Scheduling Reserved Traffic in Input-Queued Switches: New Delay Bounds via Probabilistic Techniques

Matthew Andrews and Milan Vojnović

Abstract—We consider the problem of providing delay bounds to reserved traffic in high-speed input-queued switches. We assume that the matrix of bandwidth demands is known and we use the now standard approach of decomposing this matrix into a convex combination of permutation matrices. Our problem therefore reduces to the problem of constructing a schedule for these permutation matrices.

In this paper we derive delay bounds for four algorithms that are based on probabilistic techniques. For each algorithm we first place tokens randomly in continuous time for each permutation matrix. If the $i$th token that appears corresponds to permutation matrix $M_i$, then we schedule matrix $M_i$ in the $i$th time slot. The algorithms differ in how the random token processes are defined. For two of the algorithms we are able to perform a derandomization so as to obtain deterministic schedules.

We show through numerical computation that in many situations the resulting delay bounds are smaller than the previously best-known delay bounds of Chang, Chen, and Huang [1].

Keywords—Input-queued switches, Decomposition-based schedules, Delay bounds.

I. INTRODUCTION

In recent years there has been a great deal of work on scheduling algorithms for input-queued switches. The key feature of an input-queued switch is that at each time step, each input can be connected to at most one output and each output can be connected to at most one input. The aim of the scheduler is to determine how to configure the switch at each time step so as to provide high throughput and low delays for the arriving packets.

Most of the previous work has concentrated on providing stability or 100% throughput for the switch. A scheduler is said to be stable if the queues remain bounded as long as the load on each port is less than the capacity of that port. Algorithms that provide stability generally fall into two categories depending on whether or not we know the arrival rates for each input-output pair in advance. For the case in which we do know the arrival rates we can decompose the rate matrix into a convex combination of permutation matrices. If the scheduler configures the switch according to this decomposition then we have stability. We refer to these schedulers as decomposition-based schedulers (e.g., [1], [2], [3]). Specific algorithms for performing the decomposition can be derived from results of Birkhoff [4] and von Neumann [5]. Note that the schedule can be computed in advance of the arriving traffic and hence it is acceptable for the calculation to have significant complexity.

For the case of high-speed optical switches it is reasonable to assume that the arrival rates are known to us because such switches are likely to be deployed in the core of networks where traffic engineering using MPLS is becoming more prevalent. Switches that support MPLS must be able to provide bandwidth guarantees for certain input-output pairs. For each MPLS path that passes through input $i$ and output $j$ on a switch, the switch will be required to reserve bandwidth for the path between these two ports. Another justification that the input rates are known can be found with Expedited Forwarding (EF) in the context of differentiated services [6]. There it is commonplace to assume the network is engineered such that the load of EF traffic at each node is bounded by some configured value.

For the case in which we do not know the arrival rate matrix in advance then most of the stable schedulers set up a bipartite graph whose edge weights correspond to the queue sizes of the corresponding input queues. At each time step the scheduler finds an (approximate) maximum-weight matching in this graph and configures the switch accordingly. These schedulers are sometimes known as maximum weight matching (MWM) type schedulers (e.g., [7], [8], [9], [10]). Note that an MWM scheduler must operate in real-time since it needs access to queue information and hence it must have extremely low complexity.

More recently, attention has been paid to the problem of minimizing the delay experienced by packets passing through an input-queued switch. Leonardi et al. [11] analyzed the MWM algorithm and showed that the mean delay through an arbitrary pair of input-output ports of a switch with $I$ input and $I$ output ports, uniformly loaded to $\rho < 1$, is bounded by $\frac{(I - \rho)}{(1 - \rho)}$. A related work is that of Shah and Kopike [12] who observe that for uniform Bernoulli arrivals to the switch with the scheduling policy that at each time slot takes a matching uniformly at random from the entire set of $I!$ matchings, the expected delay is $\frac{(I - 1)}{(1 - \rho)}$. Note that this is smaller than the bound obtained in [11].

In [1] Chang, Chen, and Huang showed how to derive worst-case deterministic delay bounds for a Birkhoff-von Neumann schedule in which the permutation matrices are scheduled according to Packetized Generalized Processor Sharing (PGPS) [13] such that at an instant a matrix is served, it is placed as a new arrival into the PGPS system.

In this paper we consider delay bounds for decomposition-based schedules. We show that by using probabilistic techniques we are able to tighten the bounds of [1] for the worst case input-output pairs in many scenarios. We use the term “probabilistic techniques” rather than “randomized algorithms” since for two of our algorithms we are able to derandomize the random processes so as to obtain deterministic algorithms.

It is interesting to observe that for a node to support EF it needs to conform to a rigorous definition of the per-hop-
behavior, namely, Packet Scale Rate Guarantee with rate $r$ and latency $\epsilon$; see [14]. Our work can be viewed as calculating what the value of the latency $\epsilon$ would be for an input-queued switch. Before we present our algorithms and results in detail we must present some notation and our goals.

**Assumptions and Notation**

We consider an $I \times I$ switch. Let $\rho_{ij}$ be the bandwidth that needs to be reserved between input $i$ and output $j$, normalized by the link rate. Let $M$ be the matrix whose $ij$ entry is $\rho_{ij}$. We refer to $M$ as the rate matrix. For the majority of this paper we consider the case in which $M$ is a doubly stochastic matrix, i.e. $\sum_i \rho_{ij} = 1$ and $\sum_j \rho_{ij} = 1$. This corresponds to the case in which the entire bandwidth of the switch is reserved. However, we shall sometimes consider the substochastic case in which we only have $\sum_i \rho_{ij} \leq 1$ and $\sum_j \rho_{ij} \leq 1$. In this case the residual bandwidth of the switch could be used by best effort traffic.

By standard results of Birkhoff and von Neumann (see e.g. [1]) we can decompose the matrix $M$ into a convex combination of permutation matrices,

$$M = \sum_{k=1}^{K} \varphi_k M_k,$$

where $K \leq I^2 - 2I + 2$. Here, $M_k$ is a permutation matrix (a $0-1$ matrix with exactly one “1” in each row and column), $\varphi_k$ is the rate of matrix $M_k$ and $\sum_{k=1}^{K} \varphi_k = 1$. Let $S_{ij}$ be the set of matrices in the decomposition that have a $1$ in the $ij$ position. Then $\rho_{ij} = \sum_{k \in S_{ij}} \varphi_k$. Our aim is to create a schedule in which exactly one of the permutation matrices is scheduled in each time slot. Input-output pair $ij$ is served whenever a matrix from the set $S_{ij}$ is scheduled. Hence we require that a matrix from $S_{ij}$ is scheduled approximately once every $1/\rho_{ij}$ time slots.

As an example, suppose that,

$$M = \begin{pmatrix}
1/6 & 5/6 & 0 \\
1/2 & 1/6 & 1/3 \\
1/3 & 0 & 2/3
\end{pmatrix}.$$ 

Then,

$$M = \frac{1}{2} M_1 + \frac{1}{3} M_2 + \frac{1}{6} M_3,$$

where,

$$M_1 = \begin{pmatrix}
0 & 1 & 0 \\
1 & 0 & 0 \\
0 & 0 & 1
\end{pmatrix}, M_2 = \begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 0 & 0
\end{pmatrix},$$

$$M_3 = \begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{pmatrix},$$

and a possible schedule is,

$$M_1, M_2, M_1, M_2, M_1, M_3, M_1, M_2, M_1, M_2, M_1, M_3, \ldots$$

The class of the schedulers that we consider can be formulated by the following unifying framework. We first place tokens for each matrix $M_k$ in continuous time. We schedule matrix $M_k$ in time slot $n$ if the $n$th token to appear corresponds to matrix $M_k$. (See Figure 1).

More formally, we associate with $M_k$ a counting process $N_k$ defined on $\mathbb{R}^+$. For any interval $I \subseteq \mathbb{R}^+$, $N_k I$ equals the number of tokens for $M_k$ that land in interval $I$. We require that $N_k$ has intensity $\varphi_k$, i.e. $\lim_{t \to \infty} N_k[0,t)/t = \varphi_k$.

We define the superposition process $NL = \sum_{k=1}^{K} N_k I$ to which there is an associated point process $(T_n)_{n \geq 0}$ defined on $\mathbb{R}^+$. Next, let $(Z_n)_{n \geq 0}$ be the sequence of marks such that $Z_n = k$ if and only if the $n$-th point of the superposition point process, $T_n$, belongs to $N_k$. Let $N_{ij} I$ be the number of tokens for input-output pair $ij$ that land in the interval $I$, i.e. $N_{ij} I = \sum_{k \in S_{ij}} N_k I$. Likewise, let $N_{ij} I$ be the number of tokens that land in $I$, but do not belong to $S_{ij}$.

The schedule is given by the sequence $(Z_n)_{n \geq 0}$. If for any fixed $n$, $Z_n = k$, then the matrix $M_k$ is scheduled in the $n$th slot. We say the $n$th token is of type $k$. Notice that by taking $(Z_n)_{n \geq 0}$ we in fact construct a non-idle schedule. A key feature of this schedule is,

**Observation I:** The total number of slots in which input-output pair $ij$ can be served during the time slots $n, n + 1, \ldots, n + m - 1$ is equal to $N_{ij}[T_n, T_{n+m})$.

**Service Characterization**

We give different characterizations of the service offered to an arbitrary input-output pair $ij$. Informally speaking, we would like $N_{ij}[T_n, T_{n+m})$ to be close to $\rho_{ij} m$. The following is the simplest, but weakest, service characterization: for any $n \geq 0$ and $m > 0$, and some fixed $E^{ij}_1 \geq 0$,

$$\{ N_{ij}[T_n, T_{n+m}) \geq \rho_{ij} (m - E^{ij}_1) \}.$$  

If the above event holds with probability $1 - \epsilon$, $\epsilon \geq 0$, a probabilistic interpretation of the service offered is: for any fixed $m$ one picks at random a slot $n$, then, the number of tokens given to the input-output pair $ij$ in the next $m$ slots is at least $\rho_{ij} (m - E^{ij}_1)$ with probability $1 - \epsilon$.

A natural extension of the above characterization is by requiring that for any $n \geq 0$ and some fixed $E^{ij}_2 \geq 0$,

$$\{ \forall m > 0 : N_{ij}[T_n, T_{n+m}) \geq \rho_{ij} (m - E^{ij}_2) \}.$$  

The strongest guarantee is offered by requiring, for some fixed $E^{ij}_3 \geq 0$,

$$\{ \forall n \geq 0 \forall m > 0 : N_{ij}[T_n, T_{n+m}) \geq \rho_{ij} (m - E^{ij}_3) \}.$$
If this event holds then we lower bound the service offered to input-output pair $ij$ over any interval of time slots. In particular, it can be seen that $f(m) = \rho_{ij} \max[m - E_{ij}^3, 0]$ is a strict minimum service curve offered to the $ij$th pair (see Proposition 1.3.6, Section 1.3.2, [15]). The service curve is “rate-latency” with rate $\rho_{ij}$ and latency $E_{ij}^3$.

The service characterizations introduced so far bound how much the service offered is behind the service that would be offered by an idealistic fluid system (which would serve $\rho_{ij} m$ bits in $m$ slots). Thus, these service characterizations bound the lateness of the scheduler. Analogous characterizations can be established to bound the earliness; one only needs to reverse the inequalities in the above definitions, and replace minus with plus in the rate-latency functions. Small earliness of the调度 is desirable to reduce burstiness at the output of the switch.

**What Can We Compute from the Service Characterizations?**

Consider an arbitrary input-output port pair $ij$. Let $A_{ij}[n]$ be number of the bits that arrive in $[0, n]$ at input port $i$ and are destined for output port $j$. Then, by the result known for variable-capacity nodes (see [15], Sec. 1.3.2, also Sec. 4.3.2), we know that the number of bits in $[0, n]$ observed at the output port $j$ that originate from $i$, $A_{ij}^*[n]$, satisfies,

$$A_{ij}^*[n] = \min_{1 \leq m \leq n} [A_{ij}[m] + N_{ij}[T_m, T_n]].$$

In particular, suppose that the arrivals are $(\sigma_{ij}, \rho_{ij})$-bounded, i.e., $A_{ij}[n] - A_{ij}[m] \leq \rho_{ij}(n - m) + \sigma_{ij}$ for all $m \leq n$. Then, the following is a classical network calculus result.

**Fact 2:** The backlog of $ij$ packets waiting for service at the switch is at most $\sigma_{ij} + \rho_{ij}E_{ij}^3$. If FIFO scheduling is used within the aggregate of $ij$ packets then the maximum delay for these packets is at most $\sigma_{ij}/\rho_{ij} + E_{ij}^3$.

If $\sigma_{ij} = 0$ (i.e. the arrivals are bounded by a idealized fluid system of rate $\rho_{ij}$) then the packet delay is bounded by $E_{ij}^3$ (assuming FIFO scheduling within the aggregate). However, in a perfect schedule for $ij$, a matrix in $S_{ij}$ would appear exactly once every $1/\rho_{ij}$ time slots. In this case the packet delay would be $1/\rho_{ij}$. Hence, if $\sigma_{ij} = 0$ we have,

$$\frac{\text{worst case packet delay}}{\text{optimal packet delay}} \leq \frac{\rho_{ij} E_{ij}^3}{\rho_{ij}}.$$

For these reasons our objective is to keep $E_{ij}^3$ small.

**Algorithms and results**

Our algorithms will be divided into two types, frame-based and non-frame-based. Suppose that for some fixed integers $\ell_k$ and $L$, $\varphi_k = \ell_k/L$. (We note that this is always possible if the $\varphi_k$ are rational.) We can compute a schedule for the interval $[0, L]$ that contains exactly $\ell_k$ occurrences of the permutation matrix $m_k$ and then simply repeat this schedule for all subsequent intervals of length $L$. We call such a schedule a frame-based schedule of length $L$. Notice that the frame length $L$ and number of the permutation matrices $K$ are related as $K = L/\ell$, where $\ell$ is the arithmetic mean of $\ell_k$, $k = 1, \ldots, K$. Since $\ell_k \geq 1$ for all $k$ it follows that $L \geq K$, with equality if and only if $\ell_k = 1$ for all $k$.

If the schedule is not periodic in the above way then we say that it is non-frame-based. For a non-frame-based schedule we have to define it explicitly in the entire interval $[0, \infty)$.

In [1], Chang et al. propose a non-frame-based algorithm in which the permutation matrices are scheduled according to a PGPS [13] system that is fed with its own departures (which is initialized such that all tokens arrive at time 0). In our setting, this corresponds to placing the $n$th token for matrix $M_k$ at time $n/\varphi_k$. More formally, for each $k = 1, \ldots, K$, $N_k[0, t] = \sum_{n \geq 0} 1_{(0, t]}(n/\varphi_k)$. Chang et al. [1] show that for this algorithm,

$$E_{ij}^3 \leq \min\left[\frac{K}{\rho_{ij}}, \frac{|S_{ij}|}{\rho_{ij}} + (K - 1)\right].$$

The aim of our work is to show that by using probabilistic techniques, it is possible to tighten this bound in many scenarios. Our results are as follows.

1. In Section III-A with an extremely simple frame-based scheduler in which the tokens for the permutation matrices in a frame are randomly permuted. We call this the Random Permutation scheduler. We require that (3) holds with probability $1 - \varepsilon$ and we show that, as $L \to \infty$,

$$E_{ij}^3 \to A \left(\frac{1}{\rho_{ij}} - 1\right) L,$$

where $A$ is a constant depending on $\varepsilon$ specified in this paper. For $E_{ij}^3$ the same expression holds, with $A = \frac{1}{2} \ln \varepsilon^{-1}$.

2. In Section III-B we present a deterministic frame-based algorithm. We require that (3) holds with probability 1 and we show that,

$$E_{ij}^3 \leq \frac{|S_{ij}|}{\rho_{ij}} + (2 + \sqrt{2K \ln(2L + 1)}).$$

We derive this algorithm from a randomized algorithm in which the $n$th token for matrix $M_k$ is placed at time $U_k + n/\varphi_k$ where $U_k$ is chosen uniformly at random in $[0, 1/\varphi_k)$. We call this the Random-Phase Periodic Competition scheduler. We then show how to derandomize this scheduler to obtain a deterministic algorithm using the method of conditional probabilities [16]. In Section IV we show that in many scenarios, (6) is significantly smaller than (4), largely due to the presence of the square-root in (6).

3. In Section III-C we present a deterministic non-frame-based algorithm. We require that (3) holds with probability 1 and we show that,

$$E_{ij}^3 \leq \frac{1}{\rho_{ij}} \sqrt{2|S_{ij}| \ln D} + (2 + \sqrt{2K \ln D}),$$

where $D = 1 + (4(2I^2 + 2)/\min_k \varphi_k)$. This algorithm is derived from a randomized algorithm in which the $n$th token for matrix $M_k$ is placed uniformly at random in the interval $[0, \infty)$. 

0-7803-7753-2/03/$17.00 (C) 2003 IEEE IEEE INFOCOM 2003
We call this the Random-Distortion Periodic Competition scheduler. By using the method of conditional probabilities we are able to derandomize this scheduler although the analysis is more complex than it was for the random-phase scheduler since we now have to consider the entire interval \([0, \infty)\) instead of a finite frame. In Section IV we show that in many scenarios, (7) is significantly smaller than (4), largely due to the presence of the square-roots in (7).

4. In Section III-D we analyze a non-frame-based scheduler in which the tokens for matrix \(M_k\) are placed according to a Poisson process. We call this the Poisson Competition scheduler. For this scheduler only we assume that the load \(\rho\) on each input and output is strictly less than 1. We show using the Brownian approximation [17] that

\[
E^{ij}_2 \approx \frac{1}{2} \ln \epsilon^{-1} \frac{1 - \rho}{1 - \rho} \left( \frac{1 - \rho}{\rho_{ij}} + \rho \right).
\]

The latency \(E^{ij}_2\) does not make much sense for this scheduler since the event in (3) would fail with probability 1 as we require that the inequality in (3) holds for all \(n\).

Comparison with single server polling

We remark that our problem is significantly different from the single server polling problem (e.g. see [18] and references therein) in which a single server has to poll a set of clients at predetermined times. Note that in our problem it is not sufficient for each matrix \(M_k\) to be served at evenly spaced intervals of \(1/\varphi_k\) slots. This is because input-output pair \(ij\) is served whenever a matrix in \(S_{ij}\) is served. If \(k, \ell \in S_{ij}\) and \(M_k\) and \(M_\ell\) are served close together then the service to \(ij\) is bursty even though each permutation matrix might receive smooth service. Note however that we cannot in isolation change the schedule to improve service for one particular input-output pair since each permutation matrix is a member of \(S_{ij}\) for \(I\) different pairs \(ij\).

Previous work

As mentioned earlier, papers that analyze schedulers based on decomposing the rate matrix include [1], [2], [3]. Analyses of MWM-type schedulers can be found in, for example, [7], [8], [9], [10], [11]. Some frame-based schedulers were presented in [19], [20]. If the switch fabric has an internal speedup of 2 then it is known that it can emulate output-queued switches (in which there is no contention at the inputs) [21], [22], [23]. In [24], an algorithm is presented whose aim is to “track” an idealized fluid policy.

If the switch is sufficiently underloaded then tight delay bounds can be achieved. In [25] it is shown that if the total load on any input or output is at most one quarter of the link rate, then it is possible to serve each \(ij\) pair at least once every \(1/\rho_{ij}\) steps.

The remainder of the paper is organized as follows. In Section II we derive some useful facts about the event, \(\{N_{ij}[T_n, T_{n+m}] \geq \rho_{ij}(m - E^{ij}_1)\}\). In Section III we present our four schedulers and analyze them in detail. In Section IV we present some numerical results to evaluate our bounds for specific rate matrices. We defer the proofs of some of our results to the Appendix.

II. PRELIMINARY ANALYSIS

Note that the event (1) is equivalent to

\[
\exists_{t > 0} : \ N_{ij}[T_n, T_{n+t}] \geq \rho_{ij}(m - E^{ij}_1),
\]

\[
N[0, T_{n+t}] = n + m - 1.
\]

This can be further equivalently written as

\[
\exists_{t > 0} \exists_{s < t} : \ N_{ij}[s, s + t] \geq \rho_{ij}(m - E^{ij}_1),
\]

\[
N[0, s + t] = n + m - 1, N[0, s] = n - 1.
\]

Unfortunately, it is hard in general to calculate the probability of the above event since there is too much dependence between the constituent events. It is however feasible for the case of point processes with independent increments. An example of this special case is the Poisson Competition scheduler that we analyze in Section III-D using a Geo/D/1 queue.

In the remainder of the section, we try to define a subevent of (1) whose probability is easier to bound in the general case. To that end, let \(G_{n,m}\) be the good event \(\{N_{ij}[T_n, T_{n+m}] \geq \rho_{ij}(m - E)\}\). Let \(B_{n,m}\) be the bad event \(B_{n,m} = \overline{G}_{n,m}\).

Let \(\Delta_1, \Delta_2 \in \mathbb{R}^+\) and \(\Delta_3, \Delta_4 \in \mathbb{R}^+\) satisfy

\[
\Delta_1 + \Delta_2 + (\Delta_3 + \Delta_4)/\rho_{ij} \leq E,
\]

where \(E = E^{ij}_1, E^{ij}_2\) or \(E^{ij}_3\), depending on our calculation.

Let \(t = n + m - \Delta_1\) and let \(s = n + \Delta_2\). Note that \(s \) and \(t\) are integers.

**Lemma 3:** Suppose that \(N_{ij}[s, t] \geq \rho_{ij}(t - s) - (\Delta_3 + \Delta_4)\) and \([s, t] \subseteq [T_n, T_{n+m}]\). Then, \(N_{ij}[T_n, T_{n+m}] \geq \rho_{ij}(m - E)\).

**Proof:** We have,

\[
N_{ij}[T_n, T_{n+m}] \geq \rho_{ij}(t - s) - (\Delta_3 + \Delta_4) = \rho_{ij}(n + m - \Delta_1) - (n + \Delta_2) - (\Delta_3 + \Delta_4)/\rho_{ij} \geq \rho_{ij}(m - E).
\]

The first two inequalities come from the assumptions of the lemma. The first equality comes from the definitions of \(s\) and \(t\). The final inequality comes from our constraint on the \(\Delta\)’s. ■

From the definition of \(G_{n,m}\), Lemma 3 implies

\[
G_{n,m} \supseteq \{N_{ij}[s, t] \geq \rho_{ij}(t - s) - (\Delta_3 + \Delta_4)\} \cap \{[s, t] \subseteq [T_n, T_{n+m}]\}\).
\]

By the above results we can focus on the quantity \(N_{ij}[s, t]\) and the relationship between the intervals \([s, t]\) and \([T_n, T_{n+m}]\). However, for the random processes we consider, each interval \([s, t]\) will be dependent on too many other intervals. The way to solve this problem is to concentrate on intervals that have one of their endpoints fixed. For this purpose we must refine our results.

**Lemma 4:** If \(N[0, t] < t + \Delta_1\) and \(N[0, s] \geq s - \Delta_2\) then \([s, t] \subseteq [T_n, T_{n+m}]\).

Note that we are interested in the values of \(m\) such that \(m \geq E\). This implies \(m \geq \Delta_1 + \Delta_2\), which is equivalent to \(s \leq t\). For \(m < E\) the inequalities in (1), (2), (3) do indeed hold.
formally, denote by

The token types such that there are exactly

We note that since

\[ \rho = \phi = 1/10. \]

Fig. 2. Normalized variance of \( N_{ij}(T_n, T_{n+m}) \) for varying frame size \( L \), and \( \rho_{ij} = \phi = 1/10. \).

We first discuss some elementary properties of the scheduler as described above, and then display the latencies. By a routine combinatorial argument we obtain, for \( l = 1, \ldots, \min[\ell_{ij}, m] \),

where \( \ell_{ij} = \sum_{k \in S_{ij}} \ell_k \). Notice that the probability does not depend on \( n \), which reflects the fact that the counting process is stationary. The above explicit expression enables us to compute the latency \( E^{ij}_1 \) defined by (1).

As an aside, we remark that for any fixed \( m \), \( N_{ij}(T_n, T_{n+m}) \) converges in distribution to Binomial random variable \((m, \rho_{ij})\), as \( L \to \infty \). (This can be checked, for example, by Stirling’s formula.)

One may check that the variance of \( N_{ij}(T_n, T_{n+m}) \) is

\[
\sigma^2_{ij}(m) = \frac{L^2}{L-1} \rho_{ij} (1-\rho_{ij}) \frac{m}{L} \left(1 - \frac{m}{L} \right).
\]

We omit this calculation for the benefit of space. Note that the variance forms a bridge (\( \sigma^2_{ij}(0) = \sigma^2_{ij}(L) = 0 \), with the global maximum at \( m = L/2 \)); see Figure 2. Note also that, \( \sigma^2_{ij}(m) \to L \rho_{ij} (1-\rho_{ij}) \frac{m}{L} \left(1 - \frac{m}{L} \right) \), as \( L \to \infty \).

We finally show the main results of this section; asymptotic expressions for the latencies \( E^{ij}_2 \) and \( E^{ij}_3 \). The proofs are given in Appendices A and B.

**Proposition 6:** As \( L \to \infty \),

\[
E^{ij}_2 \to \sqrt{\frac{1}{2} \ln \frac{1}{\varepsilon} \left(\frac{1}{\rho_{ij}} - 1\right) L}.
\]

Note that \( E^{ij}_2 \) scales with the frame length \( L \) as \( O(\sqrt{L}) \).

We have the following result for the latency \( E^{ij}_3 \).

\[
E^{ij}_3 \to \phi \sqrt{\frac{1}{2} \ln \frac{1}{\varepsilon} \left(\frac{1}{\rho_{ij}} - 1\right) L}.
\]

\[
E^{ij}_3 \to \phi \sqrt{\frac{1}{2} \ln \frac{1}{\varepsilon} \left(\frac{1}{\rho_{ij}} - 1\right) L}.
\]
Proposition 7: As $L \to \infty$,

$$E_{3}^{ij} \to \sqrt{A \left( \frac{1}{\rho_{ij}} - 1 \right) L},$$  

(14)

where $A$ is the positive solution of

$$\sum_{t=1}^{\infty} (4t^{2}A - 1) e^{-2t^{2}A} = \frac{1}{2} \varepsilon.$$

Note that $E_{3}^{ij}$ also scales with the frame length $L$ as $O(\sqrt{L})$, but with a different constant. In Figure 3, we plot values for $E_{3}^{ij}$ and $E_{3}^{ij}$ obtained empirically, together with the above limits, for different values of $L$.

B. Random-Phase Periodic Competition

Let $U_{k}$, $k = 1, \ldots, K$, be a collection of independent uniformly distributed random variables on $[0, 1]$. We define the scheduler as follows; for each $k = 1, \ldots, K$,

$$N_{k}(0, t) = \sum_{n=0}^{t} \lfloor (n - 1)/\varphi_{k} + 1/\varphi_{k} \rfloor U_{k}.)$$

Thus the tokens for matrix $M_{k}$ form a periodic stream of period $1/\varphi_{k}$ with the random phase shift $U_{k}/\varphi_{k}$.

We assume that the $\varphi_{k}$ have the property that we can define a frame-based scheduler (see the Introduction). Therefore we only need to concentrate on the time interval $L$. For any interval $[s, t)$, $N_{k}[s, t) \geq \varphi_{k}(t-s) - 1$. This implies,

$$N_{k}[s, t) = \sum_{k \in S_{ij}} (\varphi_{k}(t-s) - 1) \Rightarrow N_{ij}[s, t) \geq \rho_{ij}(t-s) - |S_{ij}|.$$

We follow the method of Section II and set $n = n + \Delta_{1}$, $s = n + \Delta_{2}$, where $\Delta_{1}, \Delta_{2}$ are defined below. For any permutation matrix $M_{k}$, let $a_{k} = \lfloor \varphi_{k}(t \vee) \rfloor$. Then $N_{k}[0, t) = a_{k} + X_{k}$ where $X_{k}$ is a binary random variable with mean $\varphi_{k}(t - a_{k})$. Let $\mu = E[\sum_{k} X_{k}] = \sum_{k} \varphi_{k}(t - a_{k}) - t - \sum_{k} a_{k}$. We have,

$$N[0, t) \geq t + \Delta_{1} \Rightarrow \sum_{k} (a_{k} + X_{k}) \geq t + \Delta_{1} \Rightarrow \sum_{k} X_{k} \geq t + \Delta_{1} - \sum_{k} a_{k} \Rightarrow \sum_{k} X_{k} \geq \mu + \Delta_{1}.$$ 

Therefore, by Hoeffding’s inequality [26],

$$P(N[0, t) \geq t + \Delta_{1}) \leq \exp(-2(\Delta_{1})^{2}/K).$$ 

Similarly, $P(N[0, s) < s + \Delta_{2}) \leq \exp(-2(\Delta_{2})^{2}/K)$.

Let,

$$\Delta_{1} = \Delta_{2} = \left[ \sqrt{(K/2) \ln(2L + 1)} \right],$$

$$\Delta_{3} = |S_{ij}|/2,$$

$$E_{3}^{ij} = (|S_{ij}|/\rho_{ij}) + (2 + \sqrt{2K \ln(2L + 1)}).$$

Then, by the above Hoeffding bounds and the containment (12) from Section II we have,

$$P(\cap_{ij} \cap \{N_{ij}[T_{n}, T_{m}) \geq \rho_{ij}(m - E_{3}^{ij})\})$$

$$\geq 1 - \sum_{t=1}^{L} P(N[0, t) \geq t + \Delta_{1}) -$$

$$- \sum_{s=1}^{L} P(N[0, s) < s + \Delta_{2}) \geq$$

$$1 - \sum_{t=1}^{L} \exp\left(-\frac{2(\Delta_{1})^{2}}{K}\right) - \sum_{s=1}^{L} \exp\left(-\frac{2(\Delta_{2})^{2}}{K}\right)$$

$$\geq 1 - \frac{2L}{2L + 1}.$$

(Note that since we are considering a finite frame we only need sum over $s, t \in \{1, \ldots, L\}$).

Hence with probability $1 - 2L/(2L + 1)$, the rate-latency condition (3) holds for all $ij$. We now show how to derandomize the algorithm so that condition (3) holds with probability 1.

Derandomization

We use the method of conditional probabilities (see e.g. [16]) that is motivated by the following lemma (which we prove in Appendix C).

Lemma 8: Let $Y_{1}, \ldots, Y_{n_{2}}$ be a set of random variables, let $X_{1}, \ldots, X_{n_{2}}$ be a set of independent binary random variables and let $\sigma_{1}, \ldots, \sigma_{n_{2}}$ be a set of events such that for some functions $f_{ij}(\cdot)$,

$$P(\sigma_{i} | A) \leq E \prod_{j=1}^{n_{2}} f_{ij}(X_{j}) | A),$$

(16)

for any event $A$. Then there exists a set of values $y_{1}, \ldots, y_{n_{2}}$ such that,

$$\sum_{i} P(\sigma_{i} | Y_{1} = y_{1}, \ldots, Y_{n_{2}} = y_{n_{2}}) \leq \sum_{i} E \prod_{j=1}^{n_{2}} f_{ij}(X_{j})).$$

In particular if $\sum_{i} E[\prod_{j=1}^{n_{2}} f_{ij}(X_{j})] < 1$ and $\sigma_{i}$ is completely determined by $Y_{1}, \ldots, Y_{n_{2}}$, then,

$$\sum_{i} P(\sigma_{i} | Y_{1} = y_{1}, \ldots, Y_{n_{2}} = y_{n_{2}}) = 0.$$

To compute $y_{e}$ given $y_{1}, \ldots, y_{v-1}$ we minimize,

$$\sum_{i} E[\prod_{j=1}^{n_{2}} f_{ij}(X_{j})] | Y_{1} = y_{1}, \ldots, Y_{v} = y_{v} = \sum_{i} \prod_{j=1}^{n_{2}} E[f_{ij}(X_{j})] | Y_{1} = y_{1}, \ldots, Y_{v} = y_{v},$$

as we vary $y_{v}$ over the full range of $Y_{v}$. (Recall that the random variables $X_{j}$ are independent and so we can exchange the expectation with the product).

To apply this lemma in our setting we take $Y_{1}, \ldots, Y_{n_{2}}$ to be the random phase shifts $U_{1}; U_{2}; \ldots, U_{K}$; $X_{1}, \ldots, X_{n_{2}}$ to be binary random variables of the form $N_{i}[0, t) - \lfloor \varphi_{k}(t \vee) \rfloor$ and $\sigma_{1}, \ldots, \sigma_{n_{2}}$ to be events of the form $\{N[0, t) \geq t + \Delta_{1}\}$ or $\{N[0, s) < s + \Delta_{2}\}$. The functions $f_{ij}(\cdot)$ are defined by,

$$P(N[0, t) \geq t + \Delta_{1}) \leq e^{-\theta(t + \Delta_{1})} E[\prod_{k=1}^{K} e^{\theta N_{k}[0, t]}],$$

(17)

where $\theta = \ln(b(1-a)/a(1-b)) = (t - \sum_{k=1}^{K} \lfloor \varphi_{k}(t \vee) \rfloor / K, b = (t + \Delta_{1} - \sum_{k=1}^{K} \lfloor \varphi_{k}(t \vee) \rfloor / K).$ A similar inequality holds for $P(N[0, s) < s + \Delta_{2})$.

In the derivation of (15) we showed that,

$$\sum_{t=1}^{L} P(N[0, t) \geq t + \Delta_{1}) + \sum_{s=1}^{L} P(N[0, s) < s + \Delta_{2}) \leq \frac{2L}{2L + 1},$$

using Hoeffding bounds that are derived from (17). Hence by Lemma 8 there exist fixed values $u_{1}, \ldots, u_{K}$ for the initial phase shifts such that $N[0, t) < t + \Delta_{1}$ for all $t$ and $N[0, s) \geq s + \Delta_{2}$ for all $s$.

The one complication that arises in the calculation of the $u_{k}$ is that the $U_{k}$ are continuous random variables, they do not
take discrete values. However, as \( U_k \) is varied between 0 and 1, \( N_k([0, t]) = \lfloor \varphi_k t \rfloor \) changes from 0 to 1 at one discrete point. Hence it is sufficient to consider only \( L + 1 \) values of \( U_k \). The right-hand side of (17) may be computed in time polynomial in \( K \) and \( L \), even if some of the phase shifts have already been fixed. Hence, we can fix the value of \( U_k \) in time polynomial in \( K \) and \( L \).

Theorem 9: The resulting deterministic scheduler satisfies (3) with,

\[
E_{ij}^3 = \frac{|S_{ij}|}{\rho_{ij}} + \left(2 + \sqrt{2K \ln(2L + 1)}\right).
\]

C. Random-Distortion Periodic Competition

Let \( U_k, k = 1, \ldots, K, n \in \mathbb{Z}^+ \), be a collection of independent uniformly distributed random variables on \([0, 1]\). We define the scheduler as follows; for each \( k = 1, \ldots, K \),

\[
N_k([0, t]) = n + 1_{U_{k,n} < \varphi_{k,t} - n},
\]

where \( n = \lfloor \varphi_k t \rfloor \). Another interpretation is: the \( n \)th point of the \( k \)th token type is placed uniformly at random in the interval \([n/\varphi_k, (n + 1)/\varphi_k]\).

We make use of the containment (13) from Section II. We apply Hoeffding bounds in a similar manner to the previous subsection to obtain.

\[
\begin{align*}
P(N[0, t] \geq t + \Delta_1) &\leq \exp(-2(\Delta_1^2/K), \\
P(N[0, t] < s - \Delta_2) &\leq \exp(-2(\Delta_2^2)/K), \\
P(N_{ij}[0, t] \leq \rho_{ij} t - \Delta_3) &\leq \exp(-2(\Delta_3^2)/|S_{ij}|), \\
P(N_{ij}[0, s] \geq \rho_{ij} s + \Delta_4) &\leq \exp(-2(\Delta_4^2)/|S_{ij}|).
\end{align*}
\]

Let,

\[
\begin{align*}
\Gamma_1(t) &= P(N[0, t] \geq t + \Delta_1) + \\
\Gamma_2(s) &= P(N[0, s] < s - \Delta_2) + \\
D &= 1 + (4(2I^2 + 2)/\min_k \varphi_k),
\end{align*}
\]

\[
\begin{align*}
\Delta_1 &= \Delta_2 = \lceil \sqrt{(K/2) \cdot \ln D} \rceil, \\
\Delta_3 &= \Delta_4 = \lceil (|S_{ij}|/2) \cdot \ln D \rceil, \\
E_{ij}^3 &= \frac{\Delta_1 + \Delta_2 + (\Delta_3 + \Delta_4)/\rho_{ij}}\rho_{ij}.
\end{align*}
\]

For fixed \( s \) and \( t \) we have,

\[
\Gamma_1(t) + \Gamma_2(s) \leq \frac{(2I^2 + 2)}{D}.
\]

Note that we cannot apply a union bound over \( s \) and \( t \) as we did in the previous subsection because \( s \) and \( t \) range over the entire interval \([0, \infty)\). However, note that if \( \Gamma_1(t) = \Gamma_2(s) = 0 \) for all \( s, t \) then from (13) we know that (3) holds for all \( i, j \) with probability 1. Hence we focus on derandomizing the algorithm.

Derandomization

Instead of placing the \( n \)th token for matrix \( M_k \) at random into the interval \([n/\varphi_k, (n + 1)/\varphi_k]\), we now wish to place it deterministically. Let \( P = \lfloor 2/\min_k \varphi_k \rfloor \). We divide time into intervals of length \( P \), namely, \((0, P), (P, 2P), \ldots\). Let \( A^\omega \) be the set of tokens that fall into the interval \([\omega P, (\omega + 1)P]\) with probability 1, i.e., the \( n \)th token for matrix \( M_k \) is in \( A^\omega \) if and only if \([n/\varphi_k, (n + 1)/\varphi_k] \subseteq [\omega P, (\omega + 1)P]\). Let \( B^\omega \) be the set of tokens that are not in \( A^\omega \) for any \( \omega \) and that fall into the interval \([(\omega - 1/2)P, (\omega + 1/2)P]\) with probability 1. We have chosen \( P \) sufficiently large so that all tokens are in \( A^\omega \cup B^\omega \) for some \( \omega \).

Suppose inductively that for \( \omega < \omega \) we have fixed the positions of all the tokens in \( A^\omega \cup B^\omega \). Since none of the tokens that have already been fixed affect the interval \([\omega P, (\omega + 1)P]\), our previous analysis implies,

\[
\sum_{\omega} \Gamma_1(t) + \sum_{\omega} \Gamma_2(s) \leq \frac{P(2I^2 + 2)}{D}.
\]

By applying the method of conditional probabilities in a similar manner to Section III-B, we can fix the positions of tokens in \( A^\omega \) one after the other so that we still have,

\[
\sum_{\omega} \Gamma_1(t) + \sum_{\omega} \Gamma_2(s) \leq \frac{P(2I^2 + 2)}{D}.
\]

Here, the constituent probabilities of \( \Gamma_1(t) \) and \( \Gamma_2(t) \) are now conditioned on the fact that the tokens in \( A^\omega \) are fixed. We obtain,

\[
\begin{align*}
\sum_{\omega} \Gamma_1(t) + \sum_{\omega} \Gamma_2(s) &\leq \frac{P(2I^2 + 2)}{D} < 1.
\end{align*}
\]

By the method of conditional probabilities we can fix the positions of tokens in \( B^\omega \) so that we still have,

\[
\sum_{\omega} \Gamma_1(t) + \sum_{\omega} \Gamma_2(s) \leq \frac{2P(2I^2 + 2)}{D} < 1.
\]

All tokens in \( A^\omega \cup B^\omega \) are now fixed and so we have a deterministic schedule up to time \((\omega + 1/2)P\). Recall that \( \Gamma_1(t) \) and \( \Gamma_2(s) \) are sums of probabilities. Therefore \( \Gamma_1(t) = \Gamma_2(s) = 0 \) for all \( s, t \in ([\omega - 1/2]P, (\omega + 1/2)P) \). This process can be repeated indefinitely.

Theorem 10: The resulting deterministic scheduler satisfies (3) with,

\[
E_{ij}^3 = \frac{1}{\rho_{ij}} \sqrt{2|S_{ij}| \ln D + (2 + \sqrt{2K \ln D})}.
\]

Adaptation to the substochastic case

For the previous three schedulers, we have assumed that the rate matrix \( M \) is doubly stochastic, i.e., \( \sum_i \rho_{ij} = 1 \) and \( \sum_j \rho_{ij} = 1 \). For the case in which \( M \) is only substochastic, i.e., \( \sum_i \rho_{ij} \leq 1 \) and \( \sum_j \rho_{ij} \leq 1 \), it is known by a result of von Neumann (see e.g. [1]) that there exists a matrix \( M' \) with \( ij \) entry \( \rho_{ij}' \) such that \( \rho_{ij} \leq \rho_{ij}' \) for all \( i, j \) and \( M' \) is doubly stochastic. In this case, we can apply all the results of this paper to the matrix \( M' \) to obtain latencies \( E_{i}'_1, E_{i}'_2 \) and \( E_{i}'_3 \). Note that the \( ij \) traffic might not be able to use all the service it is offered. In this case the residual bandwidth can be used for best-effort traffic.
D. Poisson Competition

For our final scheduler we require that the load on each input and output is strictly less than 1. Let $N_k$ be Poisson with intensity $\varphi_k$, all $k = 1, \ldots, K$. Then the following holds.

Lemma 11: For any $ij$, and $n, m \geq 0$, $l = 1, 2, \ldots, m$,

$$P(N_{ij}[T_n, T_{n+m}] = l) = \binom{m}{l} \rho_{ij}^l (1 - \rho_{ij})^{m-l}.$$  

The above result may be obvious to many; we give an elementary proof in Appendix D. We note that $(T_n, Z_n)_{n \geq 0}$ is a marked point process with independent identically distributed marks, where $Z_n = k$ with probability $\varphi_k$. Our naming of this scheduler is inspired by the Poisson competition theorem (Theorem 1.3, Chapter 8 [27]).

We continue further by observing the following queuing interpretation of the latencies defined in (2) and (3). Locally to this section, assume $\sum_{k=1}^K \varphi_k < 1$; we impose this condition to ensure stability. Moreover, for a fixed $ij$, let $\rho < 1$ be such that $\sum_{k \geq l} \varphi_k = \rho(1 - \rho_{ij})$. We also assume that the counting processes $N_k$ are extended to $\mathbb{R}$, the whole real line. Then, it is not difficult to observe that (2) is equivalent to

$$\{V_{ij}[0] \leq \rho_{ij} E_{ij}^1\},$$

where $V_{ij}[n]$, $n = 0, \pm 1, \pm 2, \ldots$, is the unfinished work of a slotted single server queueing system with infinite buffer capacity, service rate $(1 - \rho_{ij})$ and an arrival process that is 0 or 1 with the probability of an arrival equal to $\rho(1 - \rho_{ij})$. The above observation follows immediately by Reich’s formula,

$$V_{ij}[n] = \max_{m \geq 1}[N_{ij}[T_n, T_{n+m}] - (1 - \rho_{ij})m].$$

From Lemma 11, it follows that the unfinished work is of a Geo/D/1 queue. The distribution of the unfinished work of a Geo/D/1 queue is known in closed form [28], which in our context amounts to

$$P(V_{ij}[0] \leq v) = \frac{1 - q D}{(1 - q)^j} \sum_{l=0}^{j} q(l - q)^{D-1}(-1)^l \left( v - (D - 1)l \right),$$

where $j$ is the integer such that $jD \leq v < (j + 1)D - 1$, $q := \rho(1 - \rho_{ij})$, and $D := 1/(1 - \rho_{ij})$ is implicitly assumed to be an integer. (If $D$ would be a real, then one may redefine $D := \lceil 1/(1 - \rho_{ij}) \rceil$ to obtain a lower bound, provided that $\rho D < 1$.)

The last would enable one at least in theory to exactly compute the latency $E_{ij}^1$ in (2). In practice one may expect numerical instability as $\rho D \to 1$. The following heuristic argument gives us an approximation that is numerically stable, but also brings us some insight about the latency. By appealing to the Brownian approximation (see [17], Sec. 5.7, Equation (7.16)) we claim

$$P(V_{ij}[0] \leq \rho_{ij} E_{ij}^1) \approx 1 - e^{-\frac{1-e}{\rho_{ij}(1-\rho_{ij})} \rho_{ij} E_{ij}^1}.$$ 

Hence, we have

$$E_{ij}^1 \approx \frac{1}{2} \ln e^{-1} \rho \left( 1 - \frac{1 - \rho(1 - \rho_{ij})}{\rho_{ij}} \right).$$

Another approximation can be obtained by considering M/D/1 queue, a continuous time analog\(^2\) of Geo/D/1. A simple exponential approximation is known for M/D/1 ([29], Equation

\(^2\)A notable difference is that with Geo/D/1, in contrast to M/D/1, the number of arrivals over any interval of length $m$ is bounded by $m$. 

---

**Fig. 3.** 0.99-quantile empirical and theoretical limit latencies of Random Permutation Scheduler: (Bottom curve) $E_{ij}^1$ and (Top curve) $E_{ij}^2$. The empirical quantiles are computed from 5 independent samples each of 500 samples of random permutations. The empirical quantiles are shown as averages over the 5 samples along with 0.95-confidence intervals.

**Fig. 4.** Complementary distribution of $V_{ij}$ (dots) empirical estimates, (dotted line) M/D/1, (solid line) Brownian approximation. $V_{ij}$ is estimated by averaging over 1000 random samples of length 10000. $\rho_{ij} = 0.1$. 

As mentioned in the Introduction, the latency $E_{\delta}^{ij}$ does not make much sense for this scheduler since the event in (3) would fail with probability 1 as we require that the inequality in (3) holds for all $n$.

IV. NUMERICAL RESULTS

In this section we evaluate some of our bounds for specific rate matrices. Recall that the best possible latency for input-output pair $ij$ is $1/\rho_{ij}$. Hence the ratio between the latency provided by the scheduler, $E_{\delta}^{ij}$, and the best possible latency is $\rho_{ij}E_{\delta}^{ij}$. For this reason we define $\max_{ij} \rho_{ij}E_{\delta}^{ij}$ to be the figure of merit for a scheduler.

We evaluate the bounds (6) and (7) for the deterministic algorithms derived from Random-Phase Periodic Competition and Random-Distortion Periodic Competition. We compare them with the bound (4) for PGPS. We would like to use matrices drawn uniformly from the set of doubly-stochastic matrices. However, we do not know of a method to generate such a matrix uniformly. Hence we use the following method to generate our example matrices. We start with a uniform matrix in which all entries are equal to $I/L$ where $L = I \times I$. We then repeatedly choose parameters $i_1, i_2, j_1, j_2$ and $\delta$ uniformly at random such that $\delta \leq \min\{\rho_{i_1,j_1}, \rho_{i_2,j_2}\}$. We subtract $\delta$ from $\rho_{i_1,j_1}$ and $\rho_{i_2,j_2}$ and we add $\delta$ to $\rho_{i_1,j_2}$ and $\rho_{i_2,j_1}$. We carry out this operation 100000 times. Note that it preserves the doubly stochastic nature of the matrix. We also ensure that all entries of the rate matrix are integer multiples of $I/L$. Hence we can define frame-based schedulers with frame-length $L$.

In Figure 5 we plot the value of $\max_{ij} \rho_{ij}E_{\delta}^{ij}$ for different values of $L$, the switch size. We see that except for extremely small switches, the bound for the Random-Distortion scheduler is smaller than the bound for the Random-Phase scheduler which is in turn smaller than the bound for PGPS.

In Figure 6 we examine how $\rho_{ij}E_{\delta}^{ij}$ varies for different pairs $ij$. In particular we examine a $64 \times 64$ matrix for which $K = 2423$. For each value of $x$ we plot the fraction of $ij$ pairs for which $\rho_{ij}E_{\delta}^{ij} \leq x$. We see that the bound (6) for the Random-Phase based algorithm is consistently smaller than the bound (4) for PGPS. The bound (7) for the Random-Distortion based algorithm has a smaller range than the other two bounds. There are fewer pairs $ij$ with large values of $\rho_{ij}E_{\delta}^{ij}$ but there are also fewer pairs $ij$ with small values of $\rho_{ij}E_{\delta}^{ij}$. The reason for the latter phenomenon is that the bound (7) is typically larger than the bounds, (4), (6) when the value of $|S_{ij}|$ is small.

We remark that we cannot directly compare the expressions (5) and (8) for the Random Permutation and Poisson Competition schedulers with the bounds (4), (6) and (7) considered in this section. This is because the expression (5) is a limit and the expression (8) is for $E_{\delta}^{ij}$, not $E_{\delta}^{ij}$.

V. CONCLUSION

In this paper we have analyzed the latency of four decomposition-based schedulers for input-queued switches. We believe that there are a number of interesting open problems. First, it is possible that a tighter analysis of our framework of point processes could lead to better bounds for the existing schedulers, and may even motivate the construction of new ones. Second, we know of no non-trivial lower bounds on the best possible latency. It would be interesting to know what is the best value of $\max_{ij} \rho_{ij}E_{\delta}^{ij}$ that can be achieved.

VI. ACKNOWLEDGMENTS

The authors would like to thank Sem Borst, Jean-Yves Le Boudec and Lisa Zhang for many helpful discussions. The second author would like to thank the members of the Math Center at Bell Labs for their hospitality.
REFERENCES


[34] Frank B. Knight, Essentials of Brownian Motion and Diffusion, vol. 18, Mathematical Surveys, American Mathematical Society, 1981.

APPENDIX

I. PROOF OF PROPOSITION 6

Proof: Note that by periodicity of the counting process \( N_{ij} \), (2) is equivalent to

\[
\max_{1 \leq m \leq L} \left[ \rho_{ij} T_m - N_{ij} (T_n, T_{n+m}) \right] \leq \rho_{ij} E_{ij}^3 \]

Now, it is a standard result (e.g. Theorem 1, Section 6.3.7 [30]) that as \( L \to \infty \) we have the convergence in distribution

\[
\max_{0 \leq t \leq 1} N_{ij}(T_n, T_{n+m}) - (1 - \rho_{ij})m \Rightarrow \sqrt{\rho_{ij}(1 - \rho_{ij})L} \sup_{0 \leq t \leq 1} B_0(t),
\]

where \( B_0 \) is Brownian bridge, the Gaussian process with \( B(T_0) = 0 \) and \( \text{cov} [B_0(s), B_0(t)] = s(t - s), 0 \leq s \leq t \leq 1 \). Another definition of Brownian bridge is given by \( B_0(t) = B(t) - tB(1), t \in [0, 1] \), where \( B(t), t \geq 0 \), is standard Brownian motion. Hence, Brownian bridge is a Brownian motion conditioned on hitting 0 at \( t = 1 \).

An exact expression for the complementary distribution of the maximum of Brownian bridge is known (Doob [31]),

\[
P(\sup_{0 \leq t \leq 1} B_0(t) > b) = e^{-b^2}.
\]

From the above convergence and equating the last limit distribution with \( \varepsilon \), we obtain the stated result.

II. PROOF OF PROPOSITION 7

Proof: Note that (3) is equivalent to

\[
\max_{n \geq 0, m \geq 0} \left[ \max_{0 \leq k \leq m} N_{ij} (T_n, T_{n+m}) - (1 - \rho_{ij})m \right] \leq \rho_{ij} E_{ij}^3.
\]

From the periodicity of \( N_{ij} \), it follows

\[
Y := \max_{n \geq 0, m \geq 0} \left[ \max_{1 \leq k \leq m \leq 2L} N_{ij} (T_n, T_{n+m}) - (1 - \rho_{ij})m \right] = \max_{1 \leq k \leq m \leq 2L} X_k - \min_{1 \leq k \leq m \leq 2L} X_k = \max_{1 \leq k \leq 2L} X_k - \min_{1 \leq k \leq 2L} X_k \]

where \( X_k := N_{ij}(T_k, T_{k-1}) - (1 - \rho_{ij})k, k = 1, 2, \ldots \).

Now, similarly as in the proof of Proposition 6, we conclude that, as \( L \to \infty \), \( Y \Rightarrow \sqrt{\rho_{ij}(1 - \rho_{ij})L} W \), where \( W := \text{sup}_{0 \leq t \leq 1} B_0(t) - \text{inf}_{0 \leq t \leq 1} B_0(t) \), the range of the Brownian bridge. It is known that the range of the Brownian bridge
is equal in distribution to the maximum Brownian excursion (see Vervaat [32] and [33]). The Brownian excursion can be represented in terms of standard Brownian motion \( B \) as
\[
(Z(t))_{t \in [0,1]} = d((t_+ - \tau_-)^{-1/2}[B((1-t)\tau_+ + \tau_-)])_{t \in [0,1]},
\]
where \( \tau_+ \) is the last zero of \( B \) before 1 and \( \tau_- \) the first zero after 1. It is known that ([34] Theorem 5.2.10), for \( z > 0 \),
\[
P(\sup_{0 \leq t \leq 1} Z(t) > z) = 2 \sum_{\ell=1}^{\infty} (4\ell^2 z^2 - 1)e^{-2\ell^2 z^2}.
\]

Now let \( E_{ij}^\rho \) be equal to the right-hand side in (14) for some \( A > 0 \). It follows from the above convergence in distribution that
\[
P(\max_{n \geq 0, m > 0} [N_{ij}^\rho(T_n, T_{n+m}) - (1 - \rho_{ij})m] > \rho_{ij} E_{ij}^\rho) \to \to 2 \sum_{\ell=1}^{\infty} (4\ell^2 A - 1) e^{-2\ell^2 A}, \quad \text{as } L \to \infty.
\]
Lastly, we equate the limit in the last display to \( \varepsilon \), for a fixed \( \varepsilon \geq 0 \), which completes the proof.

III. PROOF OF LEMMA 8

We now prove Lemma 8.

Proof: Suppose inductively that we have already chosen \( y_1, \ldots, y_{v-1} \) such that,
\[
\sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j) | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}] \leq \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j)].
\]

This can trivially be done for \( v = 1 \). Then,
\[
\sum_i \sum_{y} P(Y_v = y | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}) \cdot E[\prod_{j=1}^{n_2} f_{ij}(X_j) | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}, Y_v = y] = \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j) | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}] \leq \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j)].
\]

Since \( \sum_y P(Y_v = y | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}) = 1 \), there exists a fixed value \( y_v \) such that,
\[
\sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j) | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}, Y_v = y_v] \leq \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j)].
\]

Then, by Inequality (16),
\[
P(\sigma_v | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}, Y_v = y_v) \leq \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j) | Y_1 = y_1, \ldots, Y_{v-1} = y_{v-1}, Y_v = y_v] \leq \sum_i E[\prod_{j=1}^{n_2} f_{ij}(X_j)].
\]

The proof follows by induction.

IV. PROOF OF LEMMA 11

Proof: Let us first recall definition of Poisson process. A counting process \( N \) on \( \mathbb{R} \) is Poisson with intensity \( \lambda \) if for any two disjoint intervals \( I \) and \( J \) on \( \mathbb{R} \), \( N_I \) and \( N_J \) are independent, and in addition, for any \( I \) on \( \mathbb{R} \),
\[
P(N_I = m) = \frac{(\lambda |I|)^m}{m!} e^{-\lambda |I|}, \quad m = 0, 1, \ldots
\]

Now, it is an elementary result that, if \( N_k k = 1, 2, \ldots, K \) are Poisson counting processes with respective finite intensities \( \varphi_k \) \( k = 1, 2, \ldots, K \), then \( N_{ij} = \sum_{k \in S_i} N_k I \), for any \( ij \) and \( I \subset \mathbb{R} \), is Poisson with intensity \( \rho_{ij} = \sum_{k \in S_i} \varphi_k \).

Thus, we can write
\[
P(N_{ij}(T_n, T_{n+m}) = l) = P(N_{ij}(T_n) = 0, N_{ij}(T_{n+m}) = l) + \rho_{ij} P(N_{ij}(T_n, T_{n+m}) = l - 1) \quad (18)
\]

We exercise a simple calculus, for any \( l = 0, 1, \ldots, m - 1, \)
\[
P(N_{ij}(T_n, T_{n+m}) = l) = \int_0^\infty P(N_{ij}(T_n, T_n + t) = l, N(T_n, T_n + t) = m - 1) dt = \int_0^\infty P(N_{ij}(0, t) = l, N(0, t) = m - 1 - l) dt = \int_0^\infty P(N_{ij}(T_n, T_n + t) = l) dt = \int_0^\infty \rho_{ij} (1 - \rho_{ij})^{m-1-l} e^{-\rho_{ij}} dt = \frac{\rho_{ij} (1 - \rho_{ij})^{m-1-l}}{(m-1)!} \int_0^\infty t^{m-2} e^{-t} dt = \frac{\rho_{ij} (1 - \rho_{ij})^{m-1-l}}{(m-1)!} \frac{\rho_{ij} (1 - \rho_{ij})^{m-1-l}}{(m-1)!}.
\]

The second equality is obtained by \( N_I = \sum_{i \in S_j} N_i I \), any \( I \subset \mathbb{R} \); the third equality is by independence and stationarity of the increments of \( N_{ij} \) and \( N_{ij} \); the forth equality follows by the independence of \( N_{ij} \) and \( N_{ij} \); in the fifth equality we utilize the fact that for any fixed \( I \subset \mathbb{R} \), \( N_{ij} I \) and \( N_{ij} I \) are Poisson random variables; and finally, the last equality follows from \( \int_0^\infty t^{m-2} e^{-t} dt = m! \), for \( m \) an integer.

The statement of the lemma follows by plugging the resulting identity in the last above display into (18).