# QoS Guaranteed Input Queued Scheduling Algorithms with Low Delay[1]

Jinhui Li and Nirwan Ansari
Advanced Networking Lab.
Electrical and Computer Engineering Department
New Jersey Institute of Technology
University Heights
Newark, NJ 07102, USA

*Abstract* — **The enhanced Birkhoff-von Neumann decomposition (EBVND) algorithm, a new class of scheduling algorithms for input queued (IQ) switches, is introduced. Theoretical analysis shows that the performance of EBVND is better than the Birkhoff-von Neumann decomposition algorithm in terms of throughput and cell delay, and can also provide rate and cell delay guarantees. Wave front Birkhoff-von Neumann decomposition (WFBVND) algorithm and its simplified version WFBVND with $logN$ iterations (WFBVND-$logN$), the special cases of EBVND, are also introduced and evaluated. Simulations show that WFBVND and WFBVND-$logN$ have much lower average cell delay as compared to the Birkhoff-von Neumann decomposition algorithm.**

## I. Introduction

The input-queued (IQ) switching architecture has been adopted for high speed switch implementation owing to its scalability. Employing virtual output queueing (VOQ)[1][2], in which multiple VOQs directed to different outputs are maintained at each input, IQ switches without speedup can avoid the head-of-line (HOL) blocking which limits the throughput of the IQ switch using a single FIFO queue in each input to approximately 58.6% [3]. Simulations show that IQ switches with VOQs using wrapped wave front arbiter (WWFA) [4] or parallel iterative matching (PIM) algorithm [2] can reach an asymptotic 100% throughput without the restriction of HOL blocking. Later, theoretical analysis shows that the longest queue first (LQF) [5] and the oldest cell first (OCF) [5] algorithms can achieve 100% throughput under all admissible and independent arrival processes. Though high throughput can be reached, none of the above algorithms can provide delay bound. Recently, Chang *et al.* proposed [6] the Birkhoff-von Neumann decomposition algorithm, which can provide rate and cell delay guarantees, based on a decomposition result by Birkhoff and von Neumann for a doubly stochastic matrix.

## II. Enhanced Birkhoff-von Neumann Decomposition Algorithm

Consider an $N \times N$ input-queued switch consisting of $N$ inputs, $N$ outputs, and a non-blocking switch fabric such as crossbar. The packets, which may have variable length, are broken into fixed length cells when they arrive in the inputs. After the cells crossed the fabric, they are reassembled to the original variable length packets. Time slot is defined as the time required transmitting a cell at the line rate. The VOQ directed to output $j$ at input $i$ is denoted by $Q_{i,j}$. If $Q_{i,j}$ is not empty, there will have a request from input $i$ to output $j$. The basic objective of scheduling an IQ switch is to find a contention free match based on the connection requests, i.e., at most one input can be matched to each output, and vice versa.

Denote $r_{i,j}$ as the arrival rate of VOQ $Q_{i,j}$. The input traffic is said to be admissible if the following inequalities are satisfied:

$$\sum_{j=0}^{N-1} r_{i,j} \leq 1, \forall i, \tag{1}$$

$$\sum_{i=0}^{N-1} r_{i,j} \leq 1, \forall j. \tag{2}$$

The matrix $\mathbf{R} = (r_{i,j})$ satisfying Eq. (1) and (2) is said to be doubly substochastic. For any doubly substochastic matrix $\mathbf{R}$, there exists [6] a doubly stochastic matrix $\tilde{\mathbf{R}} = (\tilde{r}_{i,j})$ such that $r_{i,j} \leq \tilde{r}_{i,j}, \forall i, j$. An algorithm to construct doubly stochastic matrix $\tilde{\mathbf{R}}$ from doubly substochastic matrix $\mathbf{R}$ is also provided in [6] with a computational complexity of $O(N^3)$. In this paper, we introduce the weighted rate filling algorithm (WRFA) to perform this task:

**Weighted Rate Filling Algorithm (WRFA)**

1. *Define* $p_i = 1 - \sum_{j=0}^{N-1} r_{i,j}$. *Calculate* $p_i$ *for all* $i$.
2. *Define* $q_j = 1 - \sum_{i=0}^{N-1} r_{i,j}$. *Calculate* $q_j$ *for all* $j$.
3. *Calculate* $\Delta = N - \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} r_{i,j}$.
4. *Let* $\tilde{r}_{i,j} = r_{i,j} + \frac{p_i q_j}{\Delta}$.

The complexity of WRFA is $O(N^2)$ which is smaller than the original one; furthermore, WRFA can share the unreserved rate more fairly among the VOQs. For example, when $\mathbf{R} = \begin{bmatrix} 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.2 \end{bmatrix}$, the original algorithm constructs the doubly stochastic matrix $\tilde{\mathbf{R}} = \begin{bmatrix} 0.4 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.4 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.4 & 0.2 \\ 0.2 & 0.2 & 0.2 & 0.4 \end{bmatrix}$.

[1] This work has been supported in part by Lucent Technologies, the New Jersey Commission on Science and Technology via the New Jersey Center for Wireless Telecommunications, and the New Jersey Commission on Higher Education via the NJI-TOWER Project.

Using WRFA, we get $\tilde{\mathbf{R}} = \begin{bmatrix} 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.25 & 0.25 & 0.25 & 0.25 \end{bmatrix}$. Apparently, WRFA shares the unreserved rate among the VOQs more fairly.

A doubly stochastic matrix $\tilde{\mathbf{R}}$ can be expressed as the linear combination of permutation matrices [6], $\tilde{\mathbf{R}} = \sum_k \phi_k \mathbf{P}_k$, where $\mathbf{P}_k$ is a permutation matrix, and $0 < \phi_k \leq 1$ such that $\sum_k \phi_k = 1$.

The Birkhoff-von Neumann decomposition algorithm schedules the cells by setting the connection of the crossbar according to the permutation matrix $\mathbf{P}_k$ with probability $\phi_k$ [6]. Let $C_{i,j}(n)$ be the cumulative number of time slots for transmission that are assigned to $Q_{i,j}$ by time slot $n$. Denote $A_{i,j}(n)$ as the total number of cells arrived in $Q_{i,j}$ at the end of time slot $n$. Then the Birkhoff-von Neumann decomposition algorithm can guarantee

$$C_{i,j}(m) - C_{i,j}(n) \geq (m-n)r_{i,j} - u_{i,j}, \qquad (3)$$

for all $i$, $j$, $m > n$, where $u_{i,j}$ is a real number less than or equal to $N^2 - 2N + 2$, if Eq. (1) and (2) are satisfied [6]. Eq. (3) implies that if $A_{i,j}(n)$ conforms to $(\sigma_{i,j}, r_{i,j})$, i.e.,

$$A_{i,j}(m) - A_{i,j}(n) \leq (m-n)r_{i,j} + \sigma_{i,j}, \qquad (4)$$

then the cell delay from input $i$ to output $j$ is bounded by $\lceil(\sigma_{i,j} + u_{i,j})/r_{i,j}\rceil$ using the Birkhoff-von Neumann decomposition algorithm [6]. The off-line and on-line computational complexity of this algorithm is $O(N^{4.5})$ and $O(logN)$, respectively [6].

We observed that the Birkhoff-von Neumann decomposition algorithm is not efficient enough because it pays no attention to the current occupancy of VOQs. For example, the connection requests at the current time slot is shown in Fig 1(a), where the non-empty VOQs are filled with crosses. Suppose the non-zero elements of the permutation matrix selected by the Birkhoff-von Neumann decomposition algorithm at the current time slot are represented by circles in Fig. 1(a). Among all the VOQs selected by $\mathbf{P}$, only VOQ $Q_{0,1}$ is non-empty which is shown by thick border box in Fig. 1(a). Hence, the Birkhoff-von Neumann decomposition algorithm can schedule only one cell in current time slot. However, two more VOQs, such as $Q_{1,2}$ and $Q_{2,3}$, can actually send cells across the fabric in the current time slot without removing the cells scheduled by the Birkhoff-von Neumann decomposition algorithm.

Based on the above observation, the enhanced Birkhoff-von Neumann decomposition (EBVND) algorithm matches the inputs and outputs which are not matched by the Birkhoff-von Neumann decomposition algorithm, and attempts to make a maximal match in every time slot. Many algorithms such as WWFA and PIM can be used to fill the "holes" left by the Birkhoff-von Neumann decomposition algorithm. Thus, EBVND will have a higher on-line computational complexity than the Birkhoff-von Neumann decomposition algorithm, but expects to have better performance.

Suppose IQ switch $B$ uses the Birkhoff-von Neumann decomposition algorithm while IQ switch $E$ uses EBVND. Denote $Q^B_{i,j}$ and $Q^E_{i,j}$ as the $Q_{i,j}$ of switch $B$ and $E$, respectively. Denote $L^E_{i,j}(n)$ and $L^B_{i,j}(n)$ be the length of $Q^E_{i,j}$ and $Q^B_{i,j}$ by the end of time slot $n$, respectively. Let $T^B_{i,j}(n)$ be the cumulative number of cells dequeued from $Q^B_{i,j}$ by the end of time slot $n$, and $T^E_{i,j}(n)$ be that of $Q^E_{i,j}$. Define $T_{i,j}(n,m) = T_{i,j}(m) - T_{i,j}(n)$. The following can be readily derived:

**Lemma 1** *For any integer $m > n$, if both $Q^B_{i,j}$ and $Q^E_{i,j}$ are constantly backlogged from time slot $n+1$ to time slot $m$, then $T^E_{i,j}(n,m) \geq T^B_{i,j}(n,m)$.*

**Theorem 1** *If the exactly same traffic is fed into switch $B$ and $E$ concurrently and no cell is dropped, then $L^E_{i,j}(n) \leq L^B_{i,j}(n)$ for any $i$, $j$, and $n$.*

Denote $D^B_c$ as the delay of certain cell $c$ in switch $B$, and $D^E_c$ as the delay in switch $E$.

**Theorem 2** *Assume all the VOQs in switch $B$ and $E$ are FIFOs. If the exactly same traffic is fed into switch $B$ and $E$ concurrently and no cell is dropped, then $D^E_c \leq D^B_c$ for any cell $c$.*

Theorems 1 and 2 imply that the performance of EBVND is better than (in the worst case at least as good as) the Birkhoff-von Neumann decomposition algorithm in terms of throughput and cell delay guarantees. Thus, EBVND does provision QoS guarantees since the Birkhoff-von Neumann decomposition algorithm was proven to provision QoS guarantees [6].

Wave front Birkhoff-von Neumann decomposition (WFBVND) algorithm, which is a special case of EBVND, matches the unmatched inputs and outputs using a method similar to WWFA [4]. WFBVND divides a time slot into $N$ phases. Assume $\mathbf{P}$ is the permutation matrix selected by the Birkhoff-von Neumann decomposition algorithm in the current time slot. In the $l$th phase, where $0 \leq l \leq N-1$, WFBVND calculates matrix $\mathbf{V}_l = (V_{l,i,j})$, where $V_{l,i,j} = P_{(i+l)modN,j}$. During the $l$th phase, WFBVND checks the VOQs corresponding to the non-zero elements of $\mathbf{V}_l$, and adds the non-empty VOQs in the match if both its input and output are unmatched. Fig. 1(a)-(d) shows an example, where the VOQs filled with crosses indicate the non-empty VOQs, the VOQs filled with circles indicate that the corresponding elements of $V_l$ are $1'$s, and the VOQs with thick border indicate they are scheduled to transmit cells. The on-line computational complexity of WFBVND is $O(N^2)$.

The complexity of $O(N^2)$ may be costly for high-speed implementation. WFBVND with *logN* iterations (WFBVND-*logN*) is thus introduced as a simplified version of WFBVND in order to reduce the complexity. It differs from WFBVND by only having the first *logN* phases of WFBVND. Since WFBVND-*logN* runs less phases than WFBVND, its performance is expected to be worse than WFBVND, but still be better than the Birkhoff-von Neumann decomposition algorithm, because it is the special case of EBVND. The on-line computational complexity of WFBVND-*logN* is $O(NlogN)$.

## III. Discussion and Simulations

The performance of the new and some other existing algorithms was simulated in a 16 × 16 IQ switch. 256 *i.i.d.* flows, each belonging to a different input-output pair, were created in the simulations. Traffic $A_{i,j}$ conforms to $(\sigma_{i,j}, r_{i,j})$ for all $i, j$, where $\sigma_{i,j}$ is set to be $1000r_{i,j}$. Thus, the delay bound is $1000 + \lceil u_{i,j}/r_{i,j}\rceil$ time slots for the Birkhoff-von Neumann decomposition algorithm, WFBVND, and WFBVND-*logN*.

Fig. 2 shows the average cell delay of the Birkhoff-von Neumann decomposition, WFBVND, WFBVND-*logN*, and

OCF versus the traffic load under *i.i.d.* $(\sigma, r)$ traffic. Fig. 3 shows the variance of cell delay versus the traffic load under the same traffic model. Fig. 2 and 3 indicate that the average cell delay and variance of WFBVND is much smaller than that of the Birkhoff-von Neumann decomposition algorithm, and close to OCF. With a reduced complexity, the average delay and the delay variance of WFBVND-$logN$ are also significantly smaller than those of the Birkhoff-von Neumann decomposition algorithm.

## IV. Conclusions

A new class of IQ scheduling algorithms with rate and delay guarantees has been proposed in this paper. Specifically, the performance of WFBVND and WFBVND-$logN$ is compared with OCF and the Birkhoff-von Neumann decomposition algorithm. It has been demonstrated both by simulations and theoretical analysis that the new algorithms can achieve much smaller average cell delay and delay variance, as well as provide QoS guarantees.

## References

[1] Y. Tamir and G.L. Frazier, "High-performance multi-queue buffers for VLSI communication switches," *Proc. 15th Annu. Int. Symp. Comput. Architecture,* Honolulu, HI, May 1988, pp. 343-354.

[2] T. Anderson, S. Owicki, J. Saxe, and C. Thacker, "High speed switch scheduling for local area networks," *ACM Trans. on Computer Systems,* vol. 11, no. 4, pp. 319-352, Nov. 1993.

[3] M. Karol, M. Hluchyi, and S. Mogan, "Input versus output queueing on a space-division packet switch," *IEEE Trans. Commun.,* vol. 35, no. 12, pp. 1347-1356, Dec. 1987.

[4] Y. Tamir and H.C. Chi "Symmetric crossbar arbiters for VLSI communication switches," *IEEE Transactions on Parallel and Distributed Systems,* vol. 4, no. 1, pp. 13-27, Jan. 1993.

[5] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Transactions on Communications,* vol. 47, no. 8, pp. 1260-1267, Aug. 1999.

[6] C.S. Chang, W.J. Chen, and H.Y. Huang, "On service guarantees for input buffered crossbar switches: a capacity decomposition approach by birkhoff and von neumann," *Proc. IEEE IWQoS'99,* London, U.K., 1999, pp.79-86.

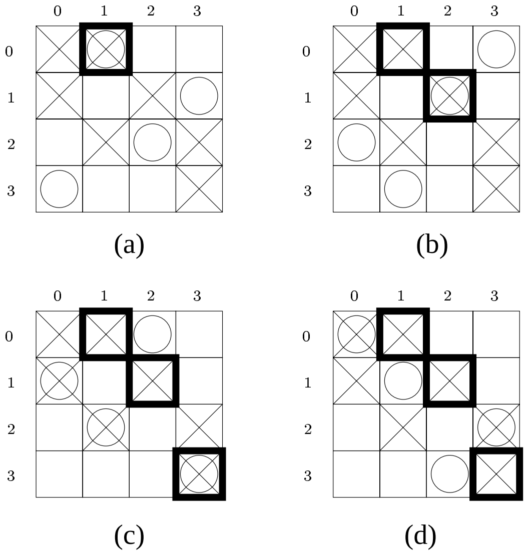

Figure 1: WFBVND algorithm, (a) phase 0, (b) phase 1, (c) phase 2, and (d) phase 3.

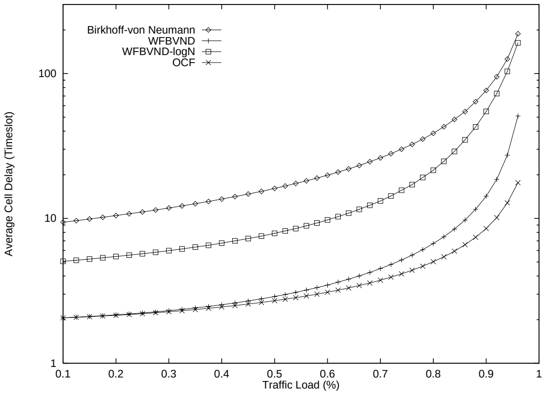

Figure 2: Average cell delay vs. traffic load under *i.i.d.* $(\sigma, r)$ arrival.

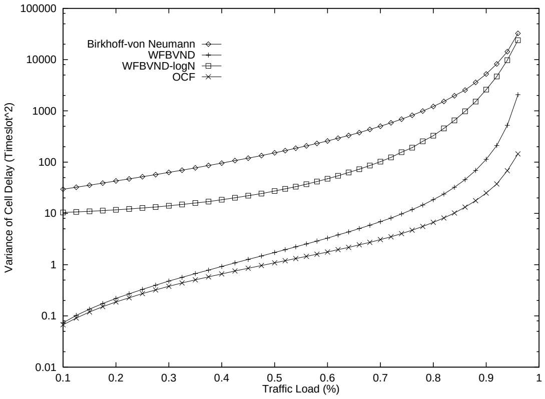

Figure 3: Variance of cell delay vs. traffic load under *i.i.d.* $(\sigma, r)$ arrival.