T_s = \frac{1}{\lambda}\left(\frac{C}{\mu_d - C} + \frac{p_{d0}C}{\mu_c - p_{d0}C} + \frac{p_{d1}C}{\mu_d - p_{d_1}C} + \right. \\ &\left. \sum_{i=2}^{n} \frac{s_i + p_{di}C}{\mu_i - s_i - p_{di}C}\right) \quad with\ C = \frac{s_1 + \sum_{i=2}^{n} s_i p_{id}}{1 - \sum_{i=2}^{n} p_{di} p_{id}} \end{aligned} \tag{8}$$

In the Eq. 8, λ is the entire system arrival rate and equal to $\sum_{i=1}^{n} s_i$.

4.3 Optimal Arrival Rate Allocation

Based on the model described in previous section, we can observe that the value of the arrival rate for each slave-NP is determined by the master-NP, but how are the optimal arrival rates determined? In other words, we should find the answers for these questions: How is the value of forward routing probability p_{di} determined? Which slave-NPs can decrease/increase the response time? If an slave-NP increases overall system response time how can this problem be overcome? Therefore, we utilize an optimal arrival allocation mechanism and find a sequence of slave-NPs to minimize system response time. Afterwards, we can use the proportional allocation to distribute the incoming items between different slave-NPs. An alternative to find the optimal arrival rate in the system is the utilization of optimal capacity allocation policy that is presented in Eq. 4. To derive the Eq. 4, we assumed that the value of arrival rates are constant and the optimal service rate has been evaluated. In here we derive a new formula to estimate the optimal arrival rates. We assume supposed that we have control over the arrival rate λ_1, λ_2, ..., λ_M which λ_i respectively is the arrival rate for different slave-NPs. The slave-NPs are managed and controlled by the master-NP, but with a constraint

that fixes the total arrival capability to a constant value c (due to the standard communication line bandwidth) as follows: $\sum_{i=1}^{M} \lambda_i = c$. For a given set of service rates μ_i, we want to find the optimal set λi that minimizes the items $\bar{N} = \sum_{i=1}^{M} N_i$, where $\bar{N}$ represents the mean number of items or queue length that can be computed using Eq. 2. Therefore, we can derive the following equation:

$$\bar{N} = \sum_{i=1}^{M} \frac{\lambda_i}{\mu_i - \lambda_i} \; with \; constraint \; \sum_{i=1}^{M} \lambda_i = c \quad (9)$$

An alternative for minimizing Eq. 9 is to utilize of Lagrangian multiplier. In mathematical optimization problems, Lagrange multipliers is a method to find the local extremum of a function of several variables subject to one or more constraints. This method reduces a problem in n variables with k constraints to a solvable problem in $n + k$ variables with no constraints[6]. Using the method of Lagrangian multipliers, Eq. 9 is rewriten as follows:

$$H = \sum_{i=1}^{M} \frac{\lambda_i}{\mu_i - \lambda_i} + x(\sum_{i=1}^{M} \lambda_i - c) \quad (10)$$

To minimize H, we differentiate and obtain the following equation:

$$\frac{\partial H}{\partial \lambda_i} = \sum_{i=1}^{M} \frac{\mu_i}{(\mu_i - \lambda_i)^2} - Mx \quad (11)$$

If we set the derivative to zero then we find that H is minimized by $\lambda_i = \mu_i - \sqrt{\frac{\mu_i}{x}}$ substituting this expression for λ_i into $\sum_{i=1}^{M} \lambda_i = c$, we find that $\frac{1}{\sqrt{x}} = \frac{\sum_{i=1}^{M} \mu_i - c}{\sum_{i=1}^{M} \sqrt{\mu_i}}$, hence the optimal value for arrival rate obtained by substituting x into the optimal value for λ_i and finally, we have:

$$\lambda_i = \mu_i - \sqrt{\mu_i}(\frac{\sum_{j=1}^{M} \mu_j - c}{\sum_{i=1}^{M} \sqrt{\mu_i}}) \quad (12)$$

The description of this concept is depicted in Figure 5.

In this figure, the curves A and B represent service and arrival rates of different NPs, respectively. Curve C represents the optimal arrival rate for these NPs, where they are estimated using Eq. 12. We can observe that the mapping of curves B and C in the same space generates different areas called overload and underload areas containing NPs. The overload area represents saturated NPs, since the arrival rate values are more than optimal arrival rates and increase response time. The underload area represents NPs that can receive more items and they can be injected to the system as slave-NPs. In the underload area, the arrival rate values are lower than the optimal arrival rates. Based on the Figure 5, the areas E and G represent the underload area and

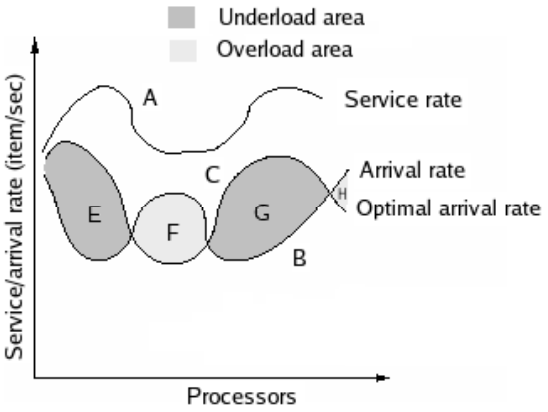

Figure 5. Optimal arrival rate allocation.

areas F and H represent overload area. The value of the arrival rate in the underload areas can increase to the optimal arrival rate values. The total arrival rate that can be handled by NPs in the underload areas is computed as follows:

$$\begin{aligned} &The\ total\ arrival\ rate = \\ &summation\ of\ arrival\ rates\ in\ underload\ area\ 1 + ... + \\ &summation\ of\ arrival\ rates\ in\ underload\ area\ k \\ &= \sum_{j=1}^{k} \sum_{i=1}^{s_j} \left(\mu_{(i,j)} - \lambda_{(i,j)}\right) \end{aligned} \quad (13)$$

Where, k represents the number of underload areas and s_j represents the number of NPs in underload area j.

The minimum response time is achieved when the arrival rate values in different NPs are less than the optimal arrival rate values for same NPs. Therefore, we can write the optimal arrival rate for the system as follows:

$$The\ total\ optimal\ arrival\ rate = \sum_{j=1}^{k} \sum_{i=1}^{s_j} \left(\lambda_{(i,j)opt} - \lambda_{(i,j)}\right) \quad (14)$$

Using the optimal arrival rate allocation, we utilize the proportional allocation to distribute the incoming items between slave-NPs and estimation of the value of p_{id} in the model.

$$p_{di} = \frac{v_i}{\sum_{i=1}^{i=n} v_i} \quad (15)$$

In the proportional allocation, each slave-NP has 'vacant' capacity (vacant capacity is computed by decrementing the arrival rate from optimal arrival rate for each NP and is represented by v_i), the value of p_{di} is estimated from the unallocated capacity divided by the summation of all unallocated capacities for all the slave-NPs.

5 Simulation Results

In this section, we present the simulation results of the proposed approach. The simulation results have been generated using Maple v.10.0. Based on the general queuing

model, the following assumptions have been made to derive a model in grid-oriented environment:

1. Each NP is analyzed by (M/M/1) or (M/M/c) queuing model where the incoming packets obeys the Poisson distribution. Additionally, the service time distribution is exponential.

2. The inter-processor communication delay has been ignored in the modeling phase because the communication line bandwidth is high. The related delay may be added after the response time estimation.

3. In many cases, the average service rate is much greater than the average arrival rate, in this case the waiting queue would not grow too long. If the input buffer is reasonably large dropping of packets is not an issue.

In this investigation, we assume a large pool of available NPs (64 in this experiment) each having a random service and arrival rate to 'mimic' reality as if they were are already in operation. Furthermore, out of this pool, the master-NP is allowed to choose up to 32 NPs as slave-NPs to assist itself in the processing of incoming packets. This investigation is not intended to be realistic for current-day processors, but instead we are trying to determine out of two selection mechanisms which one is able to improve the minimum response time. The model is used as a vessel to achieve this determination as we expect its importance will become evident when increasingly more processor cores will fit on future chips. The first selection mechanism is the first-in and first-out (FIFO) mechanism that list all possible slave NPs and deciding which one to use is solely based on which NP was entered first in the list. This is a simple selection mechanism that should be easy to implement, but will have some adverse effect on the minimum response time. The second selection mechanism is optimal arrival allocation mechanism that chooses a sequence of slave-NPs to assist the master NP. This mechanism is more complex as it requires almost continuous monitoring, but it is expected to yield better results. The FIFO mechanism behavior is depicted in Figure 6. Figure 6(A), depicts the service and arrival rates for different NPs based on the FIFO mechanism. The response time for NP-based architecture model in a grid-oriented environment is depicted in Figures 6(B). From this figure, we can observe that, the effect of overload NPs in response time, since when the number of NPs is increased up to 32, the response time is not better than when the number of NP is 1. Therefore, the overload areas from Figure 6(A) are omitted using optimal arrival rate allocation. The result is depicted in Figure 7(A). The related response time for NP-based architecture in grid-oriented environment using optimal arrival allocation is depicted in Figure 7(B). From this figure, we can observe that the NPs in underload areas that their arrival rates are lower than optimal arrival rates resulting in better response time.

6 Conclusion

In this paper, we proposed an abstract model for network processor using queueing networks (ANPQ) and open queues. Based on the ANPQ, we described the NP-based architecture model in a grid-oriented network environment using the Jackson model. In network processing environments, an important factor is to minimize the response time. Therefore, we presented an approach to optimize the rate arrival allocation. In our approach, we derive a formula that proposed a solution to select a sequence of NPs for packet processing so that the system response time to be minimized. It also removes saturated NPs from NP pool. The presented results show that the utilization of the optimal arrival allocation decreases the response time and number of residence items in the system.

References

[1] I. Adan and J. Resing. "Queuing Theory". http://www.cs.duke.edu/ fishhai/ misc/ queue.pdf, February 2001.

[2] M. Ahmadi and S. Wong. "Network Processors: Challenges and Trends". In *Proc. of the 17th Annual Workshop on Circuits, Systems and Signal Processing, ProRisc 2006*, pages 222–232, November 2006.

[3] R. O. Baldwin, N. J. Davis, S. F. Midkiff, and J. E. Kobza. "Queueing Network Analysis: Concepts, Terminology, and Methods". *J. of Systems and Software*, 66(2):99–117, December 2003.

[4] M. A. Franklin and T. Wolf. "A Network Processor Performance and Design Model with Benchmark Parameterization". In *Proc. of Network Processor Workshop in conjunction with Eighth Int. Symp. on High Performance Computer Architecture (HPCA-8)*, pages 63–74, Cambridge, MA, February 2002.

[5] P. G. Harrison and N. M. Patel. *"Performance Modelling of Communication Networks and Computer Architectures"*. Addison-Wesley Longman, 1st edition, 1992.

[6] D. Klien. "Lagrange Multipliers Without Permanent Scarring". http://www.cs.berkeley.edu/ klein/papers/lagrange-multipliers.pdf, August 2004.

[7] E. D. Lazowska, M. K. Vernon, and J. Zahojan. "An Accurate and Efficient Performance Analysis Technique for Multiprocessor Snooping Cache-Consistency Protocols". In *Proc. of the 15th Annual Int. Symp. on Computer architecture*, pages 308–315, May 1988.

[8] W. Lin, Z. Liu, C. H. Xia, and L. Zhang. "Optimal Capacity Allocation for Web systems with End-to-end Delay Guarantees". *Journal Performance Evaluation*, 62(1):400–416, October 2005.

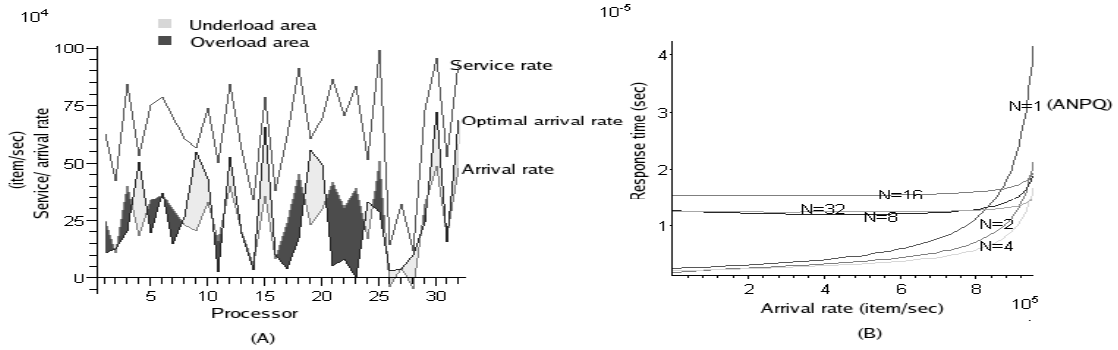

Figure 6. (A) Arrival/service rates and optimal arrival rate curves for different NPs. (B) Grid-oriented NP-based architectures model response time without optimal arrival rate allocation (N shows the number of NPs).

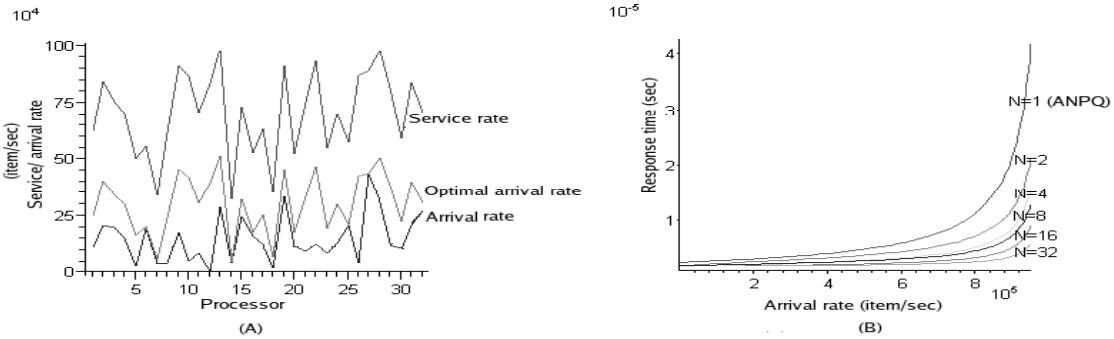

Figure 7. (A) Arrival/service rate and optimal arrival rates curves with optimal arrival rate allocation for different NPs. (B) Grid-oriented NP-based architectures model response time with optimal arrival rate allocation (N shows the number of NPs).

[9] J. Lu and J. Wang. "Analytical Performance Analysis of Network Processor-Based Application Design". In *Proc. of Intl. Conf. on Computer Communications and Networks*, pages 78–86, October 2006.

[10] P. K. Pollett. "Resource Allocation in General Queueing Networks with Applications to Data Networks". In *Proc. of the 16th National Conference of the Australian Society for Operations Research*, September 2001.

[11] T. Tsuei and W. Yamamoto. "A Processor Queuing Simulation Model for Multiprocessor System Performance Analysis". In *Proc. of 5th Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW)*, pages 58–64, 2002.

[12] J. Virtamo. "Queueing Course, Complete Lecture Notes". http://www.netlab.hut.fi/ opetus/ s383143/ kalvot/ english.shtml, 2005.