Performance Evaluation xxx (xxxx) xxx

Contents lists available at ScienceDirect

Performance Evaluation

journal homepage: www.elsevier.com/locate/peva

QPS-r: A cost-effective iterative switching algorithm for input-queued switches

Long Gong [a,*], Jun Xu [a], Liang Liu [a], Siva Theja Maguluri [b]

[a] School of Computer Science, Georgia Institute of Technology, United States of America
[b] School of Industrial & Systems Engineering, Georgia Institute of Technology, United States of America

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Crossbar scheduling
Input-queued switch
Lyapunov stability analysis

ABSTRACT

In an input-queued switch, a crossbar schedule, or a matching between the input ports and the output ports needs to be computed in each switching cycle, or time slot. Designing switching algorithms with very low computational complexity, that lead to high throughput and small delay is a challenging problem. There appears to be a fundamental tradeoff between the computational complexity of the switching algorithm and the resultant throughput and delay. Parallel maximal matching algorithms (adapted for switching) appear to have stricken a sweet spot in this tradeoff, and prior work has shown the following performance guarantees. Using maximal matchings in every time slot results in at least 50% switch throughput and *order-optimal* (*i.e.,* independent of the switch size N) average delay bounds for various traffic arrival processes. On the other hand, their computational complexity can be as low as $O(\log^2 N)$ per port/processor, which is much lower than those of the algorithms such as maximum weighted matching which ensures better throughput performance.

In this work, we propose QPS-r, a parallel iterative switching algorithm that has the lowest possible computational complexity: $O(1)$ per port. Using Lyapunov stability analysis, we show that the throughput and delay performance are identical to those of maximal matching algorithm. Although QPS-r builds upon an existing technique called Queue-Proportional Sampling (QPS), in this paper, we provide analytical guarantees on its throughput and delay under i.i.d. traffic as well as a Markovian traffic model which can model many realistic traffic patterns. We also demonstrate that QPS-3 (running 3 iterations) has comparable empirical throughput and delay performances as iSLIP (running $\log_2 N$ iterations), a refined and optimized representative maximal matching algorithm adapted for switching.

1. Introduction

The volume of network traffic across the Internet and in data-centers continues to grow relentlessly, thanks to existing and emerging data-intensive applications, such as big data analytics, cloud computing, and video streaming. At the same time, the number of network-connected devices is exploding, fueled by the wide adoption of smartphones and the emergence of the Internet of things. To transport and "direct" this massive amount of traffic to their respective

* Corresponding author.
E-mail addresses: gonglong@gatech.edu (L. Gong), jx@cc.gatech.edu (J. Xu), liuliang142857@gatech.edu (L. Liu), siva.theja@gatech.edu (S.T. Maguluri).

https://doi.org/10.1016/j.peva.2021.102197
0166-5316/

Please cite this article as: L. Gong, J. Xu, L. Liu et al., QPS-r: A cost-effective iterative switching algorithm for input-queued switches, Performance Evaluation (2021) 102197, https://doi.org/10.1016/j.peva.2021.102197.

destinations, switches and routers capable of connecting a large number of ports (called *high-radix* [1,2]) and operating at very high line rates are badly needed.

Many present day high-performance switching systems in Internet routers and data-center switches employ an input-queued crossbar to interconnect their input ports and output ports. In an $N \times N$ input-queued crossbar switch, each input port can be connected to only one output port and vice versa in each switching cycle or time slot. Hence, in every time slot, the switch needs to compute a one-to-one *matching* between input and output ports (*i.e.*, the crossbar schedule). A major research challenge in designing high-link-rate high-radix switches is to develop algorithms that can compute "high quality" matchings – *i.e.,* those that result in high switch throughput and low queueing delays for packets – in a few nanoseconds. Clearly, a suitable switching algorithm has to have very low computational complexity, yet output "fairly good" matching decisions most of time.

1.1. The family of maximal matchings

A family of parallel iterative algorithms for computing *maximal matching* (one to which no edge can be added for it to remain a matching, a definition that will be made precise in Section 2) are arguably the best candidates for switching in high-link-rate high-radix switches, because they have reasonably low computational complexities, yet can provide fairly good throughput and delay performance guarantees. More specifically, using maximal matchings as crossbar schedules results in at least 50% switch throughput in theory (and usually much higher throughput in practice) [3]. In addition, it results in low packet delays that also have excellent scaling behaviors such as *order-optimal* (*i.e.,* independent of switch size N) under various traffic arriving processes when the offered load is less than 50% (*i.e.*, within the provable stability region) [4]. In comparison, matchings of higher qualities such as maximum matching (with the largest possible number of edges) and maximum weighted matching (with the highest total edge weight) are much more expensive to compute. Hence, it is fair to say that, maximal matching algorithms overall deliver the biggest "bang" (performance) for the "buck" (computational complexity).

Unfortunately, parallel maximal matching algorithms are still not "dirt cheap" computationally. More specifically, all existing parallel algorithms that compute maximal matchings on *general* $N \times N$ bipartite graphs require a minimum of $O(\log N)$ iterations (rounds of message exchanges). This minimum is attained by the classical algorithm of Israel and Itai [5]; the PIM algorithm [6] is a slight adaptation of this classical algorithm to the switching context, and iSLIP [7] further improves upon PIM by reducing its per-iteration per-port computational complexity to $O(\log N)$ via de-randomizing a computationally expensive ($O(N)$ complexity to be exact) operation in PIM.

1.2. QPS-r: Bigger bang for the buck

In this work, we propose QPS-r, a parallel iterative switching algorithm that has the lowest possible computational complexity: $O(1)$ per port. More specifically, QPS-r requires only r (a small constant independent of N) iterations to compute a matching, and the computational complexity of each iteration is only $O(1)$; here QPS stands for Queue-Proportional Sampling, an add-on technique proposed in [8] that we will describe shortly. Yet, even the matchings that QPS-1 (running only a single iteration) computes have the same (reasonably high) quality as maximal matchings in the following sense: Using such matchings as crossbar schedules results in exactly the same aforementioned provable throughput and delay guarantees as using maximal matchings, as we will show using Lyapunov stability analysis. QPS-r performs as well as maximal matching algorithms not just in theory: We will show in Section 6 that QPS-3 (running 3 iterations) has comparable empirical throughput and delay performances as iSLIP (running $\log_2 N$ iterations), a refined and optimized representative maximal matching algorithm adapted for switching, under various workloads. Note that matchings that QPS-r computes are generally not maximal. QPS-r can make do with less (iterations) because the queue-proportional sampling operation implicitly makes use of the queue length information, which maximal matching algorithms do not. One major contribution of this work is to discover the family of (QPS-r)-generated matchings that is even more cost-effective.

Although QPS-r builds on the QPS data structure and algorithm proposed in [8], our work on QPS-r is very different in three important aspects. First, in [8], QPS was used only as an add-on to other switching algorithms such as iSLIP [7] and SERENA [9] by generating a starter matching for other switching algorithms to further refine, whereas in this work, QPS-r is used only as a stand-alone algorithm. Second, we are the first to discover and prove that (QPS-r)-generated matchings and maximal matchings provide exactly the same aforementioned performance guarantees, whereas in [8], no such mathematical similarity or connection was mentioned. Third, the establishment of this mathematical similarity is an important theoretical contribution in itself, because maximal matchings have long been established as a cost-effective family both in switching [6,7] and in wireless networking [4,10], and with this connection we have considerably enlarged this family.

Although we show that QPS-r has exactly the same throughput and delay bounds as those of maximal matchings established in [3,4,10], our proofs are different for the following reason. A *departure inequality* (see Property 1), satisfied by all maximal matching algorithms was used in the throughput analysis of [3] and the delay analysis of [4,10]. This inequality, however, is not satisfied by QPS-r in general. Instead, QPS-r satisfies this departure inequality in expectation, which is a much weaker guarantee. The methodological contributions of this work are twofold. First, we prove two

theorems stating that this much weaker guarantee is sufficient for obtaining the same throughput and delay bounds under i.i.d. traffic arrivals. Second, we generalize both results to the more general Markovian arrivals.

The rest of this paper is organized as follows. In Section 2, we provide some background on the switching problem in input-queued crossbar switches. In Section 3, we first review QPS, and then describe QPS-r. Then in Sections 4 and 5, we derive the throughput and the queue length (and delay) bounds of QPS-r, followed by the performance evaluation in Section 6. In Section 7, we survey related work before concluding this paper in Section 8.

2. Background on crossbar scheduling

In this section, we provide a brief introduction to the crossbar scheduling (switching) problem, and describe and compare the aforementioned three different types of matchings. Throughout this paper we adopt the standard assumption [11, Chapter 2, Page 21] that all the incoming variable-length packets are first segmented into fixed-length packets (also referred to as cells), and then reassembled at their respective output ports before leaving the switch. Each fixed-length packet takes one time slot to switch. We also assume that all input links/ports and output links/ports operate at the same normalized line rate of 1, and so do all wires and crosspoints inside the crossbar.

2.1. Input-queued crossbar switch

In an $N \times N$ input-queued crossbar switch, each input port has N Virtual Output Queues (VOQs) [12]. The jth VOQ at input port i serves as a buffer for packets going from input port i to output port j. The use of VOQs solves the Head-of-Line (HOL) blocking issue [13], which severely limits the throughput of the switch system.

An $N \times N$ input-queued crossbar can be modeled as a bipartite graph, of which the two disjoint vertex sets are the N input ports and the N output ports respectively. In this bipartite graph, there is an edge between input port i and output port j, if and only if the jth VOQ at input port i, the corresponding VOQ, is nonempty. The weight of this edge is defined as the length of (*i.e.,* the number of packets buffered at) this VOQ. A set of such edges constitutes a *valid crossbar schedule*, or a *matching*, if any two of them do not share a common vertex. The weight of a matching is the total weight of all the edges belonging to it (*i.e.,* the total length of all corresponding VOQs).

A matching M can be represented as an $N \times N$ *sub-permutation matrix* (a 0-1 matrix that contains at most one entry of "1" in each row and in each column) $S=(s_{ij})$ as follows: $s_{ij} = 1$ if and only if the edge between input port i and output port j is contained in M (*i.e.,* input port i is matched to output port j in M). To avoid any confusion, only S (not M) is used to denote a matching in the sequel, and it can be both a set (of edges) and a matrix.

2.2. Maximal matching

As mentioned in Section 1, three types of matchings play important roles in crossbar scheduling problems: (I) maximal matchings, (II) maximum matchings, and (III) maximum weighted matchings. A matching S is called a *maximal matching*, if it is no longer a matching, when any edge not in S is added to it. A matching with the largest possible number of edges is called a *maximum matching* or *maximum cardinality matching*. Neither maximal matchings nor maximum matchings take into account the weights of edges, whereas *maximum weighted matchings* do. A maximum weighted matching is one that has the largest total weight among all matchings. By definition, any maximum matching or maximum weighted matching is also a maximal matching, but neither converse is generally true.

As mentioned earlier, the family of maximal matchings has long been recognized as a cost-effective family for crossbar scheduling. Compared to maximal matching, maximum weighted matching (MWM) (*i.e.,* the well-known MaxWeight scheduler [14] in the context of scheduling transmissions in wireless networking) is much less cost effective. Although MWM provides stronger provable guarantees such as 100% switch throughput [15,16] and $O(N)$ average packet delay [17] whenever the offered load is less than 100% in theory (and usually much better empirical delay performance in practice as shown in [15]), the state-of-the-art serial MWM algorithm (suitable for switching) has a prohibitively high computational complexity of $O(N^{2.5} \log W)$ [18], where W is the maximum possible weight (length) of an edge (VOQ). By the same measure, maximum matching is not a great deal either: It is only slightly cheaper to compute than MWM, yet using maximum matchings as crossbar schedules generally cannot guarantee 100% throughput [19].

3. The QPS-*r* algorithm

The QPS-r algorithm simply runs r iterations of QPS (Queue-Proportional Sampling) [8] to arrive at a matching, so its computational complexity per port is exactly r times those of QPS. Since r is a small constant, it is $O(1)$, same as that of QPS. In the following two subsections, we describe QPS and QPS-r respectively in more detail.

3.1. Queue-proportional sampling (QPS)

QPS was used in [8] as an "add-on" to augment other switching algorithms as follows. It generates a starter matching, which is then populated (*i.e.*, adding more edges to it) and refined, by other switching algorithms such as iSLIP [7] and SERENA [9], into a final matching. To generate such a starter matching, QPS needs to run only one iteration, which consists of two phases, namely, a proposing phase and an accepting phase. We briefly describe them in this section for this paper to be self-contained.

Proposing Phase. In this phase, each input port proposes to *exactly one* output port – decided by the QPS strategy – unless it has no packet to transmit. Here we will only describe the operations at input port 1; that at any other input port is identical. Like in [8], we denote by $m_1, m_2, \ldots, m_N$ the respective queue lengths of the N VOQs at input port 1, and by m their total (*i.e.*, $m \triangleq \sum_{k=1}^{N} m_k$). Input port 1 simply samples an output port j with probability $\frac{m_j}{m}$, *i.e.,* proportional to m_j, the length of the corresponding VOQ (hence the name QPS); it then proposes to output port j, with the value m_j that will be used in the next phase. The computational complexity of this QPS operation, carried out using a simple data structure proposed in [8], is $O(1)$ per (input) port.

Accepting Phase. We describe only the action of output port 1 in the accepting phase; that of any other output port is identical. The action of output port 1 depends on the number of proposals it receives. If it receives exactly one proposal from an input port, it will accept the proposal and match with the input port. However, if it receives proposals from multiple input ports, it will accept the proposal accompanied with the largest VOQ length (called the "longest VOQ first" accepting strategy), with ties broken uniformly at random. The computational complexity of this accepting strategy is $O(1)$ on average and can be made $O(1)$ even in the worst case [8].

3.2. The QPS-r scheme

The QPS-r scheme simply runs r QPS iterations. In each iteration, each input port that is not matched yet, first proposes to an output port according to the QPS proposing strategy; each output port that is not matched yet, accepts a proposal (if it has received any) according to the "longest VOQ first" accepting strategy. Hence, if an input port has to propose multiple times (once in each iteration), due to all its proposals (except perhaps the last) being rejected, the identities of the output ports it "samples" (*i.e.,* proposes to) during these iterations are samples with replacement, which more precisely are i.i.d. random variables with a queue-proportional distribution.

At the first glance, sampling with replacement may appear to be an obviously suboptimal strategy for the following reason. There is a nonzero probability for an input port to propose to the same output port multiple times, but since the first (rejected) proposal implies this output port has already accepted "someone else" (a proposal from another input port), all subsequent proposals to this output port will surely go to waste. For this reason, sampling without replacement (*i.e.*, avoiding all output ports proposed to before) may sound like an obviously better strategy. However, it is really not, since compared to sampling with replacement, it has a much higher computational complexity of $O(\log N)$, but improves the throughput and delay performances only slightly according to our simulation studies.

4. Throughput and delay analysis under i.i.d. arrivals

In this section, we show that QPS-1 (*i.e.*, running a single QPS iteration) delivers exactly the same provable throughput and delay guarantees as maximal matching algorithms under i.i.d. traffic arrivals. When $r > 1$, QPS-r clearly should have better throughput and delay performances than QPS-1, as more input and output ports can be matched up during subsequent iterations, although we are not able to derive stronger bounds.

4.1. Preliminaries

In this section, we introduce the notation and assumptions that will later be used in our derivations. We define three $N \times N$ matrices $Q(t)$, $A(t)$, and $D(t)$. Let $Q(t) \triangleq \left(q_{ij}(t)\right)$ be the queue length matrix where each $q_{ij}(t)$ is the length of the jth VOQ at input port i during time slot t. With a slight abuse of notation, we refer to this VOQ as q_{ij} (without the t term).

We define $Q_{i*}(t)$ and $Q_{*j}(t)$ as the sum of the ith row and the sum of the jth column respectively of $Q(t)$, *i.e.*, $Q_{i*}(t) \triangleq \sum_j q_{ij}(t)$ and $Q_{*j}(t) \triangleq \sum_i q_{ij}(t)$. With a similar abuse of notation, we define Q_{i*} as the VOQ set $\{q_{i1}, q_{i2}, \ldots, q_{iN}\}$ (*i.e.*, those on the ith row), and Q_{*j} as $\{q_{1j}, q_{2j}, \ldots, q_{Nj}\}$ (*i.e.*, those on the jth column).

Now we introduce a concept that lies at the heart of our derivations: neighborhood. For each VOQ q_{ij}, we define its neighborhood as $Q_{i*} \bigcup Q_{*j}$, the set of VOQs on the ith row or the jth column. We denote this neighborhood as $Q_{ij}^{\dagger}$, since it has the shape of a cross. With a slight abuse of notation, we use $Q_{ij}^{\dagger}$ to also denote the set of the corresponding input–output port pairs $\{(l, w) : q_{lw} \in Q_{ij}^{\dagger}\}$. Fig. 1 illustrates $Q_{ij}^{\dagger}$, where the row and column in the shadow are the VOQ sets Q_{i*} and Q_{*j} respectively. $Q_{ij}^{\dagger}$ can be viewed as the *interference set* of VOQs for VOQ q_{ij} [4,10], as no other VOQ in $Q_{ij}^{\dagger}$ can be active (*i.e.,* transmit packets) simultaneously with q_{ij}. We define $Q_{ij}^{\dagger}(t)$ as the total length of all VOQs in (the set) $Q_{ij}^{\dagger}$ at time slot t, that is

$$Q_{ij}^{\dagger}(t) \triangleq Q_{i*}(t) - q_{ij}(t) + Q_{*j}(t). \tag{1}$$

Here we need to subtract the term $q_{ij}(t)$ so that it is not double-counted (in both $Q_{i*}(t)$ and $Q_{*j}(t)$).

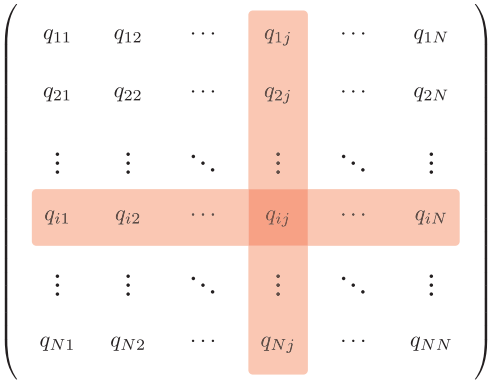

Fig. 1. $Q_{ij}^{\dagger}$: neighborhood of q_{ij}.

Let $A(t)=\big(a_{ij}(t)\big)$ be the traffic arrival matrix where $a_{ij}(t)$ is the number of packets arriving at the input port i destined for output port j during time slot t. Here, we assume that, for each $1 \le i, j \le N$, $\{a_{ij}(t)\}_{t=0}^{\infty}$ is a sequence of i.i.d. random variables, and this sequence is independent of other sequences (for a different i and/or j). Like in [20], we further assume that $a_{ij}(t)$ is upper-bounded by a_{max} for any i, j at any time slot t. Thus, the second moment of their common distribution $(= \mathbf{E}\big[a_{ij}^2(0)\big])$ is finite. Our analysis, however, holds for more general arrival processes (*e.g.*, Markovian arrivals) that were considered in [4,10], as we will elaborate shortly. Let $D(t) = \big(d_{ij}(t)\big)$ be the departure matrix for time slot t output by the switching algorithm. Similar to S, $D(t)$ is a 0-1 matrix in which $d_{ij}(t) = 1$ if and only if a packet departs from q_{ij} during time slot t. For any i, j, the queue length process $q_{ij}(t)$ evolves as follows:

$$q_{ij}(t+1) = q_{ij}(t) - d_{ij}(t) + a_{ij}(t). \tag{2}$$

Let $\Lambda = \big(\lambda_{ij}\big)$ be the (normalized) traffic rate matrix (associated with $A(t)$) where λ_{ij} is normalized (to the percentage of the line rate of an input/output link) mean arrival rate of packets to VOQ q_{ij}. With $a_{ij}(t)$ being an i.i.d. process, we have $\lambda_{ij} = \mathbf{E}\big[a_{ij}(0)\big]$. We define ρ_{Λ} as the maximum load factor imposed on any input or output port by Λ,

$$\rho_{\Lambda} \triangleq \max\Big\{\max_{1\le i\le N}\{\sum_{j}\lambda_{ij}\},\ \max_{1\le j\le N}\{\sum_{i}\lambda_{ij}\}\Big\} \tag{3}$$

A switching algorithm is said to achieve 100% throughput or be throughput-optimal if the (packet) queues are stable whenever $\rho_{\Lambda} < 1$.

As mentioned before, we will prove in this section that, same as the maximal matching algorithms, QPS-1 is stable *under any traffic arrival process* $A(t)$, defined above, whose rate matrix Λ satisfies $\rho_{\Lambda} < 1/2$ (*i.e.*, can provably attain at least 50% throughput, or half of the maximum). We also derive the average delay bound for QPS-1, which we show is *order-optimal* (*i.e.*, independent of switch size N). In the sequel, we drop the subscript term from ρ_{Λ} and simply denote it as ρ.

Similar to $Q_{ij}^{\dagger}(t)$, we define $A_{ij}^{\dagger}(t)$ as the total number of packet arrivals to all VOQs in the neighborhood set $Q_{ij}^{\dagger}$:

$$A_{ij}^{\dagger}(t) \triangleq A_{i*}(t) - a_{ij}(t) + A_{*j}(t), \tag{4}$$

where $A_{i*}(t)$ and $A_{*j}(t)$ are similarly defined as $Q_{i*}(t)$ and $Q_{*j}(t)$ respectively. $D_{ij}^{\dagger}(t)$, $D_{i*}(t)$, and $D_{*j}(t)$ are similarly defined, so is $\Lambda_{ij}^{\dagger}$. We now state some simple facts concerning $D(t)$, $A(t)$, and Λ as follows.

Fact 1. *Given any switching algorithm, for any i, j, we have, $D_{i*}(t) \le 1$ (at most one packet can depart from input port i during time slot t), $D_{*j}(t) \le 1$, and $D_{ij}^{\dagger}(t) \le 2$.*

Fact 2. *Given any i.i.d. arrival process $A(t)$ and its rate matrix is Λ whose maximum load factor is defined in (3), for any i, j, we have $\mathbf{E}[A_{ij}^{\dagger}(t)] = \Lambda_{ij}^{\dagger} \le 2\rho$.*

The following fact is slightly less obvious.

Fact 3. *Given any switching algorithm, for any i, j, we have*

$$d_{ij}(t)D_{ij}^{\dagger}(t) = d_{ij}(t). \tag{5}$$

Fact 3 holds because, as mentioned earlier, no other VOQ in $Q_{ij}^{\dagger}$ (see Fig. 1) can be active simultaneously with q_{ij}. More precisely, if $d_{ij}(t)=1$ (*i.e.*, VOQ q_{ij} is active during time slot t) then $D_{ij}^{\dagger}(t) \triangleq D_{i*}(t)-d_{ij}(t)+D_{*j}(t)=1-1+1=1$; otherwise $d_{ij}(t)D_{ij}^{\dagger}(t)=0 \cdot D_{ij}^{\dagger}(t)=0=d_{ij}(t)$.

4.2. Why QPS-1 is just as good?

The provable throughput and delay bounds of maximal matching algorithms were derived from a "departure inequality" (to be stated and proved next) that all maximal matchings satisfy. This inequality, however, is not in general satisfied by matchings generated by QPS-1. Rather, QPS-1 satisfies a much weaker form of departure inequality (Lemma 1). Fortunately, this much weaker condition is sufficient for proving the same throughput bound and delay bounds, as will be proved in Theorem 1 and Theorem 2 respectively.

Property 1 (*Departure Inequality, stated as Lemma 1 in [4,10]*)**.** *If during a time slot t, the crossbar schedule is a* maximal matching*, then each departure process $D_{ij}^{\dagger}(t)$ satisfies the following inequality*

$$q_{ij}(t)D_{ij}^{\dagger}(t) \geq q_{ij}(t). \tag{6}$$

Proof. We reproduce the proof of Property 1 with a slightly different approach for this paper to be self-contained. Suppose the contrary is true, *i.e.*, $q_{ij}(t)D_{ij}^{\dagger}(t) < q_{ij}(t)$. This can only happen when $q_{ij}(t) > 0$ and $D_{ij}^{\dagger}(t)=0$. However, $D_{ij}^{\dagger}(t)=0$ implies that no nonempty VOQ (edge) in the neighborhood $Q_{ij}^{\dagger}$ (see Fig. 1) is a part of the matching. Then this matching cannot be maximal (a contradiction) since it can be enlarged by the addition of the nonempty VOQ (edge) q_{ij}. □

Clearly, the departure inequality (6) above implies the following much weaker form of it:

$$\sum_{i,j} \mathbf{E}\left[q_{ij}(t)D_{ij}^{\dagger}(t) \mid Q(t)\right] \geq \sum_{i,j} q_{ij}(t) \tag{7}$$

In the rest of this section, we prove the following lemma:

Lemma 1. *The matching generated by QPS-1, during any time slot t, satisfies the much weaker "departure inequality" (7).*

Before we prove Lemma 1, we introduce an important definition and state four facts about QPS-1 that will be used later in the proof. In the following, we will run into several innocuous possible $\frac{0}{0}$ situations that all result from queue-proportional sampling, and we consider all of them to be 0.

We define $\alpha_{ij}(t)$ as the probability of the event that the proposal from input port i to output port j is accepted during the accepting phase, conditioned upon the event that input port i did propose to output port j during the proposing phase. With this definition, we have the first fact

$$\mathbf{E}\left[d_{ij}(t) \mid Q(t)\right] = \frac{q_{ij}(t)}{Q_{i*}(t)} \cdot \alpha_{ij}(t), \tag{8}$$

since both sides (note $d_{ij}(t)$ is a 0-1 *r.v.*) are the probability that i proposes to j and this proposal is accepted. Summing over j on both sides, we obtain the second fact

$$\mathbf{E}\left[D_{i*}(t) \mid Q(t)\right] = \sum_{j} \frac{q_{ij}(t)}{Q_{i*}(t)} \cdot \alpha_{ij}(t). \tag{9}$$

The third fact is that, for any output port j,

$$\mathbf{E}\left[D_{*j}(t) \mid Q(t)\right] = 1 - \prod_{i}\left(1 - \frac{q_{ij}(t)}{Q_{i*}(t)}\right). \tag{10}$$

In this equation, the LHS is the conditional *probability* ($D_{*j}(t)$ is also a 0-1 *r.v.*) that at least one proposal is received and accepted by output port j, and the second term on the RHS of (10) is the probability that no input port proposes to output port j (so j receives no proposal). This equation holds since when j receives one or more proposals, it will accept one of them (the one with the longest VOQ).

The fourth fact is that, for any i, j,

$$\alpha_{ij}(t) \geq \prod_{k \neq i}\left(1 - \frac{q_{kj}(t)}{Q_{k*}(t)}\right). \tag{11}$$

This inequality holds because when input port i proposes to output port j, and no other input port does, j has no choice but to accept i's proposal.

4.3. Proof of Lemma 1

Now we are ready to prove Lemma 1.
By the definition of $D_{ij}^{\dagger}(t) \triangleq D_{i*}(t) - d_{ij}(t) + D_{*j}(t)$, we have,

$$\sum_{i,j} \mathbf{E}\big[q_{ij}(t) D_{ij}^{\dagger}(t) \mid Q(t)\big] = \sum_{i,j} q_{ij}(t) \mathbf{E}\big[D_{i*}(t) \mid Q(t)\big] - \sum_{i,j} q_{ij}(t) \mathbf{E}\big[d_{ij}(t) \mid Q(t)\big] + \sum_{i,j} q_{ij}(t) \mathbf{E}\big[D_{*j}(t) \mid Q(t)\big]. \tag{12}$$

Focusing on the first term on the RHS of (12) and using (9), we have,

$$\begin{aligned} &\sum_{i,j} q_{ij}(t) \mathbf{E}\big[D_{i*}(t) \mid Q(t)\big] \\ &= \sum_{i} Q_{i*}(t) \mathbf{E}\big[D_{i*}(t) \mid Q(t)\big] \\ &= \sum_{i} Q_{i*}(t) \Big(\sum_{j} \frac{q_{ij}(t)}{Q_{i*}(t)} \cdot \alpha_{ij}(t) \Big) \\ &= \sum_{i,j} q_{ij}(t) \alpha_{ij}(t). \end{aligned} \tag{13}$$

Focusing the second term on the RHS of (12) and using (8), we have

$$-\sum_{i,j} q_{ij}(t) \mathbf{E}\big[d_{ij}(t) \mid Q(t)\big] = -\sum_{i,j} q_{ij}(t) \alpha_{ij}(t) \frac{q_{ij}(t)}{Q_{i*}(t)}. \tag{14}$$

Hence, the sum of the first two terms in (12) is equal to

$$\begin{aligned} &\sum_{i,j} q_{ij}(t) \alpha_{ij}(t) \Big(1 - \frac{q_{ij}(t)}{Q_{i*}(t)} \Big) \\ &\geq \sum_{i,j} q_{ij}(t) \Big(\prod_{k \neq i} \big(1 - \frac{q_{kj}(t)}{Q_{k*}(t)}\big) \Big) \big(1 - \frac{q_{ij}(t)}{Q_{i*}(t)}\big) \qquad (15) \\ &= \sum_{i,j} q_{ij}(t) \prod_{i} \big(1 - \frac{q_{ij}(t)}{Q_{i*}(t)}\big) \\ &= \sum_{i,j} q_{ij}(t) \Big(1 - \mathbf{E}\big[D_{*j}(t) \mid Q(t)\big] \Big). \qquad (16) \end{aligned}$$

Note that inequality (15) is due to inequality (11) and (16) is due to (10). We now arrive at the weaker departure inequality (7), when adding the third and last term in (12) to the RHS of (16).

4.4. Throughput analysis

In this section we prove, through Lyapunov stability analysis, the following theorem (*i.e.*, Theorem 1) which states that any switching algorithm that satisfies the weaker departure inequality (7), including QPS-1 as shown in Lemma 1, can attain at least 50% throughput under i.i.d. arrivals. The same throughput bound was proved in [3], through fluid limit analysis, for maximal matching algorithms using the (stronger) departure inequality (6) which as stated earlier is not in general satisfied by matchings generated by QPS-1.

Theorem 1. *Let $\{Q(t)\}_{t=0}^{\infty}$ be the queueing process of a switching system that is an irreducible Markov chain. Let the departure process of $\{Q(t)\}_{t=0}^{\infty}$ satisfy the weaker "departure inequality" (7). Then whenever its maximum load factor satisfies $\rho < 1/2$, the queueing process is stable in the following sense: (I) The Markov chain $\{Q(t)\}_{t=0}^{\infty}$ is positive recurrent and hence converges to a stationary distribution $\bar{Q}$; (II) The first moment of $\bar{Q}$ is finite.*

Proof. Here we prove only (I), since Theorem 2 that we will shortly prove implies (II). We define the following Lyapunov function of $Q(t)$: $L\big(Q(t)\big) = \sum_{i,j} q_{ij}(t) Q_{ij}^{\dagger}(t)$, where $Q_{ij}^{\dagger}(t)$ is defined earlier in (1). This Lyapunov function was first introduced in [4] for the delay analysis of maximal matching algorithms for wireless networking. By the Foster–Lyapunov stability criterion [21, Proposition 2.1.1], to prove that $\{Q(t)\}_{t=0}^{\infty}$ is positive recurrent, it suffices to show that, there exists a constant $B > 0$ such that whenever the total queue (VOQ) length $\|Q(t)\|_1 > B$ (because it is not hard to verify that the

complement set of states $\{Q(t) : \|Q(t)\|_1 \le B\}$ is finite and the drift is bounded whenever $Q(t)$ belongs to this set), we have

$$\mathbf{E}\big[L\big(Q(t+1)\big) - L\big(Q(t)\big) \mid Q(t)\big] \le -\epsilon, \tag{17}$$

where $\epsilon > 0$ is a constant. It is not hard to check (for more detailed derivations, please refer to [4]),

$$\begin{aligned} L\big(Q(t+1)\big) - L\big(Q(t)\big) =& 2\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \\ &+ \sum_{i,j}\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big). \end{aligned} \tag{18}$$

Hence the drift (LHS of (17)) can be written as

$$\begin{aligned} &\mathbf{E}\big[L\big(Q(t+1)\big) - L\big(Q(t)\big) \mid Q(t)\big] \\ =&\mathbf{E}\Big[2\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\Big] \\ &+ \mathbf{E}\Big[\sum_{i,j}\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\Big]. \end{aligned} \tag{19}$$

Now we claim the following two inequalities, which we will prove shortly.

$$\mathbf{E}\big[2\textstyle\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\big] \le 2(2\rho - 1)\|Q(t)\|_1. \tag{20}$$

$$\mathbf{E}\big[\textstyle\sum_{i,j}\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\big] \le CN^2. \tag{21}$$

With inequalities (20) and (21) substituted into (19), we have

$$\mathbf{E}\big[L\big(Q(t+1)\big) - L\big(Q(t)\big) \mid Q(t)\big] \le 2(2\rho - 1)\|Q(t)\|_1 + CN^2.$$

where $C > 0$ is a constant. Since $\rho < 1/2$, we have $2\rho - 1 < 0$. Hence, there exist $B, \epsilon > 0$ such that, whenever $\|Q(t)\|_1 > B$,

$$\mathbf{E}\big[L\big(Q(t+1)\big) - L\big(Q(t)\big) \mid Q(t)\big] \le -\epsilon.$$

Now we proceed to prove inequality (20).

$$\begin{aligned} &\mathbf{E}\Big[2\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\Big] \\ =&2\Big(\sum_{i,j}\mathbf{E}\Big[q_{ij}(t)A_{ij}^{\dagger}(t) \mid Q(t)\Big] - \sum_{i,j}\mathbf{E}\Big[q_{ij}(t)D_{ij}^{\dagger}(t) \mid Q(t)\Big]\Big) \\ \le&2\Big(2\rho\sum_{i,j}\mathbf{E}\Big[q_{ij}(t) \mid Q(t)\Big] - \sum_{i,j} q_{ij}(t)\Big) \qquad (22) \\ =&2(2\rho - 1)\|Q(t)\|_1. \qquad (23) \end{aligned}$$

In the above derivations, inequality (22) holds due to the weaker departure inequality (7), $A(t)$ being independent of $Q(t)$ for any t, and Fact 2 that $\mathbf{E}[A_{ij}^{\dagger}(t)] \le 2\rho$.

Now we proceed to prove (21), which upper-bounds the conditional expectation $\mathbf{E}\big[\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) \mid Q(t)\big]$. It suffices however to upper-bound the unconditional expectation $\mathbf{E}\big[\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\big]$, which we will do in the following, since we can obtain the same upper bounds on $\mathbf{E}[D_{ij}^{\dagger}(t)]$ and $\mathbf{E}[d_{ij}(t)]$ (2 and 1 respectively) whether the expectations are conditional (on $Q(t)$) or not. Note the other two terms $A_{ij}^{\dagger}(t)$ and $a_{ij}(t)$ are independent of (the condition) $Q(t)$.

As for any i, j, $a_{ij}(t)$ is i.i.d., we have,

$$\begin{aligned} &\mathbf{E}\big[\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\big] \qquad (24) \\ =&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t) - d_{ij}(t)A_{ij}^{\dagger}(t) - a_{ij}(t)D_{ij}^{\dagger}(t) + d_{ij}(t)D_{ij}^{\dagger}(t)] \\ =&\mathbf{E}[a_{ij}^2(t)] - \lambda_{ij}^2 + \lambda_{ij}\Lambda_{ij}^{\dagger} - \mathbf{E}[d_{ij}(t)]\Lambda_{ij}^{\dagger} - \lambda_{ij}\mathbf{E}[D_{ij}^{\dagger}(t)] + \mathbf{E}[d_{ij}(t)D_{ij}^{\dagger}(t)] \\ =&\mathbf{E}[a_{ij}^2(t)] - \lambda_{ij}^2 + \lambda_{ij}\Lambda_{ij}^{\dagger} - \mathbf{E}[d_{ij}(t)]\Lambda_{ij}^{\dagger} - \lambda_{ij}\mathbf{E}[D_{ij}^{\dagger}(t)] + \mathbf{E}[d_{ij}(t)]. \qquad (25) \end{aligned}$$

In arriving at (25), we have used (5). The RHS of (25) can be bounded by a constant $C > 0$ due to the following assumptions and facts: $\mathbf{E}[a_{ij}^2(t)] = \mathbf{E}[a_{ij}^2(0)] < \infty$ for any t, $d_{ij}(t) \le 1$, $D_{ij}^{\dagger}(t) \le 2$, $\lambda_{ij} \le \rho < 1/2$, and $\Lambda_{ij}^{\dagger} \le 2\rho < 1$. Therefore, we have (by applying $\sum_{i,j}$ to both (24) and the RHS of (25))

$$\sum_{i,j}\mathbf{E}\Big[\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\Big] \le CN^2. \quad \square$$

Remarks. Now that we have proved that $\{Q(t)\}_{t=0}^{\infty}$ is positive recurrent. Hence, we have, in steady state, for any $1 \le i, j \le N$, $\mathbf{E}[d_{ij}(t)] = \lambda_{ij}$. Therefore, we have, in steady state, for any $1 \le i, j \le N$

$$\mathbf{E}\Big[\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\Big] = \sigma_{ij}^2 - \lambda_{ij}\Lambda_{ij}^{\dagger} + \lambda_{ij}. \tag{26}$$

where $\sigma_{ij}^2 = \mathbf{E}[a_{ij}^2(t)] - \lambda_{ij}^2$ is the variance of $a_{ij}(t)$, because (25) can be simplified as the RHS of (26) in steady state.

Now, we prove the following corollary of Theorem 1 which, in combination with Lemma 1, shows that QPS-1 can attain at least 50% throughput.

Corollary 1. *Under an i.i.d. arrival process, whenever the maximum load factor satisfies $\rho < 1/2$, QPS-1 is stable in the following sense: The resulting queueing process $\{Q(t)\}_{t=0}^{\infty}$ is a positive recurrent Markov chain and its stationary distribution $\bar{Q}$ has finite first moment.*

Proof. $\{Q(t)\}_{t=0}^{\infty}$ is clearly a Markov chain, since in (2), the term $d_{ij}(t)$ is a function of $Q(t)$ and $a_{ij}(t)$ is a random variable independent of $Q(t)$. Its irreducibility is proved in Appendix A. The rest follows from Lemma 1 and Theorem 1. □

4.5. Delay analysis

In this section, we derive the bound on the expected total queue length $\mathbf{E}[\|\bar{Q}\|_1]$ (readily convertible to the corresponding delay bound using Little's Law) for QPS-1 under i.i.d. traffic arrivals using the following moment bound lemma (*i.e.*, Lemma 2) [21, Proposition 2.1.4]. This bound, shown in inequality (28), is identical to that derived in [4,10, Section III.B] for maximal matchings under i.i.d. traffic arrivals. Note this equivalence is not limited to i.i.d. traffic arrivals: As will be shown in Section 5.3, the delay analysis results for general Markovian arrivals [20] derived in [4,10] for maximal matchings (using the stronger "departure inequality" (6)) hold also for QPS-1.

Lemma 2. *Suppose that $\{Y_t\}_{t=0}^{\infty}$ is a positive recurrent Markov chain with countable state space $\mathcal{Y}$. Suppose V, f, and g are non-negative functions on $\mathcal{Y}$ such that,*

$$V(Y_{t+1}) - V(Y_t) \le -f(Y_t) + g(Y_t), \text{ for all } Y_t \in \mathcal{Y}. \tag{27}$$

Then $\mathbf{E}[f(\bar{Y})] \le \mathbf{E}[g(\bar{Y})]$, where $\bar{Y}$ is a random variable with the stationary distribution of the Markov chain $\{Y_t\}_{t=0}^{\infty}$.

Now we derive the following bound on $\mathbf{E}[\|\bar{Q}\|_1]$, which is stronger than the part (II) of Theorem 1.

Theorem 2. *Under the same assumptions and definitions as in Theorem 1, we have*

$$\mathbf{E}[\|\bar{Q}\|_1] \le \frac{1}{2(1-2\rho)} \sum_{i,j} \big(\sigma_{ij}^2 - \lambda_{ij}\Lambda_{ij}^{\dagger} + \lambda_{ij}\big). \tag{28}$$

Proof. We replace function V in Lemma 2 by L, the Lyapunov function used in the proof of Theorem 1 and Y_t by the queue length matrix $Q(t)$. Let $f(Y_t) \triangleq -2\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) + h(Y_t)$, where $h(Y_t) \triangleq 4Na_{max}\|Q(t)\|_1$, and $g(Y_t) \triangleq \sum_{i,j}\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) + h(Y_t)$. It is not hard to check that both $f(\cdot)$ and $g(\cdot)$ are non-negative functions. Inequality (27) holds for V, Y_t, f, g defined above, as it is implied by (18). Furthermore, we have proved before, $\{Q(t)\}_{t=0}^{\infty}$ is a positive recurrent Markov chain whenever the maximum load factor satisfies $\rho < 1/2$. Hence, we have, in steady state,

$$\begin{aligned}
&-2(2\rho-1)\mathbf{E}[\|\bar{Q}\|_1] \\
=&-2(2\rho-1)\mathbf{E}[\|Q(t)\|_1] \\
\le&\mathbf{E}\Big[-2\textstyle\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\Big] \qquad (29)\\
=&\mathbf{E}[f(Y_t) - h(Y_t)] \\
=&\mathbf{E}[f(\bar{Y}) - h(\bar{Y})] \\
=&\mathbf{E}[f(\bar{Y})] - \mathbf{E}[h(\bar{Y})] \\
\le&\mathbf{E}[g(\bar{Y})] - \mathbf{E}[h(\bar{Y})] \qquad (30)\\
=&\mathbf{E}[g(\bar{Y}) - h(\bar{Y})] \\
=&\mathbf{E}\Big[\sum_{i,j}\big(a_{ij}(t) - d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big)\Big] \\
=&\sum_{i,j}\big(\sigma_{ij}^2 - \lambda_{ij}\Lambda_{ij}^{\dagger} + \lambda_{ij}\big). \qquad (31)
\end{aligned}$$

In the above derivation, inequality (29) is because taking expectation on both sides of inequality (20), we have $\mathbf{E}\big[2\sum_{i,j}q_{ij}(t)\big(A^{\dagger}_{ij}(t)-D^{\dagger}_{ij}(t)\big)\big] \le 2(2\rho-1)\mathbf{E}[\|Q(t)\|_1]$. Thus, we obtain

$$-2(2\rho-1)\mathbf{E}[\|Q(t)\|_1] \le \mathbf{E}\big[-2\textstyle\sum_{i,j}q_{ij}(t)\big(A^{\dagger}_{ij}(t)-D^{\dagger}_{ij}(t)\big)\big].$$

Inequality (30) is due to Lemma 2, and (31) is due to (26).

Therefore, we have, in steady state,

$$\mathbf{E}[\|\bar{Q}\|_1] \le \frac{1}{2(1-2\rho)}\sum_{i,j}\big(\sigma^2_{ij}-\lambda_{ij}\Lambda^{\dagger}_{ij}+\lambda_{ij}\big). \quad \square$$

Since, as explained in the proof of Corollary 1, $\{Q(t)\}_{t=0}^{\infty}$ is an irreducible Markov chain under i.i.d. arrivals whose maximum load factor satisfies $\rho < 1/2$, Theorem 2 applies to QPS-1. Hence we obtain the following corollary.

Corollary 2. *The bound on* $\mathbf{E}[\|\bar{Q}\|_1]$ *as stated in inequality* (28) *holds under QPS-1 scheduling, whenever the arrival process is i.i.d. and the maximum load factor satisfies* $\rho < 1/2$.

It is not hard to check (by applying Little's Law) that the average delay (experienced by packets) is bounded by a constant independent of N (*i.e., order-optimal*) for a given maximum load factor $\rho < 1/2$, since the variance σ^2_{ij} for any i, j is assumed to be finite in Section 4.1. For the special case of Bernoulli i.i.d. arrival (when $\sigma^2_{ij} = \lambda_{ij} - \lambda^2_{ij}$), this bound (the RHS) can be further tightened to $\frac{\sum_{i,j}\lambda_{ij}}{1-2\rho}$. This implies, by Little's Law, the following "clean" bound: $\bar{\omega} \le \frac{1}{1-2\rho}$ where $\bar{\omega}$ is the expected delay averaged over all packets transmitting through the switch.

5. Throughput and delay analysis under Markovian arrivals

In this section, we will generalize our results in Section 4 to the more general Markovian arrivals [20].

5.1. Preliminaries

As mentioned earlier the traffic arrival matrix $A(t)$ now is Markovian arrivals. More precisely, for any $1 \le i, j \le N$, the number of arrivals $a_{ij}(t) \triangleq \eta(x_{ij}(t))$ for some non-negative integer-valued function $\eta(\cdot)$, where $\{x_{ij}(t)\}_{t=0}^{\infty}$ is an irreducible, positive recurrent and aperiodic discrete-time Markov chain (DTMC) on a finite state space $\mathcal{X}_{ij} \triangleq \{1, 2, \ldots, \chi_{ij}\}$ for some integer $\chi_{ij} > 0$ and $\{X(t)\}_{t=0}^{\infty}$ is also an irreducible, positive recurrent and aperiodic DTMC where $X(t) \triangleq \big(x_{ij}(t)\big)$. Note that we make no assumption as to whether or not the N^2 Markov chains in $X(t)$ are mutually independent, except where we explicitly state our assumption (e.g., in the part (II) of Theorem 4).

Define the steady-state arrival rate $\lambda_{ij} \triangleq \mathbf{E}[\eta(\bar{x}_{ij})]$, where $\bar{x}_{ij}$ is the stationary distribution that $\{x_{ij}(t)\}_{t=0}^{\infty}$ converges to. Like in [4,10], we assume that the underlying DTMC $\{x_{ij}(t)\}_{t=0}^{\infty}$ for all $1 \le i, j \le N$ is in steady state at time 0, thus each arrival process $\{a_{ij}(t)\}_{t=0}^{\infty}$ is stationary, and therefore, we have, $\mathbf{E}[a_{ij}(t)] = \mathbf{E}[\eta(\bar{x}_{ij})] = \lambda_{ij}$ for any $1 \le i, j \le N$ and any time slot $t \ge 0$. Hence, Fact 2 also holds for the arrival process $A(t)$. Like in Section 4.1, we assume that $a_{ij}(t)$ is upper-bounded by a_{max} for any i, j at any time slot t. We further assume that given any integer $\tau > 0$, there exists some $\beta > 0$ such that $\mathbf{P}[A(t') = \mathbf{0}] \ge \beta$ for any $t' \in \{t, t+1, \ldots, t+\tau\}$, where $\mathbf{0}$ is the $N \times N$ matrix with all its entries equal to 0. In other words, the switch could, with a nonzero probability, have no packet arrivals to any of its VOQs during time slots $t, t+1, \ldots, t+\tau$.

As mentioned before, we will prove in this section that, QPS-1 has the same provable throughput and delay guarantees as using maximal matchings under Markovian arrival processes $A(t)$. Before that, we state a fact concerning Markovian arrival processes $A(t)$.

Fact 4. *Given a Markovian arrival process* $\{a_{ij}(t)\}_{t=0}^{\infty}$ *defined above, let* $H(t)$ *be the history of all arrivals (for all VOQs) up to but excluding time slot* t*, then regardless of the past history* $H(t)$*, we have,*

$$\mathbf{E}\big[a_{ij}(t) \mid H(t-T)\big] \le \lambda_{ij} + \zeta_{ij}(T), \tag{32}$$

where $\zeta_{ij}(T)$ *is some non-negative function of* $T \in \mathbb{N}$ *such that it converges to* 0 *exponentially fast as* T *approaches to* ∞.

The proof of the above fact can be found in Appendix A.2 of [20].

For any non-negative integer T, we define

$$\xi(T) \triangleq \max_{l,w}\left[\Lambda^{\dagger}_{lw} + \sum_{(i,j)\in Q^{\dagger}_{lw}}\zeta_{ij}(T)\right]. \tag{33}$$

For any $\rho < 1/2$, we define

$$K_{\rho} \triangleq \min\{k \in \mathbb{N}_+ : \xi(T) < 1 \text{ for } T \ge k\}. \tag{34}$$

Note that such a K_ρ exists because $\sum_{(i,j)\in Q^{\dagger}_{lw}}\zeta_{ij}(T) \to 0$ as $T \to \infty$ for any l, w and when $\rho < 1/2$, we have $\Lambda^{\dagger}_{lw} < 1$ for any l, w.

5.2. Throughput analysis

We now present the throughput result under Markovian arrivals, which generalizes the throughput result under i.i.d. arrivals, *i.e.*, Theorem 1.

Theorem 3. *Let $A(t)$ be Markovian arrivals determined by the underlying DTMC $\{X(t)\}_{t=0}^{\infty}$ as described before in Section 5.1. Suppose the joint process $\{(Q(t), X(t))\}_{t=0}^{\infty}$ under Markovian arrivals $A(t)$ is an irreducible Markov chain. Also, assume that the departure process of $\{Q(t)\}_{t=0}^{\infty}$ satisfies the weaker "departure inequality" (7). Then whenever the maximum load factor satisfies $\rho < 1/2$, the switching system is stable in the following sense: (I) The Markov chain $\{(Q(t), X(t))\}_{t=0}^{\infty}$ is positive recurrent and hence the queueing process $\{Q(t)\}_{t=0}^{\infty}$ converges in distribution to a stationary distribution $\bar{Q}$; (II) The first moment of $\bar{Q}$ is finite.*

Proof. See Appendix B. □

It is not hard to check that Theorem 3 applies to QPS-1. The proof can be found in Appendix C. Therefore, QPS-1 can also attain at least 50% throughput under Markovian arrivals.

5.3. Delay analysis

Now, we present bounds on the expected total queue length $\mathbf{E}[\|\bar{Q}\|_1]$ for QPS-1 under Markovian arrivals.

Theorem 4. *Under the same assumptions and definitions as in Theorem 3, we have: (I) The average (total) queue length is upper-bounded as follows*

$$\mathbf{E}[\|\bar{Q}\|_1] \le \frac{1}{2(1-\xi(K_\rho))}\left(\sum_{i,j}\left(\mathbf{E}[a_{ij}(0)A_{ij}^{\dagger}(0)]+\lambda_{ij}\right) +2\sum_{i,j}\mathbf{E}\left[\sum_{k=0}^{K_\rho-1} a_{ij}(k-K_\rho)A_{ij}^{\dagger}(0)\right]\right), \tag{35}$$

where $\xi(\cdot)$ and K_ρ are defined in (33) and (34) respectively.

(II) If the arrival processes $a_{ij}(t)$ for different i, j are independent of each other, then the above queue length bound can be further simplified as follows

$$\mathbf{E}[\|\bar{Q}\|_1] \le \frac{1}{2(1-\xi(K_\rho))}\left(\sum_{i,j}\left(\sigma_{ij}^2+\lambda_{ij}\Lambda_{ij}^{\dagger}+\lambda_{ij}\right)+2\sum_{i,j}\sum_{k=1}^{K_\rho}\left(\lambda_{ij}\Lambda_{ij}^{\dagger}+\theta_{ij}(k)\right)\right), \tag{36}$$

where $\sigma_{ij}^2 \triangleq \mathbf{E}\left[a_{ij}^2(t)\right]-\lambda_{ij}^2$ is the steady-state variance of $a_{ij}(t)$ and $\theta_{ij}(k) \triangleq \mathbf{E}\left[a_{ij}(k+t)a_{ij}(t)\right]-\lambda_{ij}^2$ is the auto-correlation in $a_{ij}(t)$ in steady state.

Proof. See Appendix D. □

Similarly, we have that the bounds on $\mathbf{E}[\|\bar{Q}\|_1]$ as stated in Theorem 4 hold under QPS-1 scheduling, whenever the arrival processes are Markovian arrivals and the maximum load factor satisfies $\rho < 1/2$.

6. Evaluation

In this section, we evaluate, through simulations, the performance of QPS-r under various load conditions and traffic patterns. The main purpose of this section to show that QPS-r performs as well as maximal matching algorithms not just in theory. Hence, we do not compare QPS-r with the two recent iterative algorithms in switching[1]: RR/LQF (Round Robin combined with Longest Queue First) [22] and HRF (Highest Rank First) [23]. Instead, we compare its performance with that of iSLIP [7], a refined and optimized representative parallel maximal matching algorithm (adapted for switching). The performance of the MWM (Maximum Weighted Matching) is also included in the comparison as a benchmark. Our simulations show conclusively that QPS-1 (running 1 iteration) performs very well inside the provable stability region (more precisely, with no more than 50% offered load), and that QPS-3 (running 3 iterations) has comparable throughput and delay performances as iSLIP (running $\log_2 N$ iterations), which has a much higher per-port computational complexity of $O(\log^2 N)$.

[1] Note that they are shown to have reasonably good empirical throughput and delay performance over round-robin-friendly workloads such as uniform and hot-spot traffic when running 1 or 2 iterations. However, as described in Section 7, they need to run up to N iterations to provably attain at least 50% throughput.

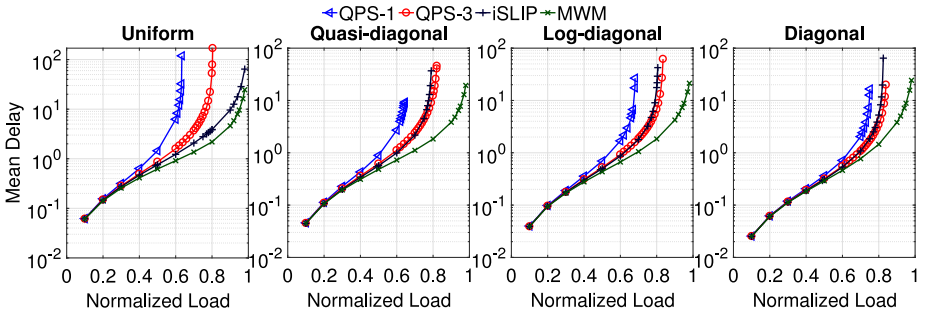

Fig. 2. Mean delays of QPS-1, QPS-3, iSLIP, and MWM under the four traffic patterns.

6.1. Simulation setup

In our simulations, we fix the number of input/output ports, N to 64. Later, we investigate how the mean delay performances of these algorithms scale with respect to N and the findings are reported in Section 6.3. To measure throughput and delay accurately, we assume each VOQ has an infinite buffer size and hence there is no packet drop at any input port. Each simulation run is guided by the following stopping rule [24,25]: The number of time slots simulated is the larger between $500N^2$ and that is needed for the difference between the estimated and the actual average delays to be within 0.01 with probability at least 0.98.

We assume in our simulations that each traffic arrival matrix $A(t)$ is Bernoulli i.i.d. with its traffic rate matrix Λ being equal to the product of the offered load and a traffic pattern matrix (defined next). Similar Bernoulli arrivals were studied in [7–9]. Later, in Section 6.2.2, we will look at burst traffic arrivals. Note that only synthetic traffic (instead of that derived from packet traces) is used in our simulations because, to the best of our knowledge, there is no meaningful way to combine packet traces into switch-wide traffic workloads. The following four standard types of normalized (with each row or column sum equal to 1) traffic patterns are used: (I) *Uniform*: packets arriving at any input port go to each output port with probability $\frac{1}{N}$. (II) *Quasi-diagonal*: packets arriving at input port i go to output port $j=i$ with probability $\frac{1}{2}$ and go to any other output port with probability $\frac{1}{2(N-1)}$. (III) *Log-diagonal*: packets arriving at input port i go to output port $j=i$ with probability $\frac{2^{(N-1)}}{2^N-1}$ and go to any other output port j with probability equal to $\frac{1}{2}$ of the probability of output port $j-1$ (note: output port 0 equals output port N). (IV) *Diagonal*: packets arriving at input port i go to output port $j=i$ with probability $\frac{2}{3}$, or go to output port $(i \bmod N)+1$ with probability $\frac{1}{3}$. These traffic patterns are listed in order of how skewed the volumes of traffic arrivals to different output ports are: from uniform being the least skewed, to diagonal being the most skewed.

6.2. Throughput and delay performances

6.2.1. Bernoulli arrivals

We first compare the throughput and delay performances of QPS-1 (1 iteration), QPS-3 (3 iterations), iSLIP ($\log_2 64 = 6$ iterations), and MWM (length of VOQ as the weight measure). Note that we have also investigated how the performance of QPS-r scales with respect to r, the results are reported in Section 6.4. Fig. 2 shows their mean delays (in number of time slots) under the aforementioned four traffic patterns respectively. Each subfigure shows how the mean delay (on a *log scale* along the y-axis) varies with the offered load (along the x-axis). We make three observations from Fig. 2. First, Fig. 2 clearly shows that, when the offered load is no larger than 0.5, QPS-1 has low average delays (*i.e.*, more than just being stable) that are close to those of iSLIP and MWM, under all four traffic patterns. Second, the maximum sustainable throughputs (where the delays start to "go through the roof" in the subfigures) of QPS-1 are roughly 0.634, 0.645, 0.681, and 0.751 respectively, under the four traffic patterns respectively; they are all comfortably larger than the 50% provable lower bound. Third, the throughput and delay performances of QPS-3 and iSLIP are comparable: The former has slightly better delay performances than the latter under all four traffic patterns except the uniform. QPS-3 performs worse than iSLIP under the uniform traffic since in this case queue-proportional sampling degenerates to uniform random sampling and as a result loses the advantage of implicitly making use of queue length information.

6.2.2. Bursty arrivals

In real networks, packet arrivals are likely to be bursty. In this section, we evaluate the performances of QPS, iSLIP and MWM under bursty traffic, generated by a two-state ON-OFF arrival process, another special case of Markovian arrivals, described in [4]. The durations of each ON (burst) stage and OFF (no burst) stage are geometrically distributed: the probabilities that the ON and OFF states last for $t \geq 0$ time slots are given by

$$P_{ON}(t) = p(1-p)^t \text{ and } P_{OFF}(t) = q(1-q)^t,$$

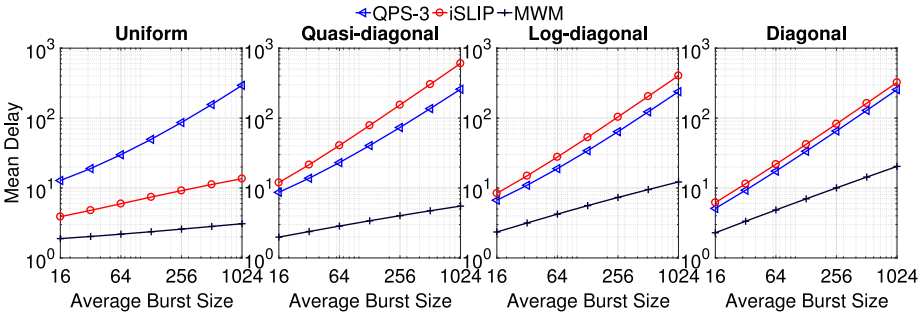

Fig. 3. Mean delays of QPS-3 against iSLIP and MWM under bursty traffic.

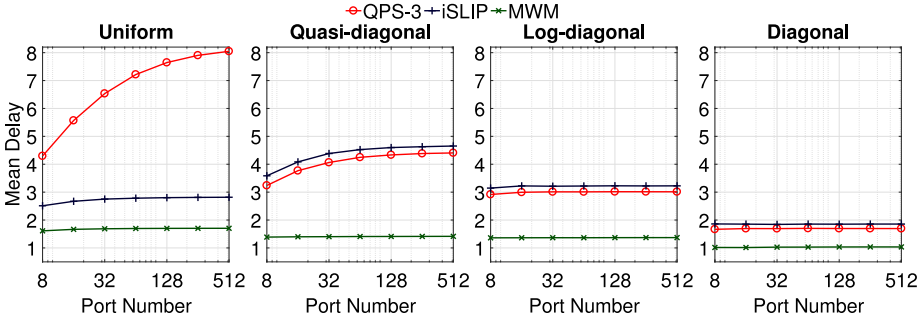

Fig. 4. Mean delays of QPS-3, iSLIP, and MWM scaling with port number N.

with the parameters $p, q \in (0, 1)$ respectively. As such, the average duration of the ON and OFF states are $(1-p)/p$ and $(1-q)/q$ time slots respectively.

In an OFF state, an incoming packet's destination (*i.e.,* output port) is generated according to the corresponding load matrix. In an ON state, all incoming packet arrivals to an input port would be destined to the same output port, thus simulating a burst of packet arrivals. By controlling p, we can control the desired average burst size while by adjusting q, we can control the load of the traffic.

We have evaluated the mean delay performances of QPS-3, iSLIP and MWM, with the average burst size ranging from 16 to 1,024 packets. We have simulated various offered loads, but here we only present the results, shown in Fig. 3, under an offered load of 0.75; other offered loads lead to similar conclusions. Fig. 3 clearly shows that QPS-3 outperforms iSLIP (under all traffic patterns except the uniform), by an increasingly wider margin in both absolute and relative terms as the average burst size becomes larger.

6.3. How mean delay scales with N

Fig. 4 shows how the mean delays of QPS-3, iSLIP (running $\log_2 N$ iterations given any N), and MWM scale with the number of input/output ports N, under the four different traffic patterns. We have simulated the following different values of N: $N = 8, 16, 32, 64, 128, 256, 512$. In all these plots, the offered load is 0.75 (like in Section 6.2.2, we have also simulated other offered loads. They are omitted here, because they lead to similar conclusions), which is quite high compared to the maximum sustainable throughputs of QPS-3 and iSLIP (shown in Fig. 2) under these four traffic patterns. Fig. 4 shows that the mean delays of QPS-3 are slightly lower (*i.e.*, better) than those of iSLIP under all traffic patterns except the uniform. In addition, the mean delay curves of QPS-3 remain almost flat (*i.e.*, constant) under log-diagonal and diagonal traffic patterns. Although they increase with N under uniform and quasi-diagonal traffic patterns, they eventually almost flatten out when N gets larger (say when $N \geq 128$). These delay curves show that QPS-3, which runs only 3 iterations, deliver slightly better delay performances, under all traffic patterns except the uniform, than iSLIP (a refined and optimized parallel maximal matching algorithm adapted for switching), which runs $\log_2 N$ iterations with each iteration has $O(\log_2 N)$ computational complexity.

6.4. How mean delay of QPS-r scales with r

Fig. 5 presents the mean delay performance of QPS-r, with $r = 1, 2, 3, 4$, as a function of the offered loads, under Bernoulli i.i.d. arrivals with the four traffic patterns. It shows that both the maximum sustainable throughput and the mean delay performance improve as r increases. However, as $r > 3$, the improvement becomes marginal.

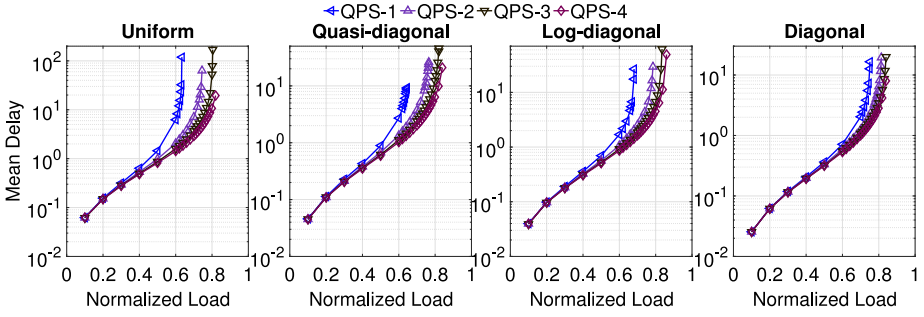

Fig. 5. Mean delays of QPS-r ($r = 1, 2, 3, 4$) under the four traffic patterns.

7. Related work

Scheduling in crossbar switches is a well-studied problem with a large amount of literature. In this section, we provide only a brief survey of prior work that is directly related to ours, focusing on those we have not described earlier.

Iterative algorithms that compute maximal matchings. As mentioned earlier, maximal matchings have long been recognized as a cost-effective family in switching. Among various types of algorithms that compute maximal matchings, the family of parallel iterative algorithms [22,23,26–29] is widely adopted. Parallel iterative algorithms compute a maximal matching via multiple iterations of message exchanges between the input and output ports. Generally, each iteration contains three stages: request, grant, and accept. In the request stage, each input port sends requests to output ports. In the grant stage, each output port, upon receiving requests from multiple input ports, grants to one. Finally, in the accept stage, each input port, upon receiving grants from multiple output ports, accepts one. Unfortunately, all these parallel iterative algorithms in switching require up to N iterations to guarantee that the resulting matching is a maximal matching. In other words, they need up to N iterations to achieve the same provable throughput and delay performance guarantees as QPS-1 (running 1 iteration).

Other algorithms that have performance guarantees. Several serial randomized algorithms, starting with TASS [30] and culminating in SERENA [9], have been proposed that have a total computational complexity of only $O(N)$ yet can provably attain 100% throughput; SERENA, the best among them, also delivers a good empirical delay performance. However, this $O(N)$ complexity is still too high for scheduling high-line-rate high-radix switches, and none of them has been successfully parallelized (*i.e.*, converted to a parallel iterative algorithm) yet, except that SERENA was recently parallelized in [31] that reduces the $O(N)$ computational complexity to $O(\log N)$ per port, which is still quite high for high-line-rate high-radix switches.

In [32], a crossbar scheduling algorithm specialized for switching variable-length packets was proposed, that has $O(1)$ total computational complexity. Although this algorithm can provably attain 100% throughput, its delay performance is poor. For example, as shown in [8], its average delays, under the aforementioned four standard traffic patterns, are roughly 3 orders of magnitudes higher than those of SERENA [9] even under a moderate offered load of 0.6.

8. Conclusion

In this work, we propose QPS-r, a parallel iterative switching algorithm with $O(1)$ computational complexity per port. We prove, through Lyapunov stability analysis, that it achieves the same throughput and delay guarantees in theory for both i.i.d. arrivals and the more general Markovian arrivals, and demonstrate through simulations that it has comparable performances in practice as the family of maximal matching algorithms (adapted for switching); maximal matching algorithms are much more expensive computationally (at least $O(\log N)$ iterations and a total of $O(\log^2 N)$ per-port computational complexity). These salient properties make QPS-r an excellent candidate algorithm that is fast enough computationally and can deliver acceptable throughput and delay performances for high-link-rate high-radix switches.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. CNS-1909048, CNS-2007006, and CCF-1850439.

Appendix A. Irreducibility of $\{Q(t)\}_{t=0}^{\infty}$

We have shown that the queueing process $\{Q(t)\}_{t=0}^{\infty}$ is a Markov chain in the proof of Corollary 1. Now we prove that this Markov is irreducible using the same approach as used in [8]. Here we provide only a proof sketch. We show that the queueing process $Q(t)$, starting from any state (*i.e.,* VOQ lengths) it is currently in, will with a positive probability return to the **0** state, in which all VOQs have a length of 0, in a finite number of steps (*i.e.,* time slots). To show this property, we claim that, for any integer $\tau > 0$, the switch could, with a nonzero probability, have no packet arrivals to any of its VOQs during time slots $t, t+1, \ldots, t+\tau$. This claim is true because, the arrival process $A(t)$ is i.i.d., and for any $1 \le i \le N$ and $1 \le j \le N$, we have $\beta_{ij} \triangleq P[a_{ij}(t) = 0] > 0$ (since the maximum load factor $\rho < 1/2$). Hence, when there are no packet arrivals during time slots $t, t+1, \ldots, t+\tau$, which happens with a nonzero probability, QPS-1 can clear all the queues during this period, with a sufficiently large τ, and return the queueing process $Q(t)$ to the **0** state. Therefore, the Markov chain is irreducible.

Appendix B. Proof of Theorem 3

In this section, we prove Theorem 3. The proof follows the same outline as the proof of Theorem 1 with some necessary modifications to allow for Markovian arrivals. Note that we prove only (I) here, since Theorem 4 that we will shortly prove implies (II).

For notational convenience, we define $Z(t) \triangleq \big(Q(t), X(t)\big)$. Unlike in the proof of Theorem 1 where we use the 1-step Foster–Lyapunov stability criterion [21, Proposition 2.1.1], here we use the n-step (multiple-step) Foster–Lyapunov stability criterion [33, Theorem 2.2.4]. We consider again the following Lyapunov function: $L\big(Z(t)\big) = \sum_{i,j} q_{ij}(t) Q_{ij}^{\dagger}(t)$, and show that its n-step drift, which is defined as $\mathbf{E}\big[L\big(Z(t+n)\big) - L\big(Z(t)\big) \mid Z(t)\big]$, is negative except in a finite set.

To compute the n-step drift, we first compute,

$$L\big(Z(t+n)\big) - L\big(Z(t)\big) = \sum_{i,j} q_{ij}(t+n) Q_{ij}^{\dagger}(t+n) - \sum_{i,j} q_{ij}(t) Q_{ij}^{\dagger}(t). \tag{B.1}$$

Using (1) and (2), and definitions of $D_{ij}^{\dagger}(t)$ and $A_{ij}^{\dagger}(t)$, we have,

$$q_{ij}(t+n) = q_{ij}(t) - \sum_{k=0}^{n-1} d_{ij}(t+k) + \sum_{k=0}^{n-1} a_{ij}(t+k), \tag{B.2}$$

and

$$Q_{ij}^{\dagger}(t+n) = Q_{ij}^{\dagger}(t) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k) + \sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k). \tag{B.3}$$

Thus, the term $q_{ij}(t+n) Q_{ij}^{\dagger}(t+n)$ in the first term on the RHS of (B.1) can be written as

$$\begin{aligned}
&q_{ij}(t+n) Q_{ij}^{\dagger}(t+n) \\
=&q_{ij}(t) Q_{ij}^{\dagger}(t) + q_{ij}(t)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \\
&+ Q_{ij}^{\dagger}(t)\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big) \\
&+ \Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big).
\end{aligned} \tag{B.4}$$

Substituting (B.4) into (B.1), we have

$$\begin{aligned}
&L\big(Z(t+n)\big) - L\big(Z(t)\big) \\
=&\sum_{i,j} q_{ij}(t)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \\
&+ \sum_{i,j} Q_{ij}^{\dagger}(t)\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big) \\
&+ \sum_{i,j}\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big).
\end{aligned} \tag{B.5}$$

Focusing on the second term above and using (1), we have

$$\begin{aligned}
&\sum_{i,j} Q_{ij}^{\dagger}(t)\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\\
=&\sum_{i,j}\sum_{(l,w)\in Q_{ij}^{\dagger}} q_{lw}(t)\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\\
=&\sum_{l,w}\sum_{(i,j)\in Q_{lw}^{\dagger}} q_{lw}(t)\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\\
=&\sum_{l,w} q_{lw}\Big(\sum_{(i,j)\in Q_{lw}^{\dagger}}\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{(i,j)\in Q_{lw}^{\dagger}}\sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\\
=&\sum_{l,w} q_{lw}\Big(\sum_{k=0}^{n-1}\sum_{(i,j)\in Q_{lw}^{\dagger}} a_{ij}(t+k) - \sum_{k=0}^{n-1}\sum_{(i,j)\in Q_{lw}^{\dagger}} d_{ij}(t+k)\Big)\\
=&\sum_{l,w} q_{lw}\Big(\sum_{k=0}^{n-1} A_{lw}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{lw}^{\dagger}(t+k)\Big)\\
=&\sum_{i,j} q_{ij}(t)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big).
\end{aligned} \tag{B.6}$$

Hence, the drift can be written as

$$\begin{aligned}
&\mathbf{E}\Big[L\big(Z(t+n)\big) - L\big(Z(t)\big) \mid Z(t)\Big]\\
=&\mathbf{E}\Big[2\sum_{i,j} q_{ij}(t)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \mid Z(t)\Big]\\
&+\mathbf{E}\Big[\sum_{i,j}\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \mid Z(t)\Big].
\end{aligned} \tag{B.7}$$

Since $0 \le a_{ij}(t) \le a_{max}$ and $0 \le d_{ij}(t) \le 1$, for any i, j and t, we have

$$\begin{aligned}
&\mathbf{E}\Big[\sum_{i,j}\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) - \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \mid Z(t)\Big]\\
\le&\mathbf{E}\Big[\sum_{i,j}\Big(\sum_{k=0}^{n-1} a_{ij}(t+k) + \sum_{k=0}^{n-1} d_{ij}(t+k)\Big)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) + \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \mid Z(t)\Big]\\
\le&\mathbf{E}\Big[\sum_{i,j}\big(na_{max} + n\big)\big((2N-1)na_{max} + (2N-1)n\big) \mid Z(t)\Big]\\
=&C_1(n).
\end{aligned} \tag{B.8}$$

where $C_1(n) \triangleq (2N-1)n^2N^2(a_{max}+1)^2$.

Now we claim there exists some $n > K_\rho$ (where K_ρ is defined in (34)) such that the following inequality holds, which we will prove shortly.

$$\begin{aligned}
&\mathbf{E}\Big[2\sum_{i,j} q_{ij}(t)\Big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\Big) \mid Z(t)\Big]\\
\le&2n\Big(\xi_\rho - 1 + \frac{(2N-1)K_\rho a_{max} - K_\rho\xi_\rho}{n}\Big)\|Q(t)\|_1 + C_2(n),
\end{aligned} \tag{B.9}$$

where K_ρ is defined in (34), $\xi_\rho \triangleq \max\{\xi(k) : k \in \{K_\rho, K_\rho+1, \ldots, n-1\}\}$ where $\xi(\cdot)$ is defined (33) and $C_2(n) \triangleq (2a_{max}+1)(n-1)nN^2$.

Define $n_0 \triangleq \min\{n \in \mathbb{N}_+ \text{ and } n > K_\rho : \xi_\rho - 1 + \frac{(2N-1)K_\rho a_{max} - K_\rho\xi_\rho}{n} < 0\}$. Such an n_0 should exist because $\xi_\rho < 1$ and $\frac{(2N-1)K_\rho a_{max} - K_\rho\xi_\rho}{n} \to 0$ as $n \to \infty$. Define $\varepsilon(n_0) \triangleq -2n_0\big(\xi_\rho - 1 + \frac{(2N-1)K_\rho a_{max} - K_\rho\xi_\rho}{n_0}\big)$, clearly $\varepsilon(n_0) > 0$. With inequalities

(B.8) and (B.9) substituted into (B.7), we have

$$\begin{aligned}&\mathbf{E}\big[L\big(Z(t+n_0)\big)-L\big(Z(t)\big)\mid Z(t)\big]\\ \le&-\varepsilon(n_0)\|Q(t)\|_1+C_1(n_0)+C_2(n_0)\\ \le&-\varepsilon(n_0)(\|Z(t)\|_1-\chi N^2)+C_1(n_0)+C_2(n_0),\end{aligned}\tag{B.10}$$

where $\chi \triangleq \max_{i,j}\{\chi_{ij}\}$.

Inequality (B.10) is due to $\varepsilon(n_0) > 0$ and $x_{ij}(t) \le \chi_{ij}$ (since each $\{x_{ij}(t)\}_{t=0}^{\infty}$ is a DTMC on a finite state space of $\{1, 2, \ldots, \chi_{ij}\}$). Because $\varepsilon(n_0) > 0$, there exist $B, \epsilon > 0$ such that, whenever $\|Z(t)\|_1 > B$,

$$\mathbf{E}\big[L\big(Z(t+n_0)\big)-L\big(Z(t)\big)\mid Z(t)\big]\le-\epsilon.$$

So Theorem 3 follows using n_0-step Foster–Lyapunov criterion [33, Theorem 2.2.4].

Now we proceed to prove inequality (B.9). By simple calculations, we have

$$\begin{aligned}&\mathbf{E}\Big[2\sum_{i,j}q_{ij}(t)\big(\sum_{k=0}^{n-1}A_{ij}^{\dagger}(t+k)-\sum_{k=0}^{n-1}D_{ij}^{\dagger}(t+k)\big)\mid Z(t)\Big]\\ =&2\Big(\sum_{i,j}\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]-\sum_{i,j}\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1}D_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\Big).\end{aligned}\tag{B.11}$$

Focusing on the first term in the parentheses above, for the conditional expectation $\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]$, we have

$$\begin{aligned}&\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\\ =&\mathbf{E}\Big[q_{ij}(t)\big(\sum_{k=0}^{K_\rho-1}A_{ij}^{\dagger}(t+k)+\sum_{k=K_\rho}^{n-1}A_{ij}^{\dagger}(t+k)\big)\mid Z(t)\Big]\\ =&\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{K_\rho-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]+\mathbf{E}\Big[q_{ij}(t)\sum_{k=K_\rho}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big].\end{aligned}\tag{B.12}$$

Since $0 \le a_{ij}(t) \le a_{max}$ for any i, j and t, we have the first expectation above, $\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{K_\rho-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]$, is bounded by

$$\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{K_\rho-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\le(2N-1)K_\rho a_{max}q_{ij}(t).\tag{B.13}$$

Using (33), (34), and Fact 4, we have, for any $k \ge K_\rho$,

$$\begin{aligned}&\mathbf{E}\Big[q_{ij}(t)A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\\ =&q_{ij}(t)\mathbf{E}\Big[A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\\ \le&q_{ij}(t)\big(\Lambda_{ij}^{\dagger}+\sum_{(l,w)\in Q_{ij}^{\dagger}}\zeta_{lw}(k)\big)\\ \le&\xi(k)q_{ij}(t)\\ \le&\xi_\rho q_{ij}(t).\end{aligned}$$

Therefore, the second expectation on the RHS of (B.12), $\mathbf{E}\Big[q_{ij}(t)\sum_{k=K_\rho}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]$, can be bounded by

$$\mathbf{E}\Big[q_{ij}(t)\sum_{k=K_\rho}^{n-1}A_{ij}^{\dagger}(t+k)\mid Z(t)\Big]\le(n-K_\rho)\xi_\rho q_{ij}(t).\tag{B.14}$$

Now, we proceed to bound the second term in the parentheses on the RHS of (B.11), $\sum_{i,j}\mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1}D_{ij}^{\dagger}(t+k)\mid Z(t)\Big]$.

Using (2), we have $q_{ij}(t+k) = q_{ij}(t) - \sum_{\tau=0}^{k-1}d_{ij}(t+\tau) + \sum_{\tau=0}^{k-1}a_{ij}(t+\tau)$ for any integer $k \ge 0$. Because $d_{ij}(t) \ge 0$ and $a_{ij}(t) \le a_{max}$ for any i, j and t, $q_{ij}(t) = q_{ij}(t+k) + \sum_{\tau=0}^{k-1}d_{ij}(t+\tau) - \sum_{\tau=0}^{k-1}a_{ij}(t+\tau) \ge q_{ij}(t+k) - ka_{max}$ for any integer

$k \geq 0$. Thus, for any integer $k \geq 0$, we have

$$\begin{aligned}
&\sum_{i,j} \mathbf{E}\Big[q_{ij}(t)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] \\
\geq &\sum_{i,j} \mathbf{E}\Big[\big(q_{ij}(t+k) - ka_{max}\big)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] \\
= &\sum_{i,j}\Big(\mathbf{E}\Big[q_{ij}(t+k)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] - \mathbf{E}\Big[ka_{max}D_{ij}^{\dagger}(t+k) \mid Z(t)\Big]\Big) \\
\geq &\sum_{i,j}\Big(\mathbf{E}\Big[q_{ij}(t+k)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] - 2ka_{max}\Big) \qquad \text{(B.15)} \\
= &\sum_{i,j} \mathbf{E}\Big[q_{ij}(t+k)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] - 2ka_{max}N^2 \\
= &\sum_{i,j} \mathbf{E}\Big[\mathbf{E}\Big[q_{ij}(t+k)D_{ij}^{\dagger}(t+k) \mid Q(t+k)\Big] \mid Z(t)\Big] - 2ka_{max}N^2 \\
= &\mathbf{E}\Big[\sum_{i,j} \mathbf{E}\Big[q_{ij}(t+k)D_{ij}^{\dagger}(t+k) \mid Q(t+k)\Big] \mid Z(t)\Big] - 2ka_{max}N^2 \\
\geq &\mathbf{E}\Big[\sum_{i,j} q_{ij}(t+k) \mid Z(t)\Big] - 2ka_{max}N^2 \qquad \text{(B.16)} \\
= &\mathbf{E}\Big[\sum_{i,j}\big(q_{ij}(t) - \sum_{\tau=0}^{k-1} d_{ij}(t+\tau) + \sum_{\tau=0}^{k-1} a_{ij}(t+\tau)\big) \mid Z(t)\Big] - 2ka_{max}N^2 \\
\geq &\mathbf{E}\Big[\sum_{i,j}\big(q_{ij}(t) - k\big) \mid Z(t)\Big] - 2ka_{max}N^2 \qquad \text{(B.17)} \\
= &\sum_{i,j}\big(q_{ij}(t) - k\big) - 2ka_{max}N^2 \\
= &\|Q(t)\|_1 - (2a_{max}+1)kN^2. \qquad \text{(B.18)}
\end{aligned}$$

Inequality (B.15) is due to $D_{ij}^{\dagger}(t) \leq 2$ for any i, j and t. Inequality (B.16) is due to the weaker departure inequality (7) and inequality (B.17) is due to $d_{ij}(t) \leq 1$ and $a_{ij}(t) \geq 0$ for any i, j and t.

Using inequality (B.18), we have

$$\begin{aligned}
&\sum_{i,j} \mathbf{E}\Big[q_{ij}(t)\sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] \\
= &\sum_{k=0}^{n-1}\sum_{i,j} \mathbf{E}\Big[q_{ij}(t)D_{ij}^{\dagger}(t+k) \mid Z(t)\Big] \\
\geq &\sum_{k=0}^{n-1}\big(\|Q(t)\|_1 - (2a_{max}+1)kN^2\big) \\
= &n\|Q(t)\|_1 - (2a_{max}+1)(n-1)nN^2/2 \qquad \text{(B.19)}
\end{aligned}$$

Now, using (B.11) and (B.12) and inequalities (B.13), (B.14) and (B.19), we have

$$\begin{aligned}
&\mathbf{E}\Big[2\sum_{i,j} q_{ij}(t)\big(\sum_{k=0}^{n-1} A_{ij}^{\dagger}(t+k) - \sum_{k=0}^{n-1} D_{ij}^{\dagger}(t+k)\big) \mid Z(t)\Big] \\
\leq &2\Big((2N-1)K_\rho a_{max}\|Q(t)\|_1 + (n-K_\rho)\xi_\rho\|Q(t)\|_1 \\
&- \Big(n\|Q(t)\|_1 - (2a_{max}+1)(n-1)nN^2/2\Big)\Big) \\
= &2n\Big(\xi_\rho - 1 + \frac{(2N-1)K_\rho a_{max} - K_\rho\xi_\rho}{n}\Big)\|Q(t)\|_1 + C_2(n), \qquad \text{(B.20)}
\end{aligned}$$

where $C_2(n) = (2a_{max}+1)(n-1)nN^2$.

Appendix C. Theorem 3 Applies to QPS-1

In this section, we prove that Theorem 3 applies to QPS-1. More precisely, under Markovian arrivals whenever the maximum load factor satisfies $\rho < 1/2$, QPS-1 is stable in the following sense: The resulting joint process $\{(Q(t), X(t))\}_{t=0}^{\infty}$ is a positive recurrent Markov chain and the stationary distribution $\bar{Q}$ of the queueing process $\{(Q(t))\}_{t=0}^{\infty}$ has finite first moment.

The joint process $\{(Q(t), X(t))\}_{t=0}^{\infty}$ is a Markov chain, by the following two facts. First, $X(t)$ is a Markov chain so $X(t)$ only depends on $X(t-1)$. Second, by (2), $Q(t)$ depends on only $Q(t-1)$, $A(t-1)$ and $D(t-1)$, where $A(t-1)$ is a function of only $X(t-1)$ and $D(t-1)$ is a function of only $Q(t-1)$.

The reasoning for the irreducibility of this Markov chain $\{(Q(t), X(t))\}_{t=0}^{\infty}$ is almost identical to those in Appendix A, so we omit it here for brevity.

Appendix D. Proof of Theorem 4

Proof of Part (I). Like in Appendix B, we let $Z(t) \triangleq (Q(t), X(t))$. Like in the proof of Theorem 2, we replace function V in Lemma 2 by L, the Lyapunov function used in the proof of Theorem 3 and Y_t by $Z(t)$. We define nonnegative functions f and g in the same way as in the proof of Theorem 2. That is, $f(Y_t) \triangleq -2\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t) - D_{ij}^{\dagger}(t)\big) + h(Y_t)$, where $h(Y_t) \triangleq 4Na_{max}\|Q(t)\|_1$, and $g(Y_t) \triangleq \sum_{i,j}\big(a_{ij}(t)-d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\big) + h(Y_t)$. Like that explained in the proof of Theorem 2, Y_t, V, f, and g thus defined satisfy a stronger form of (27), with "$\le$" replaced by "=".

Now we claim that the following two inequalities hold in steady state of the positive recurrent Markov chain $\{Z(t)\}_{t=0}^{\infty}$. We will prove both shortly.

$$\begin{aligned}&\mathbf{E}\big[2\textstyle\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\big)\big]\\ \le& 2\left((\xi(K_\rho)-1)\mathbf{E}[\|\bar{Q}\|_1]+\textstyle\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(k-K_\rho)A_{ij}^{\dagger}(0)\Big]\right),\end{aligned} \tag{D.1}$$

$$\mathbf{E}\big[\textstyle\sum_{i,j}\big(a_{ij}(t)-d_{ij}(t)\big)\big(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\big)\big] \le \sum_{i,j}\Big(\mathbf{E}[a_{ij}(0)A_{ij}^{\dagger}(0)] + \lambda_{ij}\Big), \tag{D.2}$$

where $\xi(\cdot)$ and K_ρ are defined in (33) and (34) respectively. We note that the counterpart of inequality (D.2) in the proof of Theorem 2 is an equality (i.e., (31)), because the i.i.d. property there allows a tighter derivation. As a result, the delay bound (35) in Theorem 4 is less tight than the delay bound (28) in Theorem 2.

Assuming that inequalities (D.1) and (D.2) both hold, we now prove part (I). Using similar reasoning as in the proof of Theorem 2, we have

$$\begin{aligned}&-2\left((\xi(K_\rho)-1)\mathbf{E}[\|\bar{Q}\|_1]+\sum_{i,j}\mathbf{E}\left[\sum_{k=0}^{K_\rho-1} a_{ij}(k-K_\rho)A_{ij}^{\dagger}(0)\right]\right)\\ \le&\mathbf{E}[f(\bar{Y})-h(\bar{Y})]\\ \le&\mathbf{E}[g(\bar{Y})-h(\bar{Y})]\\ \le&\sum_{i,j}\Big(\mathbf{E}[a_{ij}(0)A_{ij}^{\dagger}(0)]+\lambda_{ij}\Big).\end{aligned}$$

Therefore, we have, in steady state,

$$\begin{aligned}\mathbf{E}[\|\bar{Q}\|_1] \le \frac{1}{2(1-\xi(K_\rho))}&\left(\sum_{i,j}\Big(\mathbf{E}[a_{ij}(0)A_{ij}^{\dagger}(0)]+\lambda_{ij}\Big)\right.\\ &\left.+2\sum_{i,j}\mathbf{E}\left[\sum_{k=0}^{K_\rho-1} a_{ij}(k-K_\rho)A_{ij}^{\dagger}(0)\right]\right),\end{aligned}$$

which is the claim of the part (I) in Theorem 4. The rest of the proof is to establish inequalities (D.1) and (D.2).

Now we proceed to prove inequality (D.1).

$$\begin{aligned}&\mathbf{E}\big[2\textstyle\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\big)\big]\\ =&2\Big(\mathbf{E}\big[\sum_{i,j} q_{ij}(t)A_{ij}^{\dagger}(t)\big]-\mathbf{E}\big[\sum_{i,j} q_{ij}(t)D_{ij}^{\dagger}(t)\big]\Big).\end{aligned} \tag{D.3}$$

Focusing on $\mathbf{E}\big[\sum_{i,j} q_{ij}(t)A_{ij}^{\dagger}(t)\big]$, the first expectation term above, we have

$$\begin{aligned}
&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t)A_{ij}^{\dagger}(t)\Big]\\
=&\mathbf{E}\Big[\sum_{i,j}\big(q_{ij}(t-K_\rho)-\sum_{k=0}^{K_\rho-1} d_{ij}(t-K_\rho+k)+\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)\big)A_{ij}^{\dagger}(t)\Big] &\text{(D.4)}\\
\le&\mathbf{E}\Big[\sum_{i,j}\big(q_{ij}(t-K_\rho)+\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)\big)A_{ij}^{\dagger}(t)\Big] &\text{(D.5)}\\
=&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t-K_\rho)A_{ij}^{\dagger}(t)\Big]+\mathbf{E}\Big[\sum_{i,j}\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]\\
=&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t-K_\rho)A_{ij}^{\dagger}(t)\Big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]\\
=&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t-K_\rho)\mathbf{E}\big[A_{ij}^{\dagger}(t)\mid H(t-K_\rho)\big]\Big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]\\
\le&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t-K_\rho)\big(\Lambda_{ij}^{\dagger}(t)+\sum_{(l,w)\in Q_{ij}^{\dagger}}\zeta_{lw}(K_\rho)\big)\Big]\\
&+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big] &\text{(D.6)}\\
\le&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t-K_\rho)\xi(K_\rho)\Big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big] &\text{(D.7)}\\
=&\xi(K_\rho)\mathbf{E}\big[\|Q(t-K_\rho)\|_1\big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]. &\text{(D.8)}
\end{aligned}$$

In the above derivation, (D.4) is due to (2). Inequalities (D.5), (D.6), and (D.7) are due to $d_{ij}(t)\ge 0$ for any i,j and t, Fact 4 and definition (33) respectively.

Focusing on the second expectation term on the RHS of (D.3), we have

$$\begin{aligned}
&\mathbf{E}\big[\sum_{i,j} q_{ij}(t)D_{ij}^{\dagger}(t)\big]\\
=&\mathbf{E}\Big[\mathbf{E}\big[\sum_{i,j} q_{ij}(t)D_{ij}^{\dagger}(t)\mid Q(t)\big]\Big]\\
=&\mathbf{E}\Big[\sum_{i,j}\mathbf{E}\big[q_{ij}(t)D_{ij}^{\dagger}(t)\mid Q(t)\big]\Big]\\
\ge&\mathbf{E}\Big[\sum_{i,j} q_{ij}(t)\Big] &\text{(D.9)}\\
=&\mathbf{E}[\|Q(t)\|_1].
\end{aligned}$$

In the above derivation, inequality (D.9) is due to the weaker departure inequality (7).

Therefore, we have, in steady state

$$\begin{aligned}
&\mathbf{E}\big[2\textstyle\sum_{i,j} q_{ij}(t)\big(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\big)\big]\\
\le&2\Big(\xi(K_\rho)\mathbf{E}\big[\|Q(t-K_\rho)\|_1\big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]-\mathbf{E}[\|Q(t)\|_1]\Big)\\
=&2\Big(\xi(K_\rho)\mathbf{E}\big[\|\bar{Q}\|_1\big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(t-K_\rho+k)A_{ij}^{\dagger}(t)\Big]-\mathbf{E}[\|\bar{Q}\|_1]\Big) &\text{(D.10)}\\
=&2\Big(\big(\xi(K_\rho)-1\big)\mathbf{E}\big[\|\bar{Q}\|_1\big]+\sum_{i,j}\mathbf{E}\Big[\sum_{k=0}^{K_\rho-1} a_{ij}(k-K_\rho)A_{ij}^{\dagger}(0)\Big]\Big). &\text{(D.11)}
\end{aligned}$$

Note (D.10) is due to the fact $\mathbf{E}\big[\|Q(\tau)\|_1\big]=\mathbf{E}\big[\|\bar{Q}\|_1\big]$ for any τ in steady state.

Now we prove inequality (D.2). By simple calculations, we have

$$\begin{aligned}
&\mathbf{E}\left[\left(a_{ij}(t)-d_{ij}(t)\right)\left(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\right)\right]\\
=&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t)-d_{ij}(t)A_{ij}^{\dagger}(t)-a_{ij}(t)D_{ij}^{\dagger}(t)+d_{ij}(t)D_{ij}^{\dagger}(t)]\\
=&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t)]-\mathbf{E}[d_{ij}(t)A_{ij}^{\dagger}(t)]\\
&\quad-\mathbf{E}[a_{ij}(t)D_{ij}^{\dagger}(t)]+\mathbf{E}[d_{ij}(t)D_{ij}^{\dagger}(t)]\\
=&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t)]-\mathbf{E}[d_{ij}(t)A_{ij}^{\dagger}(t)]\\
&\quad-\mathbf{E}[a_{ij}(t)D_{ij}^{\dagger}(t)]+\mathbf{E}[d_{ij}(t)] \qquad \text{(D.12)}\\
\le&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t)]+\mathbf{E}[d_{ij}(t)]. \qquad \text{(D.13)}
\end{aligned}$$

In arriving at (D.12), we have used Fact 3, which is, for any i,j, $d_{ij}(t)D_{ij}^{\dagger}(t) = d_{ij}(t)$. Inequality (D.13) is due to $a_{ij}(t)\ge 0$, $d_{ij}(t)\ge 0$ for any i,j and t.

Since $\{Q(t),X(t)\}_{t=0}^{\infty}$ is positive recurrent, we have, in steady state, $\mathbf{E}[d_{ij}(t)]=\lambda_{ij}$, for any i,j. Hence, we have, in steady state,

$$\begin{aligned}
&\mathbf{E}\Big[\sum_{i,j}\left(a_{ij}(t)-d_{ij}(t)\right)\left(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\right)\Big]\\
=&\sum_{i,j}\mathbf{E}\Big[\left(a_{ij}(t)-d_{ij}(t)\right)\left(A_{ij}^{\dagger}(t)-D_{ij}^{\dagger}(t)\right)\Big]\\
\le&\sum_{i,j}\left(\mathbf{E}[a_{ij}(0)A_{ij}^{\dagger}(0)]+\lambda_{ij}\right).
\end{aligned}$$

Proof of Part (II). Since $\{a_{ij}(t)\}_{1\le i,j\le N}$ are now mutually independent, we have, for any i,j,

$$\begin{aligned}
&\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t)]\\
=&\mathbf{E}\left[a_{ij}(t)\left(A_{ij}^{\dagger}(t)-a_{ij}(t)\right)+a_{ij}^2(t)\right]\\
=&\mathbf{E}\left[a_{ij}(t)\left(A_{ij}^{\dagger}(t)-a_{ij}(t)\right)\right]+\mathbf{E}\left[a_{ij}^2(t)\right]\\
=&\mathbf{E}\left[a_{ij}(t)\right]\mathbf{E}\left[\sum_{(l,w)\in Q_{ij}^{\dagger}\setminus\{(i,j)\}}a_{lw}(t)\right]+\mathbf{E}\left[a_{ij}^2(t)\right]\\
=&\lambda_{ij}(\Lambda_{ij}^{\dagger}-\lambda_{ij})+\mathbf{E}\left[a_{ij}^2(t)\right]\\
=&\sigma_{ij}^2+\lambda_{ij}\Lambda_{ij}^{\dagger}, \qquad \text{(D.14)}
\end{aligned}$$

where $\sigma_{ij}^2 \triangleq \mathbf{E}\left[a_{ij}^2(t)\right]-\lambda_{ij}^2$ is the steady-state variance of $a_{ij}(t)$.

Similarly, we have, for any i,j, any t and any integer $k>0$

$$\mathbf{E}[a_{ij}(t)A_{ij}^{\dagger}(t+k)]=\lambda_{ij}\Lambda_{ij}^{\dagger}+\theta_{ij}(k). \qquad \text{(D.15)}$$

where $\theta_{ij}(k)\triangleq \mathbf{E}\left[a_{ij}(t)a_{ij}(t+k)\right]-\lambda_{ij}^2$ is the auto-correlation in $a_{ij}(t)$ in steady state.

Finally, using (D.14), (D.15), and inequality (35), we have

$$\mathbf{E}[\|\bar{Q}\|_1]\le\frac{1}{2(1-\xi(K_\rho))}\left(\sum_{i,j}\left(\sigma_{ij}^2+\lambda_{ij}\Lambda_{ij}^{\dagger}+\lambda_{ij}\right)+2\sum_{i,j}\sum_{k=1}^{K_\rho}\left(\lambda_{ij}\Lambda_{ij}^{\dagger}+\theta_{ij}(k)\right)\right).$$

References

[1] C. Cakir, R. Ho, J. Lexau, K. Mai, Modeling and design of high-radix on-chip crossbar switches, in: Proceedings of the ACM/IEEE NoCS, Vancouver, BC, Canada, 2015, pp. 20:1–20:8.
[2] C. Cakir, R. Ho, J. Lexau, K. Mai, Scalable high-radix modular crossbar switches, in: Proceedings of the HOTI, Santa Clara, CA, USA, 2016, pp. 37–44.
[3] J. Dai, B. Prabhakar, The throughput of data switches with and without speedup, in: Proceedings of the IEEE INFOCOM, Tel Aviv, Israel, 2000, pp. 556–564.
[4] M.J. Neely, Delay analysis for maximal scheduling in wireless networks with bursty traffic, in: Proceedings of the IEEE INFOCOM, 2008.
[5] A. Israel, A. Itai, A fast and simple randomized parallel algorithm for maximal matching, Inf. Process. Lett. 22 (2) (1986) 77–80.
[6] T.E. Anderson, S.S. Owicki, J.B. Saxe, C.P. Thacker, High-speed switch scheduling for local-area networks, ACM Trans. Comput. Syst. 11 (4) (1993) 319–352.

[7] N. McKeown, The iSLIP scheduling algorithm for input-queued switches, IEEE/ACM Trans. Netw. 7 (2) (1999) 188–201.
[8] L. Gong, P. Tune, L. Liu, S. Yang, J.J. Xu, Queue-proportional sampling: A better approach to crossbar scheduling for input-queued switches, Proc. ACM Meas. Anal. Comput. Syst. 1 (1) (2017) 3:1–3:33.
[9] P. Giaccone, B. Prabhakar, D. Shah, Randomized scheduling algorithms for high-aggregate bandwidth switches, IEEE J. Sel. Areas Commun. 21 (4) (2003) 546–559.
[10] M.J. Neely, Delay analysis for maximal scheduling with flow control in wireless networks with bursty traffic, IEEE/ACM Trans. Netw. 17 (4) (2009) 1146–1159.
[11] D.J. Aweya, Switch/Router Architectures: Shared-Bus and Shared-Memory Based Systems, first ed., Wiley-IEEE Press, 2018.
[12] Y. Tamir, G.L. Frazier, High-performance multi-queue buffers for VLSI communications switches, SIGARCH Comput. Archit. News 16 (2) (1988) 343–354.
[13] M. Karol, M. Hluchyj, S. Morgan, Input versus output queueing on a space-division packet switch, IEEE Trans. Commun. 35 (12) (1987) 1347–1356.
[14] L. Tassiulas, A. Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, IEEE Trans. Automat. Control 37 (12) (1992) 1936–1948.
[15] N. McKeown, A. Mekkittikul, V. Anantharam, J. Walrand, Achieving 100% throughput in an input-queued switch, IEEE Trans. Commun. 47 (8) (1999) 1260–1267.
[16] L. Tassiulas, A. Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, IEEE Trans. Automat. Control 37 (12) (1992) 1936–1948.
[17] D. Shah, M. Kopikare, Delay bounds for approximate maximum weight matching algorithms for input queued switches, in: Proceedings of the IEEE INFOCOM, Vol. 2, 2002, pp. 1024–1031.
[18] R. Duan, H.-H. Su, A scaling algorithm for maximum weight matching in bipartite graphs, in: Proceedings of the ACM-SIAM SODA, 2012, pp. 1413–1424.
[19] I. Keslassy, R. Zhang-Shen, N. McKeown, Maximum size matching is unstable for any packet switch, IEEE Commun. Lett. 7 (10) (2003) 496–498.
[20] S. Mou, S.T. Maguluri, Heavy traffic queue length behaviour in a switch under Markovian arrivals, 2020, arXiv:2006.06150.
[21] B. Hajek, Notes for ECE 467 communication network analysis, 2006, http://www.ifp.illinois.edu/~hajek/Papers/networkanalysisDec06.pdf.
[22] B. Hu, K.L. Yeung, Q. Zhou, C. He, On iterative scheduling for input-queued switches with a speedup of $2-1/N$, IEEE/ACM Trans. Netw. 24 (6) (2016) 3565–3577.
[23] B. Hu, F. Fan, K.L. Yeung, S. Jamin, Highest rank first: A new class of single-iteration scheduling algorithms for input-queued switches, IEEE Access 6 (2018) 11046–11062.
[24] J.M. Flegal, G.L. Jones, et al., Batch means and spectral variance estimators in Markov chain Monte Carlo, Ann. Statist. 38 (2) (2010) 1034–1070.
[25] P.W. Glynn, W. Whitt, et al., The asymptotic validity of sequential stopping rules for stochastic simulations, Ann. Appl. Probab. 2 (1) (1992) 180–198.
[26] Yihan Li, S. Panwar, H.J. Chao, On the performance of A dual round-robin switch, in: Proceedings of the IEEE INFOCOM, Vol. 3, Anchorage, AK, USA, 2001, pp. 1688–1697.
[27] N.W. McKeown, Scheduling Algorithms for Input-queued Cell Switches (Ph.D. thesis), Berkeley, CA, USA, 1995.
[28] A. Scicchitano, A. Bianco, P. Giaccone, E. Leonardi, E. Schiattarella, Distributed scheduling in input queued switches, in: Proceedings of the IEEE ICC, 2007, pp. 6330–6335.
[29] D. Lin, Y. Jiang, M. Hamdi, Selective-request round-robin scheduling for VOQ packet switch architecture, in: Proceedings of the IEEE ICC, 2011, pp. 1–5.
[30] L. Tassiulas, Linear complexity algorithms for maximum throughput in radio networks and input queued switches, in: Proceedings of the IEEE INFOCOM, San Francisco, CA, USA, 1998, pp. 533–539.
[31] L. Gong, L. Liu, S. Yang, J.J. Xu, Y. Xie, X. Wang, SERENADE: A parallel iterative algorithm for crossbar scheduling in input-queued switches, in: Proceedings of the IEEE HPSR, 2020, pp. 1–6.
[32] S. Ye, T. Shen, S. Panwar, An $O(1)$ scheduling algorithm for variable-size packet switching systems, in: Proceedings of the 48th Annual Allerton Conference, 2010, pp. 1683–1690.
[33] G. Fayolle, V.A. Malyshev, M.V. Menshikov, Topics in the Constructive Theory of Countable Markov Chains, Cambridge University Press, 1995.