Capacity Provisioning for Schedulers with Tiny Buffers

Yashar Ghiassi-Farrokhfal and Jörg Liebeherr
Department of Electrical and Computer Engineering
University of Toronto

***Abstract*—Capacity and buffer sizes are critical design parameters in schedulers which multiplex many flows. Previous studies show that in an asymptotic regime, when the number of traffic flows N goes to infinity, the choice of scheduling algorithm does not have a big impact on performance. We raise the question whether or not the choice of scheduling algorithm impacts the capacity and buffer sizing for moderate values of N (e.g., few hundred). For Markov-modulated On-Off sources and for finite N, we show that the choice of scheduling is influential on (1) buffer overflow probability, (2) capacity provisioning, and (3) the viability of network decomposition in a non-asymptotic regime. This conclusion is drawn based on numerical examples and by a comparison of the scaling properties of different scheduling algorithms. In particular, we show that the per-flow capacity converges to the per-flow long-term average rate of the arrivals with convergence speeds ranging from $O\left(\sqrt{\frac{\log N}{N}}\right)$ to $O\left(\frac{1}{N}\right)$ depending on the scheduling algorithm. This speed of convergences of the required capacities for different schedulers (to meet a target buffer overflow probability) is perceptible even for moderate values of N in our numerical examples.**

I. Introduction

Capacity and buffer provisioning in a scheduler which multiplexes multiple traffic flows with stringent service demands is known to be challenging. In this joint design problem, buffer size can be decreased at the cost of increasing the capacity. Recently, several arguments have been made in favour of small buffer sizes at packet switches: First, when the number of flows is large, adding small buffers usually satisfies the buffer overflow probability constraints [16]. Secondly, small buffers enable fast memory technologies such as SRAM or all-optical buffering [7]. Finally, small buffers may mitigate traffic burstiness. For instance, the polynomially decreasing overflow probability of self-similar traffic turns to exponentially decreasing buffer overflow in schedulers with small buffers [15], [16].

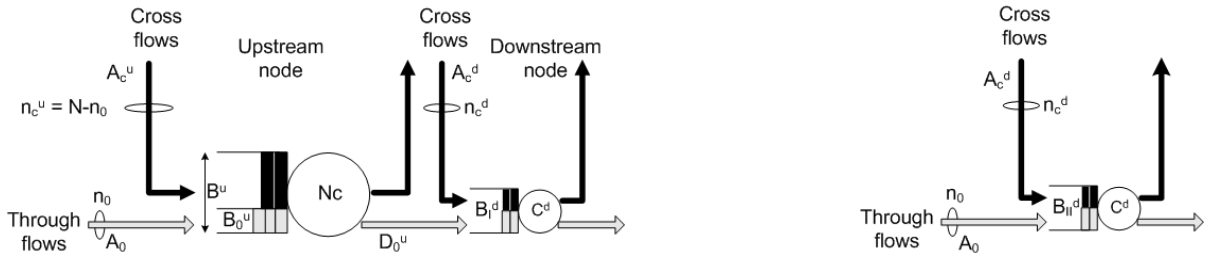
Capacity and buffer provisioning to meet a target loss probability, have been studied extensively in a many sources asymptotic when the number of i.i.d. flows N tends to infinity. Using large deviation theory, it has been shown that the steady state of the total backlog B exceeding a threshold b is asymptotically described by $P\{B > b\} \approx e^{-NI(\frac{b}{N})}$, where I is called the asymptotic rate function taking different forms depending on the input traffic and buffer sizes [1], [19]. For instance, for a given utilization and a large class of On-Off sources, asymptotic rate function for small buffers takes the form $I(\frac{b}{N}) \approx K_1 + K_2\sqrt{\frac{b}{N}}$ for some positive constants K_1 and K_2 [16]. Albeit a single flow FIFO queue is assumed for most of the asymptotic backlog analyses, it is shown that the asymptotic results can be extended to general work-conserving schedulers [8], [22]. This suggests that for sufficiently large N, capacity and buffer provisioning can be carried out regardless of the scheduling algorithm used in the switches.

The role of scheduling on the buffer/capacity provisioning for a target buffer overflow probability for finite values of N has not been fully investigated. For a general work-conserving scheduler and a class of Markov-modulated sources [6], it is shown in [5] that $O(1)$ buffers are sufficient to satisfy a target overflow probability. In a multi-flow FIFO scheduler and Markov-modulated On-Off sources, an $O(\frac{1}{N})$ per-flow capacity is shown to be sufficient to guarantee a probabilistic end-to-end delay bound [5].

Networks where buffers are small enough to limit traffic distortion, but sufficiently large to result in small loss probability justify a decomposition analysis, where each node in the network can be analyzed without regards to other nodes in the network. The viability of network decomposition is considered in asymptotic [8], [17], [20], [22] and numerically in non-asymptotic regimes [5], [6].

In this paper we aim to investigate whether or not scheduling information is a key factor for the network analysis for finite values of N. We use the non-asymptotic probabilistic backlog bound from [10] for the class of Δ-schedulers to compute the required capacity to satisfy a target overflow probability in a scheduler with a given small non-zero buffer size. We show that a per-flow buffer size can be as small as $O(\frac{1}{N})$, which justifies talking about a tiny buffer regime. We employ Network Calculus in our derivations [2], [3], [11]. In this paper, we study the impact of resource provisioning and performance analysis for finite N:

- We investigate how the per-flow backlog and output scale with the total number of flows for each choice of scheduling algorithm.
- We quantify the required capacity to satisfy a predefined overflow probability when the buffer size is arbitrarily small. We show that the scheduling algorithm determines the speed of the convergence of per-flow capacity to the long-term per-flow average rate.
- We study the viability of network decomposition in a non-asymptotic regime for different schedulers. Using numerical examples, we show that network decomposition might be valid even in a non-asymptotic regime for some



(a) Scenario I (A two-node network).

(b) Scenario II (Removing the upstream node from Scenario I).

Fig. 1: System model and network decomposition.

schedulers including FIFO.

The rest of the paper is organized as follows. In the next section, we introduce our system model. In Sec. III we study the impact of scheduling on the backlog bounds from [10] for a Δ-scheduler as the number of flows increases. Then, for a fixed and arbitrarily small buffer size, we formulate an optimization problem to derive the required capacity which satisfies a target overflow probability and explore the scaling of the per-flow capacity as a function of N. Sec. IV is devoted to network decomposition of a network of Δ-schedulers. We present numerical results in Sec. V and we conclude the paper in Sec. VI.

II. SYSTEM MODEL

Fig. 1 illustrates the models we will consider in this paper. Fig. 1a (Scenario I) demonstrates a two-node network scenario. The upstream node is fed by N input flows (n_0 i.i.d. through flows A_0 and n_c^u ($= N - n_0$) i.i.d. cross flows A_c^u) and has a total capacity Nc, where c is referred to as the per-flow capacity. After being served at the upstream node, the departures of the through flows from the upstream node D_0^u enter the downstream node with total capacity C^d. The downstream node multiplexes the through flows with n_c^d i.i.d. cross flows A_c^d. In Fig. 1b (Scenario II), the upstream node is removed, and the through flows A_0 enter the downstream node directly and are multiplexed by cross flows A_c^d. We do not assume independence between through and cross flows.

If the arrivals at a certain time exceed the capacity of the link, the exceeding arrivals are stored in a buffer to be served later. The buffer content at any time t is called the backlog B at that time. We assume that the buffer at each node is partitioned to separate through and cross flows arrivals. We denote by B_0^u (B^u) and $B_{\rm I}^d$ ($B_{\rm II}^d$), respectively, the backlog of the through flows (total flows) at the upstream node and the total backlog at the downstream node in Scenario I (Scenario II).

In a finite buffer queue, network resources are provisioned so that the losses occur rarely, i.e., by choosing large enough buffer/capacity. One of the main target design constraints is overflow probability defined as the likelihood that at a given time instant the input traffic finds the buffer full. As many other papers in the literature, we use the probability that the virtual backlog in an infinite buffer queue exceeds a threshold equal to the buffer size. Computing the latter probability is less challenging than the overflow probability and is an upper bound on the overflow probability and is shown to be very close to the buffer overflow especially if N is not small [9]. The backlog in an infinite buffer queue at any time $t \geq 0$ is the difference between the cumulative arrival A and departure processes D, i.e., $B(t) = A(t) - D(t)$.

We study the impact of scheduling on capacity provisioning in a single node in Sec. III, where the upstream node in Scenario I is used as the model for the single node scenario. We examine network decomposition (in Sec. IV) by investigating whether or not the backlog in the downstream node of Scenario I is affected by the upstream node or it has similar statistical properties to those in the single node depicted in Scenario II.

We use a continuous time fluid flow model, where $A(t)$ represents the total arrival from a process A in time interval $[0, t)$ and $A(s, t) = A(t) - A(s)$. For any arrival process A, we define the long-term average rate $\bar{a}$ as

$$\bar{a} = \lim_{t \to \infty} \frac{A(t)}{t} \ .$$

The per-flow long-term average rate of arrivals in the upstream link is defined as $\bar{a}^u = \frac{n_c^u \bar{a}_c^u + n_0 \bar{a}_0}{N}$, where $\bar{a}_0$ and $\bar{a}_c^u$ are, respectively, the long-term average rates of the through and cross flows at that node. The per-flow link capacity must satisfy $c \geq \bar{a}^u$ as stability condition. The utilization is defined as the ratio between the long-term average rate of the arrivals to the link capacity, e.g., the utilization of the upstream node is computed as

$$U^u = \frac{N\bar{a}^u}{Nc} = \frac{\bar{a}^u}{c} \ .$$

We use the stochastic network calculus for the analyses which employs envelopes to describe the probabilistic upper bounds at each time interval. A non-decreasing function G is a statistical envelope [4] for process A with bounding function ε for all s, t with $0 \leq s \leq t$, if

$$P\{A(s, t) > G(t - s; \sigma)\} \leq \varepsilon(\sigma) \ , \tag{1}$$

for any $\sigma \geq 0$, and $0 \leq \varepsilon(\sigma) \leq 1$. A statistical envelope can be inferred immediately from the definition of a large class of traffic sources known as the Exponential Bounded Burstiness (EBB) traffic arrivals [21]. An arrival process A

is an EBB traffic with parameters (M, ρ, α), represented by $A \sim (M, \rho, \alpha)$, if it satisfies that for any $\sigma \geq 0$ and any $s \leq t$

$$P\{A(s,t) > \rho(t-s) + \sigma\} \leq Me^{-\alpha\sigma} \ . \tag{2}$$

A statistical sample path envelope $\mathcal{G}$ is a stricter envelope than the statistical envelope and is a non-decreasing function that satisfies the following at any time t [10]:

$$P\Big\{\sup_{s\leq t}\{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0\Big\} \leq \varepsilon(\sigma) \ , \tag{3}$$

for any $\sigma \geq 0$, and $0 \leq \varepsilon(\sigma) \leq 1$. If $A \sim (M, \rho, \alpha)$ then, for any $\gamma > 0$ the following is a statistical envelope satisfying Eq. (3)

$$\mathcal{G}(t;\sigma) = (\rho+\gamma)t + \sigma; \quad \varepsilon(\sigma) = Me\left(1 + \frac{\rho}{\gamma}\right)e^{-\alpha\sigma} \ . \tag{4}$$

We assume that n_0 is fixed and does not scale with N. This is a common assumption in most of the papers on network decomposition [5], [6], [8], [22]. In this paper, we assume that each traffic flow is a Markov-modulated On-Off (MMOO) source which is a common model for voice traffic [18]. An MMOO flow can be modelled by a two-state Markov chain with states On and Off. In the On state, traffic is generated with rate P, and no traffic is generated in the Off state. The sojourn time between On to Off and Off to On transitions are exponentially distributed with rates λ and μ, respectively. The average cycle time to return to the same state is $T^* = \frac{\lambda+\mu}{\lambda\mu}$. If A is an MMOO process with parameters, λ, μ, and P then, for any $\alpha \geq 0$, $t \geq 0$:

$$\sup_{s\geq 0} E\Big[e^{\alpha A(s,t+s)}\Big] \leq e^{\alpha r(\alpha)t} \ , \tag{5}$$

with

$$r(\alpha) = \frac{1}{2\alpha}(P\alpha - \lambda - \mu + \sqrt{(P\alpha - \mu + \lambda)^2 + 4\mu\lambda}) \ . \tag{6}$$

Note that $r(\alpha)$ satisfying Eq. (5) is a special case of effective bandwidth [12] for process A. $r(\alpha)$ is non-decreasing in α and satisfies

$$\forall \alpha \geq 0: \qquad r(0) = \bar{a} \leq r(\alpha) \leq r(\infty) = P \ , \tag{7}$$

where $\bar{a} = \frac{P\mu}{\lambda+\mu}$ is the average rate of A. Traffic sources which satisfy Eq. (5) and in particular, MMOO sources are also used in the papers studying network decomposition in a non-asymptotic regime [5], [6].

Suppose that A is the aggregate of n independent flows each satisfying Eq. (5) with parameter r. Then applying the Chernoff bound shows that A is an EBB arrival in terms of Eq. (2) with $(1, nr(\alpha), \alpha)$. For simplicity of notation, we will use $\rho(\alpha) = nr(\alpha)$ and we drop α frequently in the paper.

We assume that the scheduling algorithm in the upstream node is a Δ-scheduler [14]. For any ordered pair of input flows (i, j), there exists a constant $\Delta_{i,j}$ that determines the precedence between the arrivals from flow i and j. More specifically, if an arrival flow i arrives to a Δ-scheduler at time t, the arrivals from flow j have higher precedence if and only if they arrive at or before $t + \Delta_{i,j}$ (Fig. 2). By the above definition, first in first out (FIFO), static priority (SP), and earliest deadline first (EDF) are examples of Δ-schedulers with

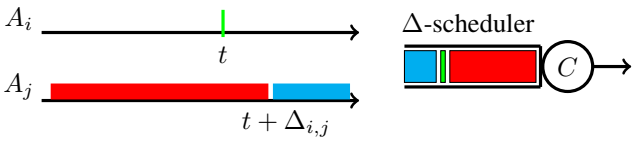


Fig. 2: Δ-scheduler algorithm.

- FIFO: $\Delta_{i,j} = 0$ for any pair of flows i, j
- SP: $\begin{cases} \Delta_{i,j} = +\infty & \text{if } j \text{ has higher priority than } i \\ \Delta_{i,j} = -\infty & \text{if } i \text{ has higher priority than } j \end{cases}$
- EDF: $\Delta_{i,j} = d_i^* - d_j^*$ for any $i, j \in \mathcal{N}$

where d_j^* for any $j \in \mathcal{N}$ in EDF is the a priori delay bound for flow j. We note that if the scheduling information is not available, performance bounds must be obtained by considering the scheduling algorithm which leads to the worst-case bounds among all work-conserving schedulers. This benchmark is referred to as blind multiplexing (BMux) and is equivalent to a two-class SP scheduler which gives the lower priority to that specific flow.

III. Scaling Properties of Backlog and Capacity

In this section we study the impact of the choice of scheduler on the backlog and capacity dimensioning at the upstream node depicted in Fig. 1a. Since we do not consider the downstream node in this section, we sometimes drop the superscript u to simplify notation. We use the following theorem from [10] which computes a per-flow backlog bound for EBB traffic sources in a Δ-scheduler.

Theorem 1 (Backlog bound for Δ-schedulers [10]): Suppose that EBB through flows with aggregate parameters (M_0, ρ_0, α_0) satisfying Eq. (2) are multiplexed with EBB cross flows with aggregate parameters (M_c, ρ_c, α_c) in a Δ-scheduler with parameter $\Delta_{0,c} = \Delta$ and capacity C. Define a vector $\underline{\sigma} = (\sigma_0, \sigma_c)$ with arbitrary positive elements. For any $0 \leq \gamma \leq \frac{C-\rho_c-\rho_0}{2}$, define the following parameters

$$\theta^* = \min\left(\frac{\sigma_c}{C - \rho_c - \gamma}, \frac{[\sigma_c + (\rho_c+\gamma)\Delta]_+}{C}\right) \ , \tag{8}$$

$$b(\underline{\sigma}) = \sigma_0 + (\rho_0 + \gamma)\theta^* \ , \tag{9}$$

$$\varepsilon(\underline{\sigma}) = M_0 e\Big(1 + \frac{\rho_0}{\gamma}\Big)e^{-\alpha_0\sigma_0} + M_c e\Big(1 + \frac{\rho_c}{\gamma}\Big)e^{-\alpha_c\sigma_c} \ . \tag{10}$$

Then, the backlog of through flows B_0 at any time $t \geq 0$ satisfies

$$P\{B_0(t) > b(\underline{\sigma})\} \leq \varepsilon(\underline{\sigma}) \ . \tag{11}$$

The above theorem can be used for our system model in which through and cross flows are EBB processes, respectively, with $A_0 \sim (1, n_0 r_0(\alpha_0), \alpha_0)$ and $A_c^u \sim (1, n_c r_c(\alpha_c), \alpha_c)$ for any non-negative α_0 and α_c. To capture the scaling of the backlog behaviour of the through flows, we

use the deterministic peak-rate envelope n_0P_0 for the through flows as a special case of the statistical sample path envelopes. To be more precise, we replace $\mathcal{G}_0(t) = (\rho_0 + \gamma)t + \sigma$ and $\varepsilon_0(\sigma) = M_0 e(1 + \frac{\rho_0}{\gamma})e^{-\alpha_0\sigma}$ in Eq. (4), respectively, with $\mathcal{G}_0(t) = n_0P_0$ and $\varepsilon_0 = 0$. This eliminates the first term in Eqs. (9)-(10) which correspond to the through flows statistical envelope bounding function and replaces $\rho_0+\gamma_0$ with n_0P_0 in Eq. (9). The resulting backlog bound exhibits an exponential decay rate in N for any $N \geq \frac{n_0(P_0-\bar{a}_c)}{c-\bar{a}_c}$ (as shown below).

Corollary 1: Suppose that any through and cross flow source in the upstream node in Fig. 1a satisfies Eq. (5), each through flow has a peak rate P_0, and $N \geq \frac{n_0(P_0-\bar{a}_c)}{c-\bar{a}_c}$. Then, there exist constants $\alpha_b > 0$ and K such that for any $\sigma \geq 0$

$$P\{B_0^u(t) > \sigma\} \leq Ke^{-N\alpha_b\sigma} \ , \tag{12}$$

where $K = O(N)$ if $\Delta \geq 0$ and $K = O(Ne^{-N\beta})$ for some constant β if $\Delta < 0$.

Proof: We consider B_0^u, separately, for $\Delta \geq 0$ and $\Delta < 0$. In both cases, the parameters are chosen such that $b(\underline{\sigma})$ from Eq. (9) (with the above replacements) is upper bounded by a fixed σ, where ε in Eq. (10) is shown to decay by $e^{-N\alpha_b\sigma}$.

• $\Delta \geq 0$**:** With the condition on N in the corollary statement, we can always choose α_c small enough so that $c \geq \frac{n_0(P_0-r_c(\alpha_c))}{N} + r_c(\alpha_c)$. For any fixed $\sigma \geq 0$, setting the parameters in Eq. (9) as follows, guarantees that $b \leq \sigma$

$$\alpha_b = \frac{\alpha_c(Nc - \rho_c(\alpha_c) - \gamma_c)}{Nn_0P_0}; \qquad \sigma_c = \frac{N\alpha_b\sigma}{\alpha_c} \ . \tag{13}$$

Replacing the above parameters in Eq. (11), yields

$$P\{B_0^u(t) > \sigma\} \leq e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-N\alpha_b\sigma} \ . \tag{14}$$

Since $\alpha_b = O(1)$ (from Eq. (13)) and $\rho_c = (N - n_0)r_c$, Eq. (12) holds with $K = e\Big(1 + \frac{\rho_c}{\gamma_c}\Big) = O(N)$.

• $\Delta < 0$**:** In this case, θ^* from Eq. (8) is always evaluated to the second term. Choose α_c small enough so that $c \geq \frac{n_0(P_0-r_c(\alpha_c))}{N} + r_c(\alpha_c)$. For any fixed $\sigma \geq 0$, selecting parameters in Eq. (9) as follows guarantees that $b < \sigma$

$$\alpha_b = \frac{\alpha_c c}{n_0P_0}; \qquad \sigma_c = \frac{N\alpha_b\sigma}{\alpha_c} - (\rho_c + \gamma_c)\Delta \ . \tag{15}$$

By replacing these choices of parameters in Eqs. (8)-(11), we have

$$P\{B_0^u(t) > \sigma\} \leq e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{\alpha_c(\rho_c+\gamma_c)\Delta}e^{-N\alpha_b\sigma} \tag{16}$$

which implies that Eq. (12) holds with $K = O(Ne^{-N\beta})$ and $\beta = -\frac{N-n_0}{N}\alpha_c\bar{a}_c\Delta = \Theta(1)$. ∎

Corollary 1 shows that the decay rate of the backlog of the through flows varies with the choice of scheduling algorithms. It also indicates that for a predefined overflow probability, the buffer size can be as small as $O(\frac{1}{N})$ suggesting that a tiny buffer is enough for large values of N.

A. Capacity provisioning for Δ-schedulers

Suppose that the through flows in the upstream node in Fig. 1a have a target overflow probability ε^*. Using Theorem 1, we can quantify the required capacity for a given buffer size B. The per-flow capacity must satisfy the stability condition, i.e., $Nc \geq \rho_0+\rho_c+2\gamma$. Hence, we can set $c = \frac{1}{N}(\rho_0+\rho_c+2\gamma+X)$ for some positive slack variable $X \geq 0$. Equating b in Eq. (9) to B and solving the result for c, gives us another constraint on c with free parameters $\alpha_0, \alpha_c, \gamma$. Combining both conditions on c and enforcing the target overflow probability constraint, we can write

$$c = \frac{1}{N}\inf_{\alpha_0,\alpha_c,\gamma,X}\{\rho_0 + \rho_c + 2\gamma + X\} \tag{17}$$

$$s.t. \ \ X \geq 0 \tag{18}$$

$$\varepsilon^* \geq e\left(1 + \frac{\rho_0}{\gamma}\right)e^{-\alpha_0\sigma_0} + e\left(1 + \frac{\rho_c}{\gamma}\right)e^{-\alpha_c\sigma_c} \tag{19}$$

$$X + \rho_0 + \rho_c + 2\gamma \geq \frac{\rho_0(\alpha_0) + \gamma}{B - \sigma_0} \times \tag{20}$$

$$\min\Big(\sigma_c + \frac{(B - \sigma_0)(\rho_c(\alpha_c) + \gamma)}{\rho_0(\alpha_0) + \gamma}, [\sigma_c + (\rho_c(\alpha_c) + \gamma)\Delta]_+\Big) \ .$$

The above optimization problem can be used as follows to find how the required capacity to meet a target buffer overflow probability scales with N.

Corollary 2 (Per-flow capacity scaling properties): The per-flow capacity from Eq. (17) for MMOO input flows satisfies

$$c - \bar{a}^u = J(N) \ , \tag{21}$$

where $J(N) = O\left(\sqrt{\frac{\log N}{N}}\right)$ if $\Delta \geq 0$ and $J(N) = O\left(\frac{\log N}{N}\right)$ if $\Delta < 0$.

Corollary 2 shows that if N is sufficiently large, for any arbitrarily small buffer size, a target overflow probability can be satisfied even when the utilization is large.

Proof: Replace the left-hand side of Eq. (20) with Nc to get

$$c \geq \frac{1}{N}\frac{\rho_0(\alpha_0) + \gamma}{B - \sigma_0} \times \tag{22}$$

$$\min\Big(\sigma_c + \frac{(B - \sigma_0)(\rho_c(\alpha_c) + \gamma)}{\rho_0(\alpha_0) + \gamma}, [\sigma_c + (\rho_c(\alpha_c) + \gamma)\Delta]_+\Big) \ .$$

In addition, turn the inequality in Eq. (19) into an equality and split ε^* equally between the terms of the right-hand side of that equation to get

$$\sigma_0 = \frac{1}{\alpha_0}\log\left(\frac{2e(1 + \frac{\rho_0}{\gamma})}{\varepsilon^*}\right); \ \sigma_c = \frac{1}{\alpha_c}\log\left(\frac{2e(1 + \frac{\rho_c}{\gamma})}{\varepsilon^*}\right) \ . \tag{23}$$

Using the Taylor expansion of the rate r for an MMOO process from Eq. (6) for $\alpha \ll 1$ and since $r_c(0) = \bar{a}_c$, we have $r_c(\alpha_c) = \bar{a}_c + O(\alpha_c)$, or equivalently

$$\rho_c(\alpha_c) = (N - n_0)(\bar{a}_c + O(\alpha_c)) \ . \tag{24}$$

Combining the above equation with Eq. (17), we get

$$c - \bar{a}^u = O\Big(\frac{\alpha_c(N-n_0)}{N}\Big) + \frac{n_0(r_0(\alpha_0) - \bar{a}_0) + 2\gamma + X}{N} \ , \tag{25}$$

where the last term is $O(\frac{1}{N})$ noting that $X \geq 0$ can be chosen to be a constant with respect to N and $\lim_{N\to\infty} \frac{n_0}{N} = 0$. If $\alpha_c = \Theta\left(\sqrt{\frac{\log N}{N}}\right)$, we find from Eqs. (23)-(24), that $\sigma_c, \rho_c = O(\sqrt{N \log N})$. Thus, there is a choice of α_c which can satisfy Eq. (22), and for such an α_c, we have $c - \bar{a}^u = O\left(\sqrt{\frac{\log N}{N}}\right)$ from Eq. (25) .

If $\Delta < 0$ then, choose $\sigma_c = -(N - n_0)\bar{a}_c\Delta$ and find α_c from Eq. (23) which is $\alpha_c = O(\frac{\log N}{N})$. This choice of σ_c relaxes the constraint in Eq. (22) by setting the second term in the minimum to zero. Combining all of the above results and from Eq. (25), we have $c - \bar{a}^u = O\left(\frac{\log N}{N}\right)$. ■

A tighter bound on the capacity than that in Corollary 2 for $\Delta < 0$ can be obtained asymptotically, when N is sufficiently large. Set α_c to a constant (w.r.t. N) to have $\sigma_c = O(\log N)$ from Eq. (23). The second term in the minimum of Eq. (22) is zero for large enough N and from Eq. (25) we have $c - \bar{a}^u = O(\frac{1}{N})$.

IV. Non-asymptotic network decomposition for Δ-schedulers

In the previous section, we studied the role of scheduling algorithms in capacity provisioning for a target buffer overflow probability. In this section, we investigate how the viability of network decomposition can be affected by the choice of scheduling algorithms. Network decomposition (if valid) simplifies network analysis by eliminating the other nodes and analyzing each node in isolation (Fig. 1). The viability of network decomposition can be justified by showing that the random processes governing a multi-node network converge to those in a single node network when the number of flows N is large. In the literature, two types of convergence are considered:

1- Convergence of D_0^u to A_0: The upstream node can be disregarded in the backlog analysis of the downstream node if the statistical properties of D_0^u and A_0 are similar. Wischik [20] provides conditions under which the output traffic D_0^u satisfies the same moment generating function as the input A_0. In a FIFO link which is fed by two classes of arrivals, through and cross flows, each consisting of independent, identical leaky-bucket arrivals, the random burstiness of the output of the through flows converges to the input deterministic leaky-bucket burstiness of the aggregate through flows [22]. Finally, for MMOO traffic flows, the EBB characteristics of the output process in an isolated FIFO queue is shown to converge to those of the input process exponentially fast in N [5], [6].

2- Convergence of $B_{\mathbf{I}}^d$ to $B_{\mathbf{II}}^d$: The convergence of the backlog process B_{I}^d in the two node scenario in Fig. 1a to the backlog process B_{II}^d in the single node scenario in Fig. 1b can be used to show that upstream nodes have only a negligible impact. This provides an even stronger argument for decomposition than the convergence of D_0^u to A_0. Almost sure convergence is proved for regulated traffic and in probability for non-regulated traffic in [8]. In a non-asymptotic regime, the convergence of B_{I}^d to B_{II}^d is shown numerically even for moderate values of N in [5], [6]. Finally, it is shown that the loss ratio at the downstream in Scenario I converges to that of Scenario II when $N \to \infty$ [17].

We will consider both convergence criteria in our analysis of Δ-schedulers.

A. Output characterization

We use backlog characterization from Corollary 1 to formulate an envelope for the departures D_0^u from the downstream node in Scenario I in Fig. 1a.

Theorem 2 (Output EBB characterization): Consider the upstream node in Fig. 1a. With the assumptions in Corollary 1, the aggregate departures of through flows D_0^u is an EBB process with $D_0^u \sim (M_0^{out}, \rho_0, \alpha_0^{out})$, where $\alpha_0^{out} = \left(\frac{1}{\alpha_0} + \frac{1}{N\alpha_b}\right)^{-1}$ and

$$M_0^{out} = \left[M_0\Big(1 + \frac{\alpha_0}{N\alpha_b}\Big)\right]^{\frac{N\alpha_b}{\alpha_0 + N\alpha_b}} \left[K\left(1 + \frac{N\alpha_b}{\alpha_0}\right)\right]^{\frac{\alpha_0}{\alpha_0 + N\alpha_b}} . \tag{26}$$

Proof: The departures of through flows from the upstream node D_0^u in a time interval $[s,t)$ satisfies

$$D_0^u(s,t) \leq A_0^u(s,t) + B_0^u(s) \ . \tag{27}$$

Using Eq. (27), we can write

$$\begin{aligned} &P\{D_0^u(s,t) > \rho_0(t-s) + \sigma\} \\ &\quad \leq P\{A_0^u(s,t) + B_0^u(s) > \rho_0(t-s) + \sigma\} && (28) \\ &\quad \leq \inf_{\sigma_0+\sigma_b=\sigma} \Big\{ P\{A_0^u(s,t) > \rho_0(t-s) + \sigma_0\} \\ &\qquad\qquad + P\{B_0^u(s) > \sigma_b\} \Big\} && (29) \\ &\quad \leq \inf_{\sigma_0+\sigma_b=\sigma} \{M_0 e^{-\alpha_0\sigma_0} + K e^{-N\alpha_b\sigma_b}\} && (30) \\ &\quad = M_0^{out} e^{-\alpha_0^{out}\sigma} \ , && (31) \end{aligned}$$

where Eq. (27) is used in the second line. The event in Eq. (28) is a subset of the union of the events in Eq. (29). Combining this with the union bound yields Eq. (29). The next line applies the EBB characterizations of A_0^u and the backlog probabilistic bound from Corollary 1, and the last line uses Lemma 3 in [4].

■

Replacing K from Corollary 1 to the above theorem shows that if $N \to \infty$, the EBB parameters of the aggregate through flows is unchanged as it passes through the node, i.e., $\lim_{N\to\infty} D_0^u \sim (M_0, \rho_0, \alpha_0)$. This result relaxes the assumption on the independence between the through and cross flows in [5], [6] and the statistical independence of the through flows in any time window with the backlog status at the start of that time window in [6].

B. Non-asymptotic downstream node buffer overflow

As shown in [8], to guarantee the convergence of $B_{\rm I}^d$ to $B_{\rm II}^d$ as N increases, it is sufficient to show that a sample path backlog bound for B_0^u decays to zero. To construct a sample path for B_0^u, we use the following lemma:

Lemma 1: Suppose that the upstream through flows in Fig. 1a has a peak-rate n_0P_0 and there is a non-decreasing function ε_b such that at any time t and for any $\sigma > 0$,

$$P\{B_0^u(t) > \sigma\} \leq \varepsilon_b(\sigma) \ , \tag{32}$$

then, for any τ_s and $T \geq 0$,

$$P\Big\{ \sup_{0\leq t\leq T}\{B_0^u(t)\} > \sigma\Big\} \leq \Big\lceil \frac{T}{\tau_s}\Big\rceil \varepsilon_b(\sigma - n_0P_0\tau_s) \ . \tag{33}$$

Proof: We discretize time by letting τ_s be a time unit. The difference between the backlog of through flows at a given time instant t and at the most recent discrete time slot cannot be larger than the total arrivals of the through flows (at the upstream node) in an interval of size τ_s. Thus, by defining $T_Z = \big\{0, \tau_s, 2\tau_s, \ldots, \big(\lceil \frac{T}{\tau_s}\rceil - 1\big)\tau_s\big\}$, we have

$$\begin{aligned}\sup_{0\leq t\leq T}\{B_0^u(t)\} &\leq \max_{s\in T_Z}\Big(B_0^u(s) + A_0(s, s+\tau_s)\Big) \\ &\leq \max_{s\in T_Z}\big(B_0^u(s)\big) + n_0P_0\tau_s \ . \end{aligned} \tag{34}$$

With this result, we can write

$$\begin{aligned} P\Big\{ \sup_{0\leq t\leq T}\{B_0^u(t)\} > \sigma\Big\} &\leq P\Big\{\max_{s\in T_Z}\big(B_0^u(s)\big) > \sigma - n_0P_0\tau_s\Big\} \\ &\leq \sum_{s\in T_Z} P\Big\{B_0^u(s) > \sigma - n_0P_0\tau_s\} \\ &= \Big\lceil \frac{T}{\tau_s}\Big\rceil \varepsilon_b(\sigma - n_0P_0\tau_s) \ , \end{aligned}$$

where the second line uses Boole's inequality and the last line uses the fact that T_Z has $\lceil \frac{T}{\tau_s}\rceil$ elements. ∎

If T in Eq. (33) is a busy period bound, Lemma 1 provides a sample path backlog bound. A busy period of a work-conserving link refers to a time duration in which the backlog is non-zero. A busy period bound is formulated in [13] as follows. Suppose that process A is the arrival traffic to a link with capacity C and D is the corresponding departure process. Fix time t and define $\hat{x}_t$ to be the start of the busy period containing t. That is

$$\hat{x}_t = \sup\{s \leq t \mid A(s) \leq D(s)\} \ . \tag{35}$$

If $t - \hat{x}_t \leq T$ for any t then, T is a deterministic busy period bound which reduces the Reich's backlog equation to

$$B(t) = \sup_{0\leq s\leq T}\{A(t-s, t) - Cs\} \ . \tag{36}$$

If $\mathcal{G}$ is a statistical sample path envelope for A with bounding function ε then, a probabilistic busy period bound is given by

$$T(\sigma) = \inf\{s \mid \mathcal{G}(s;\sigma) \leq Cs\} \tag{37}$$

in the sense that for any $t \geq 0$

$$P\{t - \hat{x}_t > T(\sigma)\} \leq \varepsilon(\sigma) \ . \tag{38}$$

We use the above method to compute probabilistic busy periods on the downstream node for both scenarios in Fig. 1.

We first study Scenario II. Assume that through and cross flows at the downstream node are EBB with parameters $(1, \rho_0^{\rm II}, \alpha_0^{\rm II})$ and $(1, \rho_c^{\rm II}, \alpha_c^{\rm II})$. Here, we have simplified notation by setting $\rho_0^{\rm II} = n_0r_0(\alpha_0^{\rm II})$ and $\rho_c^{\rm II} = n_cr_c(\alpha_c^{\rm II})$. Inserting the EBB sample path envelopes from Eq. (4) in Eq. (37), a probabilistic busy period bound can be obtained. If $\alpha_c^{\rm II}$, $\alpha_0^{\rm II}$, and $\gamma^{\rm II}$ are chosen such that $C^d > \rho_0^{\rm II} + \rho_c^{\rm II}$ and $\gamma^{\rm II} \leq \frac{C^d - \rho_c^{\rm II} - \rho_0^{\rm II}}{2}$ then, for any $\sigma \geq 0$

$$T_{\rm II}(\sigma) = \frac{\sigma}{C^d - \rho_c^{\rm II} - \rho_0^{\rm II} - 2\gamma^{\rm II}} \tag{39}$$

is a probabilistic busy period bound for the node in Scenario II in the sense of Eq. (38) with bounding function

$$\varepsilon^{T_{\rm II}}(\sigma) = e\Big(2 + \frac{\rho_0^{\rm II}}{\gamma^{\rm II}} + \frac{\rho_c^{\rm II}}{\gamma^{\rm II}}\Big)e^{-\alpha_d^{\rm II}\sigma} \ , \tag{40}$$

where $\alpha_d^{\rm II} = (\frac{1}{\alpha_c^{\rm II}} + \frac{1}{\alpha_0^{\rm II}})^{-1}$.

To compute a busy period bound at the downstream node in Scenario I, we apply the upstream departure characterization from Theorem 2. Multiplexing the resulting departure process with EBB cross flows at the downstream node with parameters $(1, n_c^d r_c^{\rm I}(\alpha_c^{\rm I}), \alpha_c^{\rm I})$, a busy period bound for Scenario I at the upstream network can be obtained similar to that of $T_{\rm II}(\sigma)$. Then, for any $\gamma^{\rm I} \leq \frac{C^d - \rho_0^{\rm I} - \rho_c^{\rm I}}{2}$,

$$T_{\rm I}(\sigma) = \frac{\sigma}{C^d - \rho_0^{\rm I} - \rho_c^{\rm I} - 2\gamma^{\rm I}} \tag{41}$$

is a busy period bound in the sense of Eq. (38) with bounding function

$$\varepsilon^{T_{\rm I}}(\sigma) = e\Big(1 + M_0^{out} + M_0^{out}\frac{\rho_0^{\rm I}}{\gamma^{\rm I}} + \frac{\rho_c^{\rm I}}{\gamma^{\rm I}}\Big)e^{-\alpha_d^{\rm I}\sigma} \ , \tag{42}$$

where $\alpha_d^{\rm I} = (\frac{1}{\alpha_c^{\rm I}} + \frac{1}{\alpha_0^{out}})^{-1}$. Since $M_0^{out} = O(1)$, we have $\varepsilon^{T_{\rm I}}(\sigma) = O(e^{-\alpha_d^{\rm I}\sigma})$ and $\varepsilon^{T_{\rm II}}(\sigma) = O(e^{-\alpha_d^{\rm II}\sigma})$. The above bounds are used in the next section to compare $B_{\rm I}^d$ with $B_{\rm II}^d$.

C. Almost sure network decomposition

We studied the network decomposition in the sense of the convergence of D_0^u to A_0 in Theorem 2. In this section, we study the network decomposition in the sense of the convergence of $B_{\rm I}^d$ to $B_{\rm II}^d$. The following theorem shows that $B_{\rm I}^d$ converges to $B_{\rm II}^d$ almost surely in the number of flows N for all Δ-schedulers. The rate of convergence is faster for negative values of Δ (compare $L(\sigma)$ for $\Delta \geq 0$ and $\Delta < 0$ in Theorem 3). In the specific case of FIFO ($\Delta = 0$) and for traffic sources with bounded peak-rate satisfying Eq. (5), the following theorem strengthens the results of [5], [8] from convergence in probability to almost sure convergence.

Theorem 3 (a.s. convergence of B_I^d to B_{II}^d): Consider the scenarios depicted in Fig. 1 and keep the assumptions in Corollary 1. Then, there exists a constant $\alpha > 0$ and a non-negative function L such that for any $\sigma \geq 0$

$$P\{|B_{\rm I}^d(t) - B_{\rm II}^d(t)| > \sigma\} \leq L(\sigma)e^{-N\alpha\sigma} \ , \tag{43}$$

where $L(\sigma) = O(N^2)$ if $\Delta \geq 0$ and $L(\sigma) = O(N^2 e^{-N\beta})$ for some constant β if $\Delta < 0$.

Proof: Denote by A_c^d the cross flows at the downstream node. Suppose that $T_{\rm I}$ and $T_{\rm II}$ are, respectively, the probabilistic busy period bounds computed from Eqs. (41) and (39) and define $T^{max} = \max\{T_{\rm I}, T_{\rm II}\}$. Then, for any $\sigma, \tau_s \geq 0$:

$$
\begin{aligned}
&P\{|B_{\rm I}^d(t) - B_{\rm II}^d(t)| > \sigma\} \\
&\leq P\Big\{\big| \sup_{0\leq u\leq T^{max}} \{A_0(t-u,t) + A_c^d(t-u,t) - C^d u\} \\
&\quad - \sup_{0\leq u\leq T^{max}} \{D_0^u(t-u,t) + A_c^d(t-u,t) - C^d u\}\big| > \sigma\Big\} \\
&\qquad + P\{t - \hat{x}_t^{\rm I} > T^{max}\} + P\{t - \hat{x}_t^{\rm II} > T^{max}\} \quad (44) \\
&\leq P\Big\{\sup_{0\leq u\leq T^{max}} \{A_0(t-u,t) - D_0^u(t-u,t)\} > \sigma\Big\} \\
&\qquad + P\{t - \hat{x}_t^{\rm I} > T^{max}\} + P\{t - \hat{x}_t^{\rm II} > T^{max}\} \quad (45) \\
&\leq P\Big\{\sup_{0\leq u\leq T^{max}} \{B_0^u(t-u) - B_0^u(t)\} > \sigma\Big\} \quad (46) \\
&\qquad + P\{t - \hat{x}_t^{\rm I} > T_{\rm I}(N\sigma)\} + P\{t - \hat{x}_t^{\rm II} > T_{\rm II}(N\sigma)\} \\
&\leq P\Big\{\sup_{0\leq u\leq T^{max}} \{B_0^u(t-u)\} > \sigma\Big\} \quad (47) \\
&\qquad + P\{t - \hat{x}_t^{\rm I} > T_{\rm I}(N\sigma)\} + P\{t - \hat{x}_t^{\rm II} > T_{\rm II}(N\sigma)\} \\
&\leq \left\lceil \frac{T^{max}}{\tau_s} \right\rceil \varepsilon_b(\sigma - n_0 P_0 \tau_s) + \varepsilon^{T_{\rm I}}(N\sigma) + \varepsilon^{T_{\rm II}}(N\sigma) \ , \quad (48)
\end{aligned}
$$

where $\hat{x}_t^{\rm I}$ and $\hat{x}_t^{\rm II}$ are the starts of the busy periods containing t at the downstream node, respectively, in Scenarios I and II. For the first inequality, we use Eq. (36) and the fact that $P(X) \leq P(X|Y) + P(Y')$ for any events X and Y, and $P(Y') = 1 - P(Y)$. Eq. (45) uses the fact that $\sup|X| - \sup|Y| \leq \sup|X - Y|$ for any X and Y. The last line is an application of Lemma 1 and the busy period bounds. ε_b is formulated in Eq. (14) for $\Delta \geq 0$, and in Eq. (16) for $\Delta < 0$. Moreover, $\varepsilon^{T_{\rm I}}$ and $\varepsilon^{T_{\rm II}}$ are formulated, respectively, in Eqs. (42) and (40).

Proper choices of the parameters in Eq. (48) completes the proof as follows. The last two terms are $O(e^{-N\alpha_{min}\sigma})$, where $\alpha_{min} = \min(\alpha_{\rm I}, \alpha_{\rm II})$. Set $\tau_s = \frac{T^{max}}{N^{1+\upsilon}}$ for some $\upsilon > 0$ to ensure that $n_0 P_0 \tau_s$ decays to zero as N increases. Inserting these choices in Eq. (48) shows that for any $\upsilon > 0$ Eq. (43) holds with $L(\sigma) = O(N^{2+\upsilon})$ if $\Delta \geq 0$ and $L(\sigma) = O(N^{2+\upsilon} e^{-N\beta})$ for some constant β if $\Delta < 0$. The theorem is then proved by noting that υ can be arbitrarily close to zero. ■

The above results imply that in addition to statistical multiplexing, the choice of scheduling algorithm is an important factor affecting the viability of network decomposition. Although a network can be decomposed in a many sources asymptotic regime for any work-conserving scheduler as also shown in [8] and [22], the viability of decomposition for moderate values of N is a matter of the scheduling algorithm.

V. Numerical Examples

In this section we examine our analytical results in numerical examples for scenarios in Fig. 1. Each through and cross flow is an MMOO source with parameters $\lambda = 1ms^{-1}$, $\mu = 0.11ms^{-1}$ which has an average rate of $\bar{a} = 0.15$ Mbps, and an average cycle time of $T^* = 10$ ms. The violation probabilities of the bounds in all examples are set to $\varepsilon^* = 10^{-6}$. We do not evaluate the free parameters to those from the previous sections. Instead we numerically optimize our formulations over the free parameters α and γ.

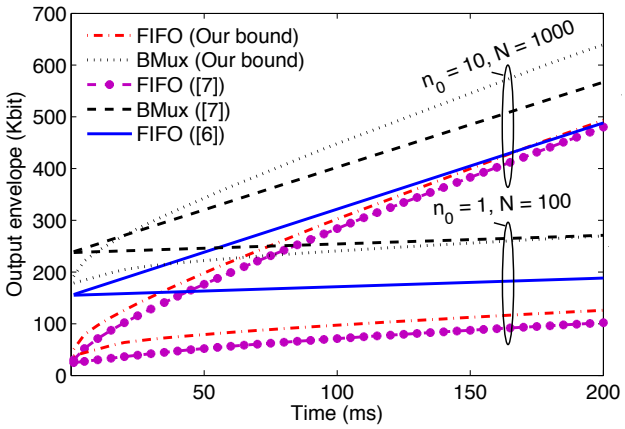


Fig. 3: Comparison of the output envelopes in a FIFO or BMux link with $n_0 = 1, 10$ through flows, $N = 100, 1000$ Mbps, $U = 90\%$, and $\varepsilon^* = 10^{-6}$. Each flow is an MMOO process with $P = 1.5$ Mbps and $T^* = 10$ ms.

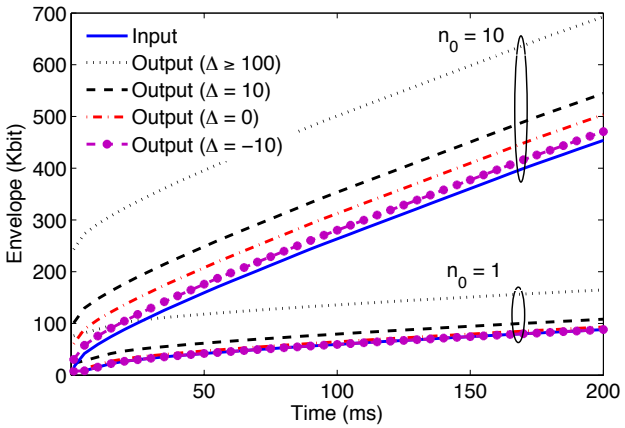


Fig. 4: Comparison of the input and output envelopes in a Δ-scheduler with $n_0 = 1, 10$ through flows, $C = 100$ Mbps, $U = 90\%$ ($N = 600$), and $\varepsilon^* = 10^{-6}$. Each flow is an MMOO process with $P = 1.5$ Mbps and $T^* = 10$ ms.

A. Output characterization

We first present our work for a single node. We consider the upstream node in Fig. 1a. In Fig. 3, we compare the output bound from Eq. (29) with the existing output envelopes in [5], [6]. Note that our model does not require the independence assumption between through and cross flows in [5], [6]. In addition, the FIFO output envelope in [6] is obtained by further assuming that the through flows arrivals in any time window are independent of the backlog at the start of that time window. The traffic mix of through and cross traffic is set by fixing the

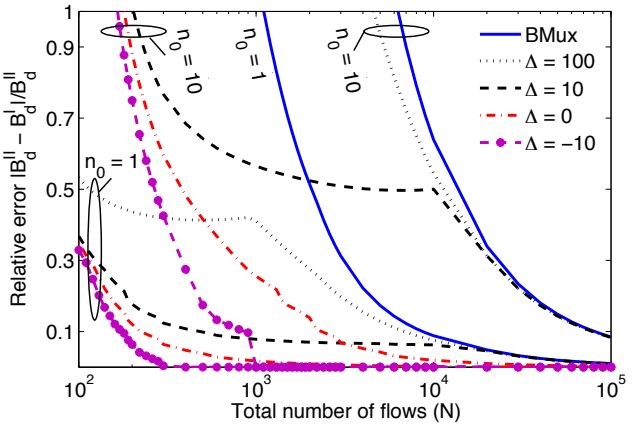


Fig. 5: Comparison of B_d^{I} with B_d^{II} by computing an upper bound on the relative difference between them as a function of N with $n_0 = 1$, $U = 90\%$, and $\varepsilon^* = 10^{-6}$. Each flow is an MMOO process with $P = 1.5$ Mbps and $T^* = 10$ ms.

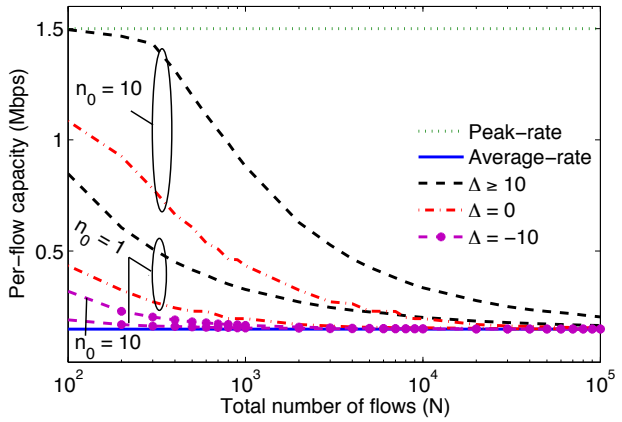


Fig. 6: The required per-flow capacity as a function of the total number of flows to meet a target overflow probability $\varepsilon^* = 10^{-6}$, with $n_0 = 1$ through flow and $N - n_0$ cross flows. Each flow is an MMOO process with $P = 1.5$ Mbps and $T^* = 10$ ms.

ratio $\frac{n_0}{N} = \frac{1}{100}$. In particular, we use parameters $(n_0 = 1, N = 100)$ and $(n_0 = 10, N = 1000)$. We find a statistical envelope (Eq. (1)) for the departures of through flows by taking the point-wise minimum of multiple EBB characterizations using different values of α. We compare the output envelopes from Eq. (30) with those from [5] and [6].

The graph shows that even though our model has fewer independence assumptions, our output envelopes are still comparable and in fact, often tighter than those in [5], [6]. In particular, only the corresponding envelope to the FIFO scheduler from [6] is smaller than our FIFO envelope for all time intervals.

In Fig. 4 we compare the input/output envelopes by fixing $C = 100$ Mbps and $U = 90\%$ $(N = 600)$ and computing the input and output envelopes for $n_0 = 1$ and $n_0 = 10$ for different schedulers. The plot shows that for a set of schedulers including FIFO, the output envelope is comparable to the input envelope even for the moderate value of N $(= 600)$ used in this example. However, for some other schedulers $(\Delta \geq 100)$ the output envelope is much larger than the input envelope, suggesting that for moderate values of N, the network decomposition might be valid for some schedulers and invalid for some others.

B. Decomposing a network of Δ-schedulers

In this example, we examine our analytical results on network decomposition by comparing the downstream backlog statistics in the downstream node in Scenario I with the backlog statistics in Scenario II in Fig. 1. We set the per-flow capacity such that the utilization at both upstream and downstream nodes are fixed to 90% independent of N.

We compare B_d^{I} with B_d^{II} by computing the probabilistic bound on B_d^{I} from Theorem 1 and the probabilistic upper bound on $|B_d^{\text{II}} - B_d^{\text{I}}|$ from Eq. (48). We plot the normalized difference $\frac{|B_d^{\text{II}} - B_d^{\text{I}}|}{B_d^{\text{II}}}$ in Fig. 5.

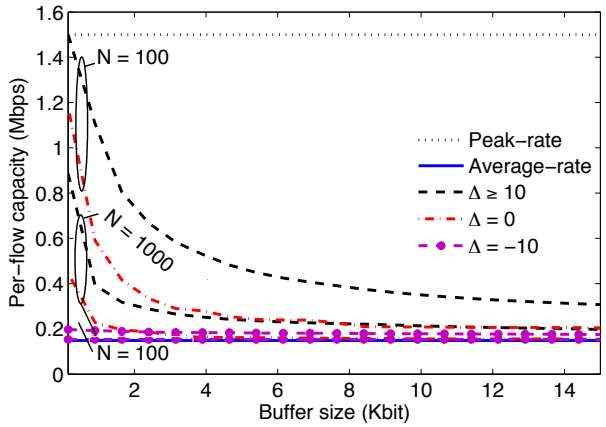


Fig. 7: The required per-flow capacity as a function of through flow buffer size when the buffer overflow is bounded by $\varepsilon^* = 10^{-6}$, with $n_0 = 1$ through flow and $N - n_0$ cross flows. Each flow is an MMOO process with $P = 1.5$ Mbps and $T^* = 10$ ms.

We use Eqs. (39)-(42) to compute $T^{max} = \max\{T_{\text{I}}(N\sigma), T_{\text{II}}(N\sigma)\}$. We set $\varepsilon^{T_{\text{I}}}, \varepsilon^{T_{\text{II}}} = \frac{\varepsilon^*}{N^2+2}$ and $\left\lceil \frac{T^{max}}{\tau_s} \right\rceil \varepsilon_b(\sigma - n_0 P_0 \tau_s)$ to $\frac{N^2 \varepsilon^*}{N^2+2}$, where ε_b is formulated in Eq. (11) and $\tau_s = \frac{T_{max}}{N^2}$. Then, the backlog bound from Eq. (9) with $N\sigma_0 = \frac{\sigma_c}{N} = \sigma$ and $\frac{\alpha_0}{N^2} = \alpha_c = \alpha$ is an upper bound on $|B_d^{\text{II}} - B_d^{\text{I}}|$. We use Eq. (11) to obtain the upper bound on B_d^{II}.

Fig. 5 shows that if N is not sufficiently large, scheduling algorithm is a key factor on the accuracy of estimating B_d^{I} by B_d^{II}. For instance, when $n_0 = 1$, for a set of schedulers $(\Delta \leq 10)$ including FIFO, the normalized error of estimating B_d^{I} by B_d^{II} $(\frac{|B_d^{\text{II}} - B_d^{\text{I}}|}{B_d^{\text{II}}})$ is less than 10% when N is only few hundred while, this is happening for BMux when N is as large as 10^4.

C. Capacity provisioning

In Fig. 6 we compute the required per-flow capacity for the upstream node in Fig. 1a to satisfy a target overflow probability ε^* using the optimization problem in Eq. (17). We have also included the per-flow peak-rate and average rate as benchmark values for the per-flow capacity. We fix the through flows buffer threshold to $b_0 = 1.5$ Kbits which is equal to the traffic generated in 1 ms from each flow in the On state. Then, for the choices of $n_0 = 1$ and $n_0 = 10$, we compute the required per-flow capacity c as a function of N. Fig. 6 exhibits the considerable impact of scheduling on the rate of convergence of the per-flow capacity to the average rate. For some schedulers including FIFO ($\Delta \leq 0$) the convergence happens when N is as small as few hundred, suggesting that a utilization close to 1 is achievable for those schedulers for moderate values of N and small buffer sizes. However, for some other schedulers ($\Delta \geq 10$) this happens only when N is much larger. By increasing the ratio of the through flows in the traffic mix, the required per-flow capacity increases substantially.

In Fig. 7 we let $N = 100$ and $N = 1000$ and compute the required per-flow capacity for through flow buffer sizes ranging from 0.15 Kbit to 15 Kbit. As shown in the figure, the required per-flow capacity decreases with a much faster pace (as buffer size increases) for small buffer sizes compared to large buffers. This plot shows that the impact of adding a small buffer on the buffer overflow decrease is substantially affected by the scheduling algorithm.

VI. CONCLUSIONS

Recent studies showed that the capacity and buffer provisioning in a scheduler in a many sources asymptotic regime is insensitive to the scheduling algorithm. In this paper, we have investigated the impact of scheduling on capacity provisioning when the buffer size is small. Using a non-asymptotic per-flow backlog bound formulation which applies to a large set of schedulers (Δ-schedulers) and assuming MMOO traffic sources, we showed that the scheduling algorithm determines the decay rate of the buffer overflow probability substantially for moderate values of N. Then, by fixing the buffer threshold to an arbitrary small value, we derived the required per-flow capacity which can satisfy a predefined overflow probability. We showed that the difference between the per-flow capacity and per-flow long-term average rate ranges from $O\left(\sqrt{\frac{\log N}{N}}\right)$ to $O\left(\frac{\log N}{N}\right)$ (and even $O(\frac{1}{N})$ when N is large enough) depending on the scheduling algorithm. Combining the observations from the numerical examples and the comparison of the scaling properties for different schedulers indicates that for moderate values of N, having the scheduling information is a key factor in capacity provisioning.

We have also considered the viability of network decomposition in a non-asymptotic regime for different schedulers. Numerical results show that for a class of schedulers including FIFO, the convergence can happen even when N is as small as few hundred. We analytically showed that the queue statistics of the downstream node in a two-node scenario converge to those of the simplified scenario when the upstream node is removed. The mode of convergence is almost sure for all schedulers, but the rate of convergence depends on the scheduling algorithm.

REFERENCES

[1] R. R. Bahadur and R. R. Rao. On deviations of the sample path mean. *The Annals of Mathematical Statistics*, 31(4):1015 – 1027, 1960.

[2] J. Y. Le Boudec and P. Thiran. *Network Calculus*. Springer Verlag, Lecture Notes in Computer Science, LNCS 2050, 2001.

[3] C. S. Chang. *Performance guarantees in communication networks*. Springer Verlag, 2000.

[4] F. Ciucu, A. Burchard, and J. Liebeherr. Scaling properties of statistical end-to-end bounds in the network calculus. *IEEE Transactions on Information Theory*, 52(6):2300 – 2312, June 2006.

[5] F. Ciucu and O. Hohlfeld. Scaling of buffer and capacity requirements for voice traffic in packet networks. In *Proc. of ITC-21*, pages 1 – 8, September 2009.

[6] F. Ciucu and J. Liebeherr. A case for decomposition of FIFO networks. In *Proc. of IEEE INFOCOM*, pages 1071 – 1079, April 2009.

[7] M. Enachescu, Y. Ganjali, A. Goel, N. McKeown, and T. Roughgarden. Part iii: routers with very small buffers. *SIGCOMM Computer Communication Review*, 35(3):83 – 90, July 2005.

[8] D. Y. Eun and N. B. Shroff. Network decomposition: theory and practice. *IEEE Transactions on Networking*, 13(3):526 – 539, June 2005.

[9] Y. Ghiassi-Farrokhfal and F. Ciucu. On the impact of finite buffers on per-flow delays in fifo queues. In *Proc. of ITC*, pages 1 – 8, September 2012.

[10] Y. Ghiassi-Farrokhfal, J. Liebeherr, and A. Burchard. The impact of link scheduling on long paths: Statistical analysis and optimal bounds. In *Proc. of IEEE INFOCOM*, pages 1242 – 1250, April 2011.

[11] Y. Jiang and Y. Liu. *Stochastic Network Calculus*. Springer Verlag, 2008.

[12] F. P. Kelly. Notes on effective bandwidths. In *Stochastic Networks: Theory and Applications. (Editors: F.P. Kelly, S. Zachary and I.B. Ziedins) Royal Statistical Society Lecture Notes Series, 4*, pages 141 – 168. Oxford University Press, 1996.

[13] C. Li, A. Burchard, and J. Liebeherr. A network calculus with effective bandwidth. *IEEE/ACM Transactions on Networking*, 16(6):1442 – 1453, December 2008.

[14] J. Liebeherr, Y. Ghiassi-Farrokhfal, and A. Burchard. Does link scheduling matter on long paths? In *Proc. of IEEE ICDCS*, pages 199 – 208, June 2010.

[15] N. Likhanov and R. R. Mazumdar. Cell loss asymptotics in buffers fed with a large number of independent stationary sources. In *Proc. of IEEE INFOCOM*, pages 339 – 346, March 1998.

[16] S. Mao and S. S. Panwar. Large deviations for small buffers: An insensitivity result. *Queueing Systems: Theory and Applications*, 37(4):349 – 362, March 2001.

[17] O. Ozturk, R. R. Mazumdar, and N. Likhanov. Many sources asymptotics for a feedforward network with small buffers. In *Proc. of Allerton conference, IL*, pages 353 – 358, October 2002.

[18] M. Schwartz. *Broadband Integrated Networks*. Prentice Hall, 1996.

[19] A. Weiss. A new technique for analyzing large traffic systems. *Performance Evaluation*, 18(2):506 – 532, June 1986.

[20] D. J. Wischik. The output of a switch, or, effective bandwidths for networks. 32(4), February 1999.

[21] O. Yaron and M. Sidi. Generalized processor sharing networks with exponentially bounded burstiness arrivals. In *Proc. of IEEE INFOCOM*, pages 628 – 634, June 1994.

[22] Y. Ying, R. Mazumdar, C. Rosenberg, and F. Guillemin. The burstiness behavior of regulated flows in networks. In *Proc. of Networking 2005*, pages 918 – 929, May 2005.