Wireless Netw (2009) 15:931–943
DOI 10.1007/s11276-007-0095-9

Analysis of performance trade-offs for an adaptive channel-aware wireless scheduler

Raphael Rom · Hwee-Pink Tan

Published online: 29 December 2007

Abstract In this paper, we consider the scheduling problem where data packets from K input-flows need to be delivered to K corresponding wireless receivers over a heterogeneous wireless channel. Our objective is to design a wireless scheduler that achieves good throughput and fairness performance while minimizing the buffer requirement at each wireless receiver. This is a challenging problem due to the unique characteristics of the wireless channel. We propose a novel idea of exploiting both the long-term and short-term error behavior of the wireless channel in the scheduler design. In addition to typical first-order Quality of Service (QoS) metrics such as throughput and average delay, our performance analysis of the scheduler permits the evaluation of higher-order metrics, which are needed to evaluate the buffer requirement. We show that variants of the proposed scheduler can achieve high overall throughput or fairness as well as low buffer requirement when compared to other wireless schedulers that either make use only of the instantaneous channel state or are channel-state independent in a heterogenous channel.

Keywords Adaptive wireless scheduling · QoS · Heterogenous channel · Buffer requirements · Fairness

This work was carried out when the author was a Ph.D. candidate in the Department of Electrical Engineering, Technion, Israel Institute of Technology.

R. Rom (✉)
Department of Electrical Engineering, Technion, Israel Institute of Technology, Technion City, Haifa 32000, Israel
e-mail: rom@ee.technion.ac.il

H.-P. Tan
CTVR, Trinity College Dublin, Dublin 2, Ireland
e-mail: tanhp@tcd.ie

1 Introduction

Wireless scheduling is an important enabler of Quality of Service (QoS) provisioning in wireless networks. Due to the dynamic nature of wireless channels, channel-state dependent (CSD) wireless schedulers (e.g., [1, 2]) only transmit to wireless receivers with *predicted* (instantaneous) error-free channels to optimize channel efficiency. Unlike recently proposed CSD schedulers that only exploit the instantaneous behavior of the wireless channel, we propose an adaptive CSD scheduler that exploits the long-term behavior (burstiness) as well. In addition to first order metrics such as throughput and average delay, our quantitative analysis allows the computation of second-order metrics essential for the evaluation of the wireless receiver buffer requirement.

We consider an infrastructured wireless network as depicted in Fig. 1(a). We assume an application such as localized content distribution, where fixed-size packets are dispatched according to some known distribution from fixed hosts, and are to be delivered to K wireless receivers (users). Upon arrival at the access point (AP) B, they are queued into K input-flows, where flow j comprises packets destined for user j. Due to cross-network traversal, the arrival distribution at each input-flow will be hard to evaluate, especially analytically. Hence, for our analysis, we assume that these input-flows are continuously backlogged and have equal priority to be selected by the AP for transmission.

A wireless scheduler is deployed at B to select the user to transmit to at each instant, as shown in Fig. 1(b). Our objective is to design such a scheduler that achieves a good trade-off amongst various QoS performance metrics. Stochastic channel error models are an important part of the performance evaluations of such wireless mechanisms, and there is typically a tradeoff between model complexity and

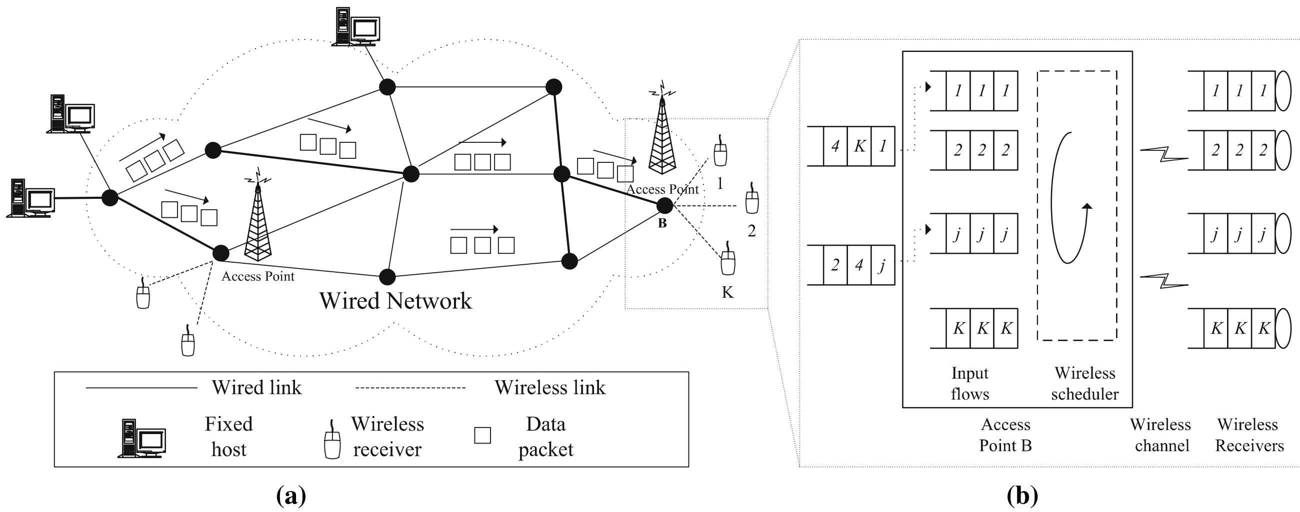


Fig. 1 (**a**) A generic wireless network where data packets are delivered to wireless receivers via access points and (**b**) an illustration of a wireless scheduling problem at an access point B

accuracy of performance evaluation. Although sophisticated models [3, 4] exist, an evaluation of these models [4] suggests that a simple two-state Gilbert-Elliot model [5] gives quite good results for *aggregate* metrics such as average packet delay. Moreover, the comparatively small gain achieved with complex models does not justify the increased model complexity (14, 30 or 40 states compared to 2). Since our performance evaluation is concerned with aggregate performance metrics, and our focus is on obtaining insight through analytical results, we use the Gilbert-Elliot model, and believe it could provide indicative results for the relative performance of various schedulers.

We first define the notations used in the rest of the paper, before describing the channel model, problem scenario and our proposed scheduler in the next few sections.

1.1 Notations

For any discrete variable x_i^j, the superscript j and subscript i are assigned to *flow* and *slot* indices, respectively. We denote by x^j the corresponding variable in steady-state, i.e., $x^j = \lim_{i\to\infty} x_i^j$. In addition, $\underline{x}^j$ and $\underline{x}_i$ are vectors that comprise the elements $\{x_i^j\}_{i=1}^I$ and $\{x_i^j\}_{j=1}^K$, respectively, where I and K are relevant spaces spanned by i and j, respectively.

We reserve the letter p for probability-related notations, where p^ε is the probability of occurrence of event ε and $p_{x_i^j}(X) \equiv \text{Prob}(x_i^j = X)$. Accordingly, $p_{x^j}(X)$ is the steady-state probability density function (pdf) of x^j. We use $E[x]$ and $Var[x]$ to denote the mean and variance of x, respectively.

1.2 Wireless channel model

For the Gilbert–Elliott model, the channel state of flow j in slot i, $c_i^j \in \{0, 1\}$, evolves according to a stationary Two-State Markov Chain (2SMC), where 0 and 1 correspond to *Good* and *Bad* states, respectively. If flow j attempts transmission, it will fail with probability $\gamma^j(c_i^j)$, where $\gamma^j(0) \ll \gamma^j(1)$. For the *simple* Gilbert–Elliott model considered in this study, we have $\gamma^j(0) = 0$ and $\gamma^j(1) = 1$.

We denote by $p_{c^j}(0) \equiv \lim_{i\to\infty} \text{Prob}(c_i^j = 0)$ the steady-state probability of flow j's channel being in state 0. This parameter varies according to the distance of user j from the AP: the further away it is, the smaller is the value of $p_{c^j}(0)$. It is an indication of the *quality* of the channel, where the upper bound to the throughput of flow j is given by $p_{c^j}(0)$.

We define g^j as the level of agility of the error behavior across successive slots for flow j, and it varies according to the mobility of user j as well as its environment. For small ε, we can categorize the channel according to g^j as follows:

$$g^j = \begin{cases} \epsilon, & \text{Persistent channel;} \\ 1, & \text{Uncorrelated channel;} \\ 2-\epsilon, & \text{Oscillatory channel.} \end{cases}$$

A *persistent* channel is one that is very likely to remain in the same state across successive slots (very slow fading), while the channel state in any slot in an *uncorrelated* channel is independent of the corresponding state in the previous slot (very fast fading). An *oscillatory* channel is one where the channel toggles from one state to another in successive slots.

1.3 A channel-heterogeneous wireless scheduling scenario

We consider a K-user channel-*heterogeneous* scheduling scenario, where user channels are independent and identical in terms of quality (i.e., $p_{c^j}(0) = p_c(0)$), but differ in terms of the agility. Such a scenario may arise when users are approximately equidistant from the AP but sufficiently separated spatially, and η users are quasi-stationary, e.g., in an office (persistent channels) while the other $K - \eta$ are mobile, e.g., in any moving vehicle (uncorrelated channels). Quantitatively, the scenario is specified by $K, 0 \le \eta \le K, p_c(0)$ and ε, where:

$$g^j = \begin{cases} \epsilon, & 1 \le j \le \eta; \\ 1.0, & \eta + 1 \le j \le K. \end{cases} \tag{1}$$

1.4 Performance metrics

For the scenario defined in Sect. 1.3, our objective is to design a wireless scheduler that achieves a good trade-off amongst the following performance metrics:

1.4.1 Overall throughput (T)

Let n^j denote the interval between consecutive successful packet transmissions (or Head-of-Line (HOL) packet delay) of flow j under steady-state conditions. The throughput of flow j, T^j, is given by the expected number of flow j packets successfully transmitted in each slot. Since each input-flow is continuously backlogged, $T^j = \frac{1}{n^j}$ and the overall throughput, T is:

$$T = \sum_{j=1}^{K} T^j = \sum_{j=1}^{K} \frac{1}{E[n^j]}. \tag{2}$$

Since wireless bandwidth is a scarce resource, it is desirable to maximize the overall throughput.

1.4.2 Throughput fairness (FM)

A good scheduler should maintain some level of fairness, i.e., where every flow expects to be treated fairly relative to any other flow. We define the notion of *worst-case* relative throughput fairness, *FM*, based on the Relative Fairness Bound [6] between any pair of flows (*j*,*k*) as follows:

$$FM = K \max_{1 \le j,k \le K} |T^j - T^k|, \tag{3}$$

where a small value of *FM* indicates good throughput fairness. It is challenging to maintain good throughput fairness, particularly in a heterogeneous scheduling environment.

1.4.3 Wireless receiver buffer requirement (b)

Let us assume that the content to be distributed in our application is streaming traffic e.g., voice. A jitter buffer is typically used at each wireless receiver to smooth the streaming playback, and overflow can occur due to excessive packet arrivals. The resulting packet losses create streaming gaps, which can result in clicks, muting or unintelligible speech. Therefore, it is important to cater for sufficient receiver buffer such that the level of overflow is within tolerable limits. However, increasing memory size in mobile devices, for example, has a substantial contribution to the total cost and adversely affects the power budget of these devices. Although the corresponding buffer requirement at access points is also important, they are not constrained by power budgets and the cost of memory does not contribute as significantly to the total cost of the AP. In addition, end-to-end flow control mechanisms such as TCP are well established and can be used to effectively regulate the packet sending rate out of fixed hosts to minimize buffer overflow at the AP.

Under high load conditions and assuming zero propagation delay in the wireless media, the minimum buffer size, b^j_{min}, to sustain a packet dropping rate, β, for flow j can be approximated by [7]:

$$b^j_{min} \approx \frac{\left\lceil \frac{\ln \beta}{\ln[Var[n^j] - 2E[n^j](1-\rho)] - \ln Var[n^j]} - 1 \right\rceil}{\rho \cdot E[n^j]}, \tag{4}$$

where ρ is the utilization factor at the wireless receiver and $\lceil y \rceil$ denotes the smallest integer greater than or equal to y. For a given $E[n^j]$ (i.e., given throughput), b^j_{min} increases with $Var[n^j]$, and hence, it is desirable for the wireless scheduler to have a small HOL packet delay variation.

1.5 Related work

Although the design of scheduling policies to meet QoS guarantees over a wired link is a well-studied problem ([8–10], to name a few), it is necessary to adapt these policies for QoS provisioning over a wireless link. One approach is to utilize feedback from each receiver to predict the *instantaneous* channel state (i.e., whether it is erroneous or error-free) and the *long-term* behavior (burstiness) of that channel. Channel efficiency can be optimized by restricting the candidates for transmission to those with *predicted* error-free channels in channel-state dependent (CSD) schedulers proposed in [1, 2]. In [11, 12], the authors proposed an exponential rule that optimizes the throughput for downlink scheduling in a CDMA system, where the channel information is embedded in the measured data rates.

A comprehensive survey of variants of CSD schedulers with different mechanisms for selecting the *instantaneous* 'best' flow to transmit while trading-off amongst various performance constraints such as throughput, fairness and delay can be found in [13]. In particular, the concept of 'compensation' is introduced in CSD schedulers proposed in [14–19] to achieve a trade-off between channel efficiency and *short-term fairness* provision. These schedulers can be mapped to the Unified Wireless-Fair Queueing (UWFQ) architecture proposed in [20], where an evaluation of *first-order* QoS metrics is carried out.

In prior work [21], we considered a special case of a channel-*homogeneous* scenario, i.e., $\eta = K$ in Eq. 1. In that work, a stochastic analysis of a CSD scheduler (see Sect. 2.1) is performed and the stationary HOL packet delay pdf is derived, from which various useful performance metrics are obtained. There, a channel-independent Fair-Aggregation (FA) scheduler is introduced, where packets from each input-flow are dispatched in a round robin manner into a single queue before FIFO transmission into the wireless media. It is deduced that while the FA scheduler achieves better QoS performance when the channel is uncorrelated, the CSD scheduler is superior when the channel is persistent. In [7], we developed a performance analysis framework to evaluate the HOL packet delay pdf of each flow for the CSD scheduler.

1.6 Contributions of this paper

In this paper, we propose an adaptive wireless scheduler that exploits the relative merits of the CSD and FA schedulers for a channel-heterogeneous environment. It does so by partitioning the users according to the burstiness of their channel, and then applying a different scheduling mechanism to each partition. Our performance analysis shows that variants of the adaptive scheduler can achieve a good balance between wireless receiver buffer requirements and throughput or fairness.

Hence, our contributions are two-fold: (a) Unlike recently proposed CSD schedulers that only exploit the instantaneous behavior of the wireless channel, our scheduler introduces the novel concept of exploiting the long-term behavior (burstiness) as well and (b) Contrary to prior work on QoS analysis that focused on first-order metrics such as throughput and average delay, our quantitative analysis allows the computation of second-order metrics essential for the evaluation of the wireless receiver buffer requirement.

The rest of the paper is organized as follows: In Sect. 2, we define our proposed adaptive scheduler which is analyzed in Sect. 3. Numerical results that illustrate the trade-off amongst buffer requirement, throughput and fairness amongst various schedulers are presented in Sect. 4. In Sect. 5, we discuss the impact that channel parameter estimation has on the scheduler's performance and how other forms of heterogeneity e.g., user-heterogeneity can be incorporated in the current analysis. Concluding remarks are presented in Sect. 6.

2 An adaptive channel-state dependent scheduler for heterogeneous channels

For the scenario defined by Eq. 1, we propose a novel adaptive CSD scheduler that achieves the relative merits of CSD and FA scheduling by partitioning the input-flows into two groups, ($\mathbf{C}^1$,$\mathbf{C}^2$) according to g^j and applying the respective scheduling mechanism to each group. We denote such an adaptive scheduler as a (K,η) CSD-FA scheduler, whose architecture is shown in Fig. 2(a).

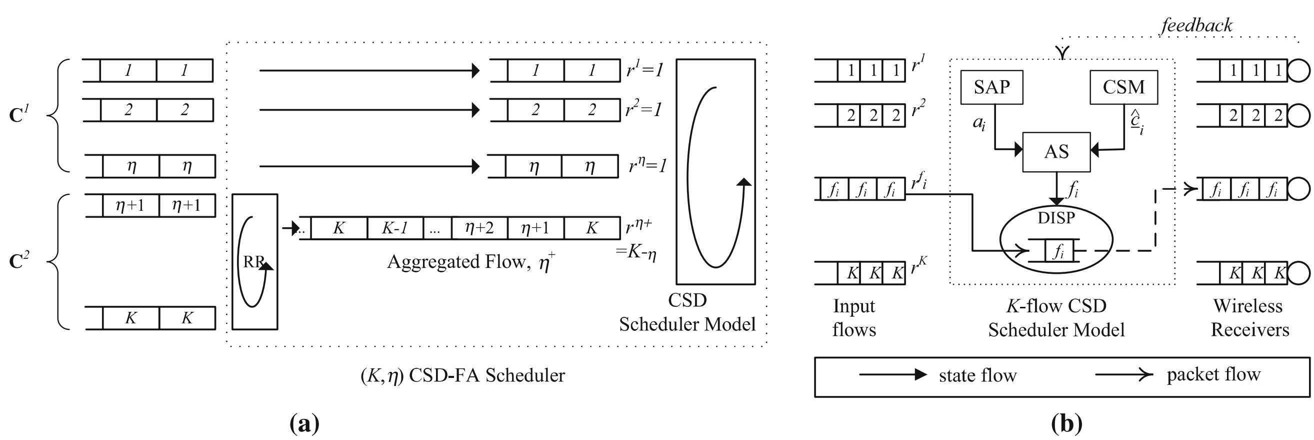


Fig. 2 (**a**) (K,η) CSD-FA Scheduler Model: Flows in $\mathbf{C}^2$ are aggregated into a single flow η^+; Flows in $\mathbf{C}^1 \cup \eta^+$ are then scheduled by a $\eta + 1$-flow weighted Channel-State Dependent (CSD) scheduler with $\underline{r} = [1, \ldots, 1, K - \eta]$ and $\underline{g} = [\epsilon, \ldots, \epsilon, 1.0]$ and (**b**) CSD scheduler model, with illustration of state flow, downlink packet flow (dashed) and uplink packet flow (dotted) in slot i

2.1 Mechanism of (K,η) CSD-FA Scheduler

The mechanism of the scheduler can be described in two stages (refer to Fig. 2(a)). In the first stage, the scheduler dispatches packets from flows in $\mathbf{C}^2$ in a round robin manner into a single queue. If we denote this queue by η^+, then the second stage comprises a η+1-flow *weighted* CSD scheduler (with flow composition given by $\mathbf{C}^1 \cup \eta^+$), with *weights* given by $\underline{r} = [1, \ldots, 1, K - \eta]$.

We consider a CSD scheduler model that is similar to the one defined in [1] and maps to the UWFQ architecture defined in [20]. It comprises a Slot Allocation Policy (SAP), a Channel Status Monitor (CSM), an Arbitration Scheme (AS) and a Packet Dispatcher (DISP), as depicted in Fig. 2(b). The SAP determines the mechanism of the scheduler under *error-free* conditions. We consider a simple Weighted Round Robin (WRR) SAP that allocates slots to each flow according to $\underline{r}$. In this case, it cyclically allocates one slot each to flow $j \in \mathbf{C}^1$ followed by $K - \eta$ slots to flow η^+.

The CSM maintains c_{i-1}^j based on feedback (assumed to be error-free) from wireless receiver j and uses it as the predicted channel state in slot i, i.e., $\hat{c}_i^j = c_{i-1}^j$. To maximize channel efficiency, while trying to emulate the SAP under error-prone conditions, the AS selects a flow f_i for transmission as follows:

$$f_i = \begin{cases} a_i, & a_i \in \mathbf{G}_i; \\ Arb(\mathbf{G}_i), & \text{otherwise}, \end{cases} \tag{5}$$

where $\mathbf{G}_i = \{\arg_{1 \le m \le K} \hat{c}_i^m = 0\}$ contains the set of *eligible* flows that are likely to transmit successfully and a_i is the flow allocated by the SAP in slot i. We consider a simple *uniform* arbitration rule as follows:

$$\text{Prob}(Arb(\mathbf{G}_i) = j) = \begin{cases} \frac{1}{|\mathbf{G}_i|}, & j \in \mathbf{G}_i; \\ 0, & \text{otherwise}. \end{cases} \tag{6}$$

The DISP dispatches the HOL packet of flow f_i for transmission. Due to imperfect channel prediction, packet transmissions may fail, and the choice of an ARQ mechanism for re-transmission is important since it affects the QoS performance of the wireless scheduler. In this study, we consider a simple Stop-and-wait ARQ [22], where packets for re-transmission are indistinguishable from newly-arrived packets, and re-transmission takes place until a packet is successfully transmitted.

2.2 Illustration of Mechanism of (K,η) CSD-FA Scheduler

We illustrate the mechanism of our proposed scheduler by considering a (4,2) CSD-FA scheduler, which is equivalent to a 3-flow CSD scheduler with $\underline{g} = [\epsilon, \epsilon, 1.0]$ and $\underline{r} = [1, 1, 2]$, as depicted in Fig. 3(a). According to the WRR allocation policy, the allocation sequence, $\underline{a}$, is given as follows:

$$\underline{a} = [\ldots, 2, 2^+, 2^+, 1, 2, 2^+, 2^+, 1, \ldots]. \tag{7}$$

Let us assume the following initial conditions: $a_0 = 1$ and a flow 3 packet is HOL at flow 2^+ at the end of slot 0. If TX_i denotes the flow index of the packet transmitted in slot i, then the evolution of TX corresponding to some channel process is depicted in Fig. 3(b).

Since $a_0 = 1$, according to Eq. 7, $a_1 = 2$; similarly, since $c_0^2 = 0, \hat{c}_1^2 = 0$. Hence, according to Eq. 5, flow 2 is selected for transmission. However, since $c_1^2 = 1$, the transmission is unsuccessful. The next slot is allocated to flow 2^+. Since the HOL packet of flow 2^+ belongs to flow 3 and $c_1^3 = 0$, flow 2^+ is selected for transmission. The transmission is successful since $c_2^3 = 0$.

Slot 3 is again allocated to flow 2^+ according to Eq. 7. However, since its HOL packet belongs to flow 4 and $c_2^4 = 1$, $\hat{c}_3^4 = 1$, and hence its transmission is deferred. Since $c_2^1 = c_2^2 = 0$, $\hat{c}_3^1 = \hat{c}_3^2 = 0$, and according to Eq. 6, flow 1 and 2 are equally likely to be selected for transmission. We assume that flow 2 is selected, and its transmission is successful since $c_3^2 = 0$. Subsequent values of TX can be evaluated in a similar manner.

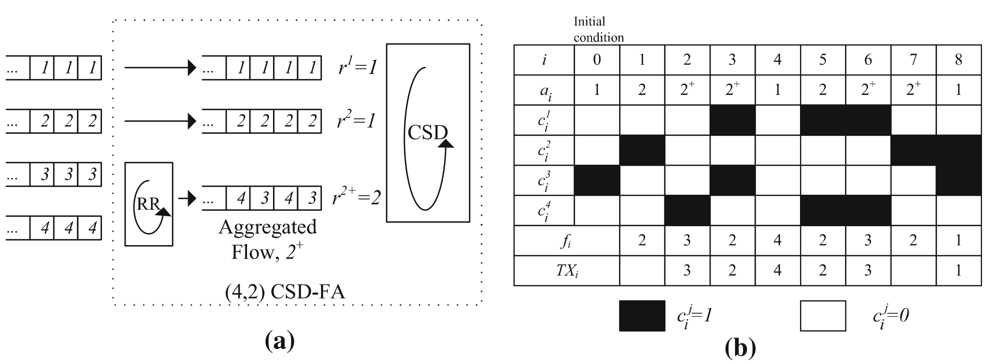


Fig. 3 (**a**) Architecture and (**b**) illustration of the mechanism of a (4,2) CSD-FA scheduler

3 Performance ansalysis

We derive the performance metrics defined in Sect. 1.4 for the (K,η) CSD-FA, K-flow CSD and K-flow FA schedulers for the channel-heterogeneous scenario defined by Eq. 1 in this section.

3.1 Throughput and fairness performance of (K,η) CSD-FA scheduler

Let us define the following probabilistic parameters:

$p_{\mathcal{D}} \equiv$ Prob(a flow defers its transmission attempt)

$p_{\mathcal{S}^1|m} \equiv$ Prob(a flow of $\mathbf{C}^1$ transmits successfully given that m other eligible flows exist)

$p_{\mathcal{S}^{\eta^+}|m} \equiv$ Prob(flow η^+ transmits successfully given that m other eligible flows exist)

The throughput and fairness performance of the (K,η) CSD-FA Scheduler can be expressed in terms of $(p_{\mathcal{D}}, p_{\mathcal{S}^1|0}, p_{\mathcal{S}^{\eta^+}|0})$ according to the following theorem (See Appendix A for proof):

Theorem 1 *For the scheduling scenario defined in Eq. 1, the per-flow throughput and worst-case unfairness metric achieved by the (K,η) CSD-FA scheduler are given as follows*:

$$T^j = \begin{cases} \frac{p_{\mathcal{S}^1|0}}{K} + \frac{(K-1)p_{\mathcal{S}^1|0}p_{\mathcal{D}}(1-p_{\mathcal{D}}^{\eta})}{K\eta(1-p_{\mathcal{D}})}, & j \in \mathbf{C}^1; \\ \frac{p_{\mathcal{S}^{\eta^+}|0}}{K} + \frac{p_{\mathcal{S}^{\eta^+}|0}p_{\mathcal{D}}(1-p_{\mathcal{D}}^{\eta})}{K(K-\eta)(1-p_{\mathcal{D}})}, & j \in \mathbf{C}^2. \end{cases}$$

$$FM = |p_{\mathcal{S}^1|0} - p_{\mathcal{S}^{\eta^+}|0} + \frac{p_{\mathcal{D}}(1-p_{\mathcal{D}}^{\eta})}{1-p_{\mathcal{D}}}[\frac{(K-1)p_{\mathcal{S}^1|0}}{\eta} - \frac{p_{\mathcal{S}^{\eta^+}|0}}{K-\eta}]|,$$

where

$$p_{\mathcal{D}} = 1 - p_c(0),$$
$$p_{\mathcal{S}^1|0} = p_c(0)(1 - \epsilon + \epsilon \cdot p_c(0)),$$
$$p_{\mathcal{S}^{\eta^+}|0} = p_c^2(0).$$

Substituting Theorem 1 into Eq. 2, $T_{\text{CSD-FA}}$ can be evaluated and is given in Eq. 8.

$$T_{\text{CSD}-\text{FA}} = \frac{1}{K}[\eta \cdot p_{\mathcal{S}^1|0} + (K-\eta)p_{\mathcal{S}^{\eta^+}|0} + \frac{p_{\mathcal{D}} - p_{\mathcal{D}}^{\eta+1}}{1-p_{\mathcal{D}}}((K-1)p_{\mathcal{S}^1|0} + p_{\mathcal{S}^{\eta^+}|0})]. \tag{8}$$

3.2 Wireless receiver buffer requirement for (K,η) CSD-FA scheduler

According to Sect. 2.1, our (K,η) CSD-FA scheduler is equivalent to a $\eta + 1$ flow weighted CSD scheduler, for which the framework of [7] can be applied to evaluate $p_{n^j}(N^j)$ for each flow $j \in \mathbf{C}^1 \cup \eta^+$. Using $p_{n^j}(N^j)$, $(E[n^j], Var[n^j])$ can be computed, from which the wireless receiver buffer requirement can be evaluated using Eq. 4.

Let us consider flow η^+, which is an aggregate of the flows in $\mathbf{C}^2$. To obtain the statistics of each flow $j \in \mathbf{C}^2$ from $p_{n^{\eta^+}}(N^j)$, we consider the transmission sequence of flow η^+ that begins and terminates with successive flow $\eta + 1$ packets. According to Fig. 4, the inter-packet departure delay of flow $\eta + 1$, $n^{\eta+1}$ can be written as follows:

$$n^{\eta+1} = \sum_{j=\eta+1}^{K} n_A^j, \tag{9}$$

where n_A^j is the HOL delay of flow η^+, given that the packet transmitted belongs to flow $j \in \mathbf{C}^2$.

Since $\{n_A^j\}_{j=\eta+1}^K$ are identically distributed according to $p_{n^{\eta^+}}(N^j)$ with mean $E[n^{\eta^\wedge+}]$, we have the following expression for $E[n^j]$, $j \in \mathbf{C}^2$:

$$E[n^j] = (K-\eta)E[n^{\eta^+}].$$

The evaluation of $Var[n^j]$, $j \in \mathbf{C}^2$ is not straightforward since $\{n_A^j\}_{j=\eta+1}^K$ are mutually dependent. However, $Var[n^j]$ for each flow $j \in \mathbf{C}^2$ can be evaluated [23] if flow η^+ is permitted to transmit *only* in slots allocated to it.

3.3 Performance evaluation of K-flow FA scheduler

The analysis in Sect. 3.2 for the flows in $\mathbf{C}^2$ for the (K,η) CSD-FA scheduler can be used to evaluate ($E[n^j]$,$Var[n^j]$) for each flow j for the K-flow FA scheduler with $\eta = 0$. In this case, Eq. 9 can be written as follows for $1 \le j \le K$:

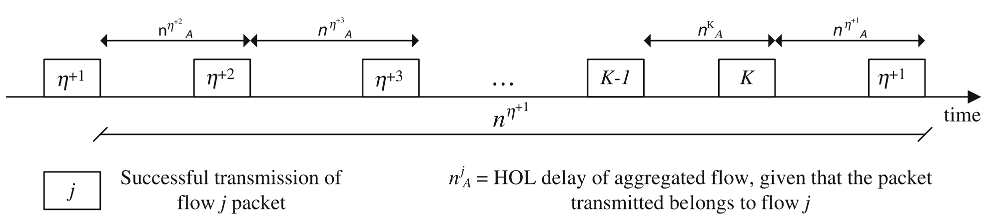


Fig. 4 Transmission sequence of flow η^+ that begins and terminates with successive flow $\eta + 1$ packets

$$n^j = \sum_{j=1}^{K} n_A^j.$$

Since $\{n_A^j\}_{j=1}^K$ are mutually independent, $p_{n_A^j}(N^j)$ can be evaluated independently for each j and is given as follows:

$$p_{n_A^j}(N^j) = \begin{cases} p_c(0), & N^j = 1; \\ \epsilon \cdot p_c(0)(1-p_c(0)) \cdot & j \le \eta, \\ (1-\epsilon p_c(0))^{N^j-2}, & N^j > 1; \\ p_c(0)(1-p_c(0))^{N^j-1}, & j > \eta, \\ & N^j > 1. \end{cases}$$

From $p_{n_A^j}(N^j)$, we obtain the overall throughput and fairness performance of the K-flow FA scheduler as follows:

$$T_{\mathrm{FA}} = \frac{K\epsilon p_c(0)}{\eta - p_c(0)\eta(1-\epsilon) + (K-\eta)\epsilon},$$
$$FM_{\mathrm{FA}} = 0.$$

We note that the FA scheduler is throughput-fair regardless of the parameters of the scenario. In addition, we have the following:

$$E[n^j] = E[n^k] = \sum_{m=1}^{K} E[n_A^m],$$

$$Var[n^j] = Var[n^k] = \sum_{m=1}^{K} Var[n_A^m],$$

which can be substituted into Eq. 4 to obtain b_{FA}^j.

3.4 Performance evaluation of K-flow CSD-scheduler

Using the framework in [7], we can obtain the corresponding expressions for the throughput and worst-case unfairness metric with a K-flow CSD scheduler as follows:

$$T_{\mathrm{CSD}}^j = \begin{cases} \frac{p_{\mathcal{S}^1|0}(1-p_{\mathcal{D}}^K)}{K(1-p_{\mathcal{D}})}, & j \in \mathbf{C}^1; \\ \frac{p_{\mathcal{S}^{\eta^+}|0}(1-p_{\mathcal{D}}^K)}{K(1-p_{\mathcal{D}})}, & j \in \mathbf{C}^2, \end{cases}$$

$$FM_{\mathrm{CSD}} = \frac{(1-p_{\mathcal{D}}^K)}{1-p_{\mathcal{D}}} |p_{\mathcal{S}^1|0} - p_{\mathcal{S}^{\eta^+}|0}|.$$

The expression for $T_{\mathrm{CSD}} = \sum_{j=1}^{K} T_{\mathrm{CSD}}^j$ is therefore:

$$T_{\mathrm{CSD}} = \frac{1-p_{\mathcal{D}}^K}{(1-p_{\mathcal{D}})K}[\eta \cdot p_{\mathcal{S}^1|0} + (K-\eta)p_{\mathcal{S}^{\eta^+}|0}].$$

Using the framework derived in [7], we can obtain (E[n^j], Var[n^j]), from which b_{CSD}^j can be computed using Eq. 4.

4 Numerical results

In this section, we evaluate and compare the performance of various schedulers for a K-user channel-heterogeneous scheduling scenario specified by Eq. 1 for $0.5 \le p_c 0 \le 0.9$ and $0.1 \le \epsilon \le 1.0$.

According to Sect. 1.4, for a given K, β and ρ, performance metrics depend on the flow composition, η, as well as the channel parameters, $p_c(0)$ and ε. Unless otherwise stated, representative numerical results are presented for $K = 7$, $\eta = 3$, $p_c(0) = 0.9$, $\varepsilon = 0.1$, $\beta = 0.01$ and $\rho = 0.99$.

4.1 Performance comparison amongst CSD, CSD-FA and FA schedulers

Based on the analysis in Sect. 3, we evaluate and compare the overall throughput, T, throughput unfairness, FM as well as the *average* wireless receiver buffer requirement, $\check{b}$, amongst the (K,η) CSD-FA, K-flow CSD and K-flow FA schedulers, where $\check{b}_\pi$ (corresponding to scheduler π) is defined as follows:

$$\check{b}_\pi = \frac{1}{K} \sum_{j=1}^{K} b_\pi^j,$$

where b_π^j is the wireless receiver buffer requirement for flow j with scheduler π. For the CSD-FA scheduler, we impose transmission restrictions on the aggregate flow η' such that it can only transmit in slots allocated to it (see Sect. 3.2).

While the FA scheduler is throughput-fair ($FM_{\mathrm{FA}} = 0$), we expect the CSD scheduler to be *more* throughput-fair than the CSD-FA scheduler. This is because flows in $\mathbf{C}^1$($\mathbf{C}^2$) would achieve a *higher* (lower) throughput in the CSD-FA scheduler than the CSD scheduler since transmission opportunities lost by each flow in $\mathbf{C}^2$($\mathbf{C}^1$) are available to $\mathbf{C}^1$ *flows only* (all other flows). Since this is consistent with the numerical results presented in the following subsections, we will focus our discussion on the comparison of throughput and buffer requirement for various scenarios.

4.1.1 Scenario A: variation of ε

We consider a scenario comprising users with good channel conditions, and investigate the impact of ε on $\check{b}$, T and FM in Fig. 5. We note that a value of ε close to 0 (1.0) indicates a channel-*heterogeneous* (homogeneous) scenario.

For the CSD schedulers, as ε is decreased, the throughput of each flow $\in \mathbf{C}^1$ is increased since the accuracy of channel prediction for flows in $\mathbf{C}^1$ is increased,

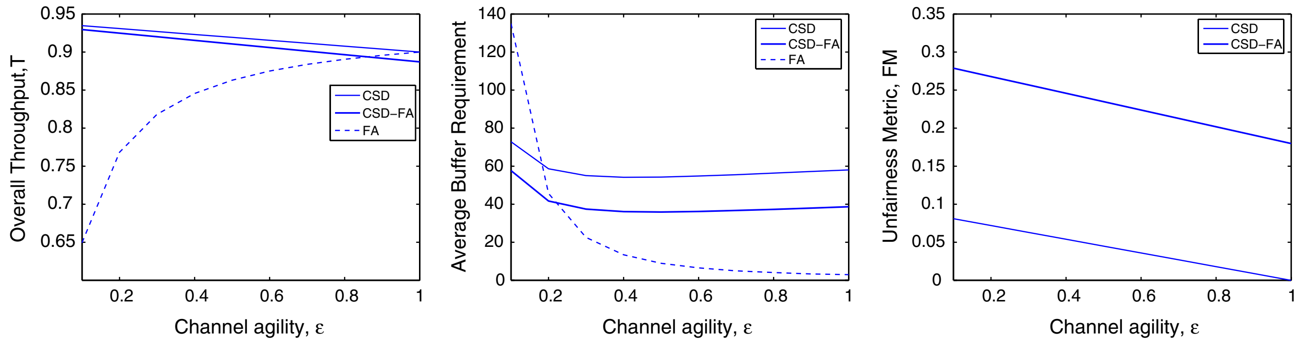


Fig. 5 Effects of channel agility on overall throughput (left), average buffer requirement (center) and throughput unfairness (right) of various schedulers for $\eta = 3$ and $p_c(0) = 0.9$

reducing the likelihood of erroneous transmissions. Since the throughput of each flow $\in \mathbf{C}^2$ is invariant with ε, FM_{CSD} and $FM_{\text{CSD-FA}}$ is increased. On the other hand, a reduction in ε causes more severe HOL blocking with FA scheduling, resulting in poorer throughput and buffer performance.

When channel conditions are good, the FA scheduler performs best when user channels are homogeneous and uncorrelated ($\varepsilon \approx 1.0$); however, in channel heterogeneous scenarios, the CSD schedulers achieve similar throughput levels, with the CSD-FA scheduler having a lower buffer requirement.

4.1.2 *Scenario B: variation of* $p_c(0)$

Next, we consider a channel-heterogeneous scenario and investigate the impact of $p_c(0)$ on $\check{b}$, T and FM in Fig. 6.

When channel conditions are poor, the CSD-FA scheduler achieves a higher throughput than the CSD scheduler at the expense of higher buffer requirement. However, as channel conditions are improved, the performance of both schedulers are improved, since flows are more likely to transmit in slots allocated to them, resulting in high throughput and low delay variance (low buffer requirement). Although the FA scheduler has a relatively low and constant buffer requirement under the range of channel conditions, its throughput is significantly lower than that obtained with the CSD schedulers.

In terms of throughput fairness, when channel quality is degraded, flows $\in \mathbf{C}^1$ can benefit from the transmission opportunities given up by flows $\in \mathbf{C}^2$, giving rise to a larger margin between the throughput of flows $\in \mathbf{C}^1$ and $\mathbf{C}^2$ (i.e., larger $FM_{\text{CSD-FA}}$).

4.1.3 *Scenario C: variation of* η

Lastly, we consider a channel-heterogeneous scenario and investigate the impact of η on $\check{b}$, T and FM in Fig. 7.

While both CSD schedulers achieve similar throughput levels, the CSD-FA scheduler achieves a savings in buffer requirement as the size of $\mathbf{C}^1$, η, is reduced. The FA scheduler performs worse than the CSD schedulers in terms of throughput and buffer requirement regardless of the flow composition.

In terms of throughput fairness, when η is reduced, fewer flows $\in \mathbf{C}^1$ contend for more transmission opportunities given up by flows $\in \mathbf{C}^2$, giving rise to a larger margin between the throughput of flows $\in \mathbf{C}^1$ and $\mathbf{C}^2$ (i.e., larger $FM_{\text{CSD-FA}}$).

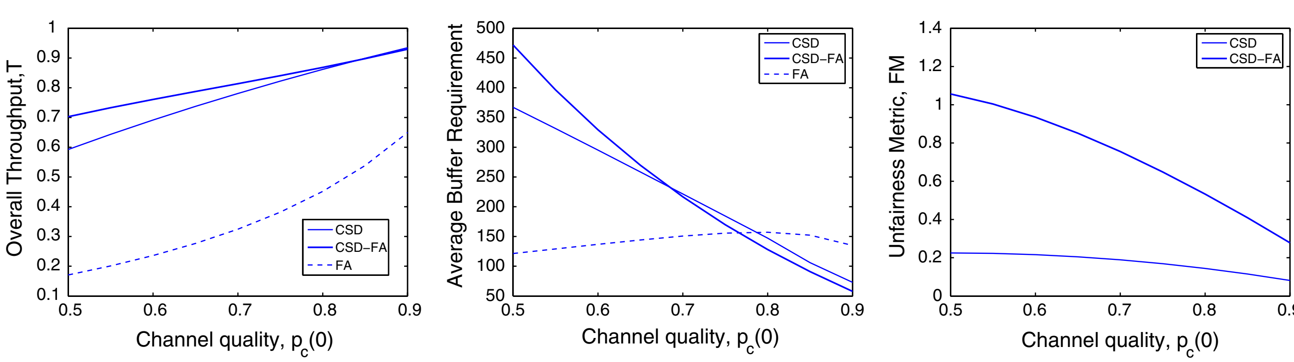


Fig. 6 Effects of channel quality on overall throughput (left), average buffer requirement (center) and throughput unfairness (right) of various schedulers for $\eta = 3$ and $\varepsilon = 0.1$

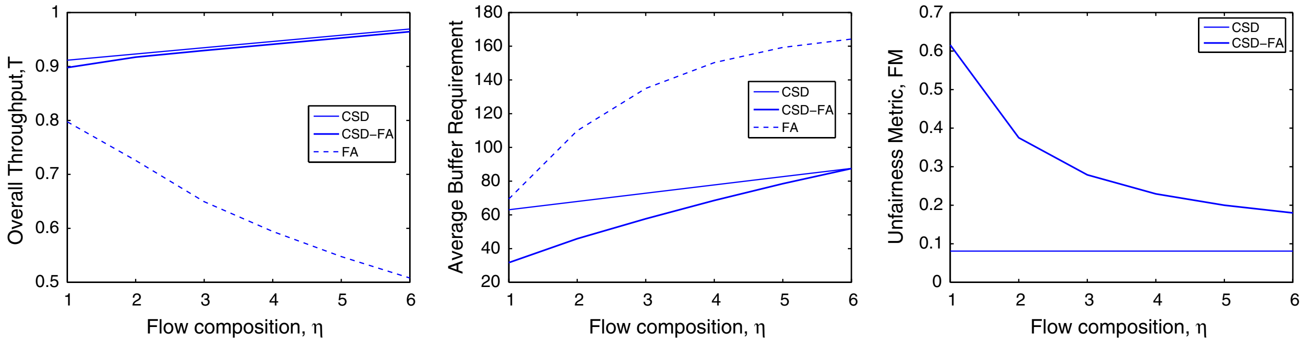


Fig. 7 Effects of flow composition on overall throughput (left), average buffer requirement (center) and throughput unfairness (right) of various schedulers for $p_c(0) = 0.9$ and $\varepsilon = 0.1$

4.1.4 Impact of user mobility and channel variations

Thus far, we have assumed a *static* scheduling scenario characterized by K, $p_c(0)$ and $\underline{g} = \{g^1, g^2, ..., g^K\}$. While K remains constant as long as input queues are continuously backlogged, the channel parameters may change over the packet transfer duration due to user mobility or channel variations. This may result in the migration of a user from $\mathbf{C}^1$ to $\mathbf{C}^2$ and vice versa (i.e., variation in η) and its impact on the scheduler performance can be assessed by the results presented for Scenario C. Similarly, user mobility resulting in variations in ε or $p_c(0)$ can be manifested in Scenario A and B, respectively and the impact on the scheduler performance can be assessed accordingly.

However, to adapt the scheduling mechanism to each new scenario, the channel parameters have to be accurately measured based on traces collected from the wireless receivers [3]. The scheduler would be operating according to *stale* channel parameters during this adaptation period. We demonstrate the resulting performance deviation due to migration of a user from $\mathbf{C}^1$ to $\mathbf{C}^2$ and vice versa in Figs. 8 and 9, respectively.

According to Figs. 8 and 9, the migration of a user from $\mathbf{C}^x$ to $\mathbf{C}^y$ has no impact on the performance of users in $\mathbf{C}^2$. This is because each flow in $\mathbf{C}^2$ is allocated the same proportion of slots before and after the user migration and are permitted to transmit only in these allocated slots.

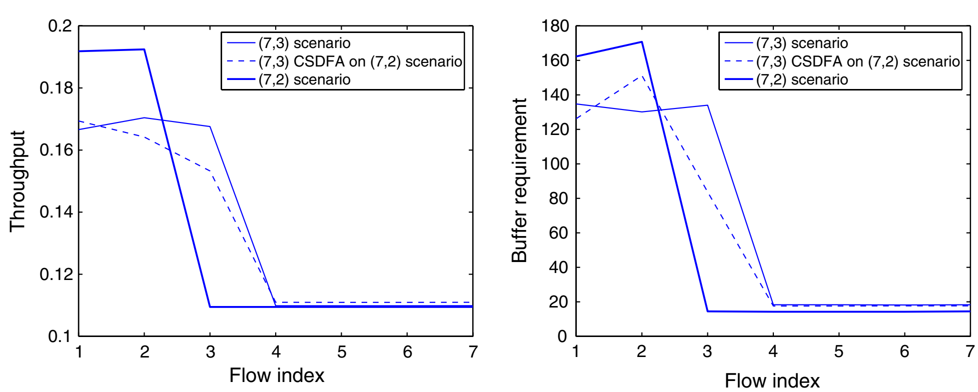
After a user migrates to $\mathbf{C}^2$ ($\mathbf{C}^1$), users in $\mathbf{C}^1$ ($\mathbf{C}^2$) receive *higher* (lower) throughput since there are *more transmission opportunities* (fewer slots given up by $\mathbf{C}^2$ flows) and *less* (more) competition for these slots. Consequently, the throughput of the migrating user is *reduced* (increased) to the level of the *aggregated* (non-aggregated) flows. The observations in terms of buffer requirement can be explained in a similar way.

During the adaptation interval, after a user migrates to $\mathbf{C}^2$, the CSD-FA scheduler *underestimates* (overestimates) the throughput and buffer requirement of *users 1 and 2* (user 3). On the other hand, after a user migrates to $\mathbf{C}^1$, the CSD-FA scheduler *overestimates* (underestimates) the buffer requirement of *all users except user 4* (user 4) and *overestimates* (underestimates) the throughput of *users 1-3* (user 4).

4.2 Performance comparison of CSD-FA schedulers

In the CSD-FA scheduler considered thus far, we imposed transmission restrictions on flow η^+ such that it can only transmit in slots allocated to it in order to obtain a

Fig. 8 Impact of user mobility from $\mathbf{C}^1$ to $\mathbf{C}^2$ on per-flow throughput (left) and buffer requirement (right) of CSD-FA scheduler for $K = 7$, $\eta = 3$, $p_c(0) = 0.9$ and $\varepsilon = 0.1$



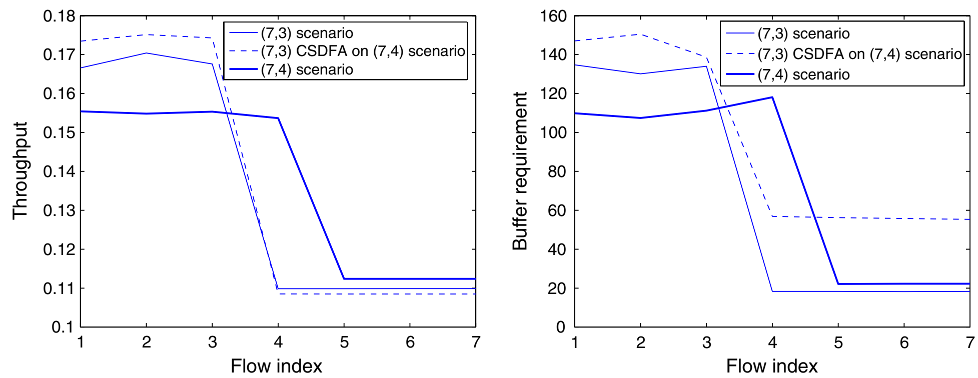


Fig. 9 Impact of user mobility from $\mathbf{C}^2$ to $\mathbf{C}^1$ on per-flow throughput (left) and buffer requirement (right) of CSD-FA scheduler for $K = 7$, $\eta = 3$, $p_c(0) = 0.9$ and $\varepsilon = 0.1$

closed-form expression for the wireless receiver buffer requirement (see Sect. 3.2). However, other variants based on different transmission restrictions are possible:

- CSD-FA^1: In this variant, flows in $\mathbf{C}^1$ are only permitted to transmit in slots allocated to $\mathbf{C}^1$.
- CSD-FA^2: Here, flows in $\mathbf{C}^x$ are restricted to transmit in slots allocated to $\mathbf{C}^x$ for $x = 1,2$.
- CSD-FA^3: There are no transmission restrictions imposed in this variant.

Since the wireless receiver buffer requirement cannot be obtained analytically for the above variants, we obtain their performance using discrete event simulation. We simulate a K-flow wireless scheduling scenario over a duration of 10,000 slots, using each variant of CSD-FA scheduler. We store the HOL delay of each flow j, n^j (in slots), from which the sample mean and variance, ($E[n^j]$, Var$[n^j]$) is computed and used to determine each performance metric according to Sect. 1.4.

We compare the per-flow throughput and buffer requirement of each scheduler for $\eta = 3$, and compare the overall throughput-fairness for $1 \leq \eta \leq K - 1$, and the results are plotted in Fig. 10. Based on performance, we can group the CSD-FA schedulers as $\mathbf{G}^1 = \{$CSD-FA, CSD-FA$^3\}$ and $\mathbf{G}^2 = \{$CSD-FA^1, CSD-FA$^2\}$, where transmission restrictions are (not) imposed on $\mathbf{C}^1$ in $\mathbf{G}^1$ ($\mathbf{G}^2$). When a flow in $\mathbf{C}^1$ gives up its allocated slot, the likelihood of flow η^+ utilizing that slot is small since there are $\eta - 1$ other competing flows. This suggests that imposing transmission restrictions on flow η^+ has marginal impact on the scheduler's performance, which explains the grouping.

On the other hand, a slot given up by the aggregate flow due to bad channels is highly likely to be used by a flow in $\mathbf{C}^1$, and hence, imposing transmission restrictions on the latter flows (in $\mathbf{G}^2$ schedulers) will incur a significant reduction in throughput. Hence, schedulers in $\mathbf{G}^1$ achieve higher throughput at the expense of higher buffer requirement and throughput unfairness compared to $\mathbf{G}^2$ schedulers.

4.3 Impact of packet arrival statistics

In the last 2 sections, we evaluate the schedulers' performance by assuming that each input queue is continuously backlogged. Here, we investigate the impact of packet arrival statistics (Poisson arrivals, λ packets per slot) on the performance achieved by the scheduler. We compare the per-flow throughput and buffer requirement of each scheduler for $\eta = 3$ and the results are plotted in Fig. 11.

We observe that the per-user throughput obtained with the *always backlogged* assumption approximates the corresponding performance obtained with Poisson arrivals

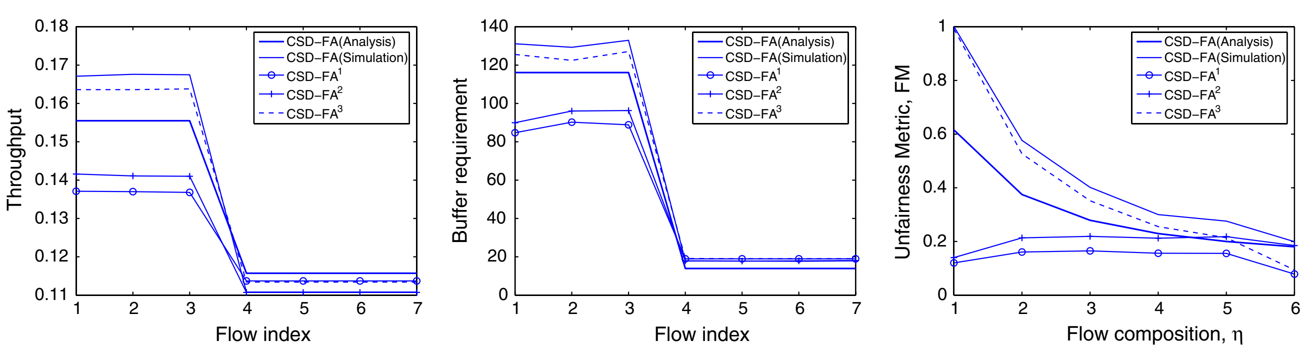


Fig. 10 Comparison of per-flow throughput (left), buffer requirement (center) and throughput unfairness (right) of CSD-FA schedulers for $p_c(0) = 0.9$ and $\varepsilon = 0.1$

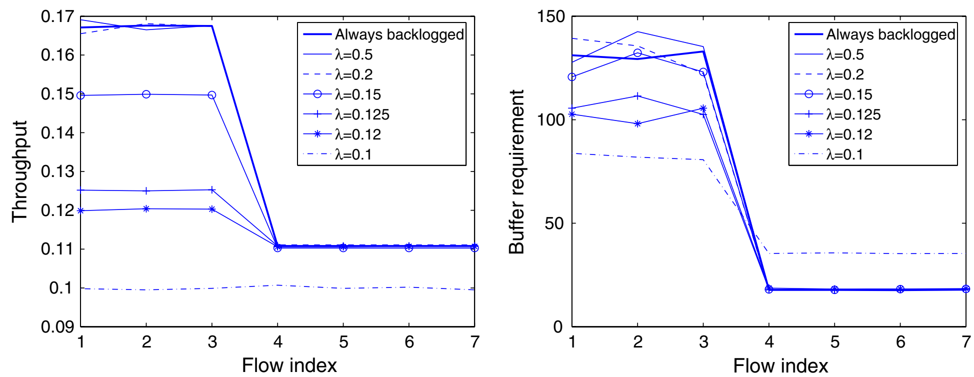


Fig. 11 Comparison of per-flow throughput (left) and buffer requirement (right) of CSD-FA schedulers for $p_c(0) = 0.9$, $\varepsilon = 0.1$ and $\eta = 3$

well for $\lambda \geq 0.2$ for users in $\mathbf{C}^1$ and $\lambda > 0.1$ for users in $\mathbf{C}^2$. As λ is reduced further, the throughput for users in $\mathbf{C}^1$ approaches the packet arrival rate. Eventually, the interval between successive packet arrivals becomes sufficiently large that the channel of users in $\mathbf{C}^1$ becomes memoryless (uncorrelated) and hence, all users will achieve similar throughput levels.

The per-user wireless receiver buffer requirement with the always backlogged assumption approximates the corresponding requirement for Poisson arrivals well for $\lambda \geq 0.15$ for users in $\mathbf{C}^1$ and $\lambda > 0.1$ for users in $\mathbf{C}^2$. As λ is reduced, the buffer requirement is *reduced* (increased) for users in $\mathbf{C}^1$ ($\mathbf{C}^2$).

5 Discussions

5.1 Deduction of channel parameters

The analysis and numerical results presented in Sect. 3 and 4 are based on knowledge of the channel parameters, ($p_c(0)$, ε). In practice, traces are collected from the wireless receivers, from which $p_c(0)$ can be estimated from the average burst length of each channel state and ε can be estimated based on the autocorrelation function of successive measurements. The longer the traces are, the more accurate will be the estimated channel parameters [3].

Since channel characteristics change dynamically, the channel parameters have to be 'refreshed' to achieve the performance gain offered by the CSD-FA scheduler. Since the scheduler will be operating on *stale* channel parameters during the refresh interval, a shorter refresh interval is desirable; however, a shorter trace would result in poorer accuracy of the estimated channel parameters.

5.2 Other heterogeneous scenarios

For the wireless scheduling problem considered in this paper, *heterogeneity* can manifest itself in many forms. For tractable analysis, we considered a specific channel-heterogeneous scenario specified by Eq. 1 and outlined situations for which such a scenario applies. Factors such as QoS requirements and nature of traffic (e.g., voice versus data) introduce user-heterogeneity to the problem. One way of incorporating this in the analysis is through the weight vector, $\underline{r}$ (see Sect. 2.1).

6 Conclusions

In this paper, we consider the scheduling problem where data packets from K input-flows need to be delivered to K corresponding wireless receivers via a heterogeneous wireless channel. Our objective is to design a wireless scheduler that achieves good throughput performance while minimizing the buffer requirement at each wireless receiver.

We propose an adaptive channel-state dependent (CSD-FA) scheduler that first partitions the flows according to their long-term error behavior (persistent/uncorrelated) such that flows with uncorrelated channels are fairly aggregated. The aggregated flow is then scheduled alongside the remaining flows with a channel-state dependent (CSD) scheduler, that utilizes the instantaneous channel state to maximize channel efficiency.

Numerical results suggest that the CSD-FA scheduler achieves similar throughput levels, but has lower buffer requirements at the expense of worse throughput-fairness, compared to a non-adaptive CSD scheduler. By imposing transmission restrictions, the performance of the CSD-FA scheduler in terms of buffer requirement and throughput-fairness can be improved at the expense of reduced throughput.

While our current analysis assumes a simplistic WRR scheduler for the SAP, we study the performance of various loop schedulers in terms of its delay variation in [24]. Our analysis indicates that the WRR scheduler exhibits the worst-case performance over the entire class of loop schedulers. Hence, the performance of the CSD-FA scheduler can be enhanced by considering other loop schedulers for the SAP. Several arbitration schemes are proposed in [7] which may result in performance enhancement over uniform arbitration, which is assumed in the current study.

7 Appendix: proof of Theorem 1

We begin with the derivation of the expressions for per-flow throughput and fairness in terms of $(p_{\mathcal{D}}, p_{\mathcal{S}^1|0}$ and $p_{\mathcal{S}^{\eta^+}|0})$. Let $T^{j|x_i}$ denote the throughput of flow j in slot i given x_i. From [7], the per-flow throughput of a K-flow CSD scheduler with uniform arbitration is given as follows:

$$T^{j|a_i} = \begin{cases} p_{\mathcal{S}|0}, & a_i = j; \\ \dfrac{p_{\mathcal{S}|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{K-1})}{(K-1)(1-p_{\mathcal{D}})}, & \text{otherwise,} \end{cases}$$

where $p_{\mathcal{S}|m} = \text{Prob}$(a flow transmits successfully given that m other eligible flows exist). Applying the above expression for $T^{j|a_i}$ in our (η+1)-flow CSD scheduling scenario, we obtain the following:

$$T^{j \in \mathbf{C}^1|a_i} = \begin{cases} p_{\mathcal{S}^1|0}, & a_i = j; \\ \dfrac{p_{\mathcal{S}^1|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{\eta})}{\eta(1-p_{\mathcal{D}})}, & \text{otherwise,} \end{cases}$$

$$T^{\eta^+|a_i} = \begin{cases} p_{\mathcal{S}^{\eta^+}|0}, & a_i = \eta^+; \\ \dfrac{p_{\mathcal{S}^{\eta^+}|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{\eta})}{\eta(1-p_{\mathcal{D}})}, & \text{otherwise.} \end{cases}$$

According to Eq. 6, for any i, we have the following:

$$Prob(a_i = j) = \begin{cases} \frac{1}{K}, & j \in \mathbf{C}^1; \\ \frac{K-\eta}{K}, & j = \eta^+. \end{cases}$$

Hence, unconditioning the expressions for $T^{j|a_i}$ on a_i, we obtain the following expressions:

$$T^{j \in \mathbf{C}^1} = \frac{1}{K} p_{\mathcal{S}^1|0} + \frac{K-1}{K} \frac{p_{\mathcal{S}^1|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{\eta})}{(\eta)(1-p_{\mathcal{D}})},$$

$$T^{\eta^+} = \frac{K-\eta}{K} p_{\mathcal{S}^{\eta^+}|0} + \frac{\eta}{K} \frac{p_{\mathcal{S}^{\eta^+}|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{\eta})}{(\eta)(1-p_{\mathcal{D}})}.$$

Let us consider the aggregate flow, η^+, which comprises packets of flows $\eta + 1$, $\eta + 2$,...,K. Since the weight of each flow is identical, the probability that a flow j packet is HOL at any instant is identical and given by $\frac{1}{K-\eta}$ for $\eta + 1 \leq j \leq K$.

Hence, for $j \in \mathbf{C}^2$, we obtain the following:

$$T^j = \frac{1}{K-\eta} T^{\eta^+} = \frac{p_{\mathcal{S}^{\eta^+}|0}}{K} + \frac{p_{\mathcal{S}^{\eta^+}|0} \cdot p_{\mathcal{D}}(1 - p_{\mathcal{D}}^{\eta})}{K(K-\eta)(1-p_{\mathcal{D}})}.$$

Substituting the expressions for T^j into Eq. 3, we obtain the expression for FM as given in Theorem 1.

Next, we derive the expressions for $p_{\mathcal{D}}$, $p_{\mathcal{S}^1|0}$ and $p_{\mathcal{S}^{\eta^+}|0}$ in terms of $(p_c(0), \varepsilon)$.

According to our transmission algorithm, a flow j will defer its transmission in slot i only if it is not *eligible* for transmission, i.e., when $\hat{c}_i^j = 1$. The corresponding probability is given as follows:

$$p_{\mathcal{D}} = \sum_{x=0}^{1} \text{Prob}(\hat{c}_i^j = 1 | c_{i-1}^j = x) \cdot \text{Prob}(c_{i-1}^j = x) = 1 - p_c(0).$$

We note that $p_{\mathcal{D}}$ is independent of the channel agility, g, and hence, it is the same for all flows.

In the absence of other eligible flows, a flow j will transmit successfully in slot i as long as $\hat{c}_i^j = c_i^j = 0$. Therefore, we can evaluate $p_{\mathcal{S}^1|0}$ as shown below, where $p_{x,y}(X,Y) \equiv \text{Prob}(x = X, y = Y)$ and $p_{x|y}(X|Y) \equiv \text{Prob}(x = X | y = Y)$:

$$p_{\mathcal{S}^1|0} = \sum_{x=0}^{1} p_{c_i^j, \hat{c}_i^j | c_{i-1}^j}(0,0|x) \cdot p_{c_{i-1}^j}(x) = \sum_{x=0}^{1} p_{c_i^j|c_{i-1}^j}(0|x) \cdot p_{\hat{c}_i^j|c_{i-1}^j}(0|x) = p_c(0) p_{0|0}.$$

Substituting for $p_{0|0}$ in terms of $(p_c(0), \varepsilon)$, we obtain the expressions as given in Theorem 1. The corresponding expression for flow η^+ is obtained by replacing ε with 1.0.

References

1. Bhagwat, P., Bhattacharya, P., Krishna, A., & Tripathi, S. (1996). Enhancing throughput over wireless LANs using channel state dependent packet scheduling. *Proceedings of the IEEE INFOCOM, 3*, 1133–1140.
2. Liu, X., Chan, E. K. P., & Shroff, N. B. (2001). Opportunistic transmission scheduling with resource-sharing constraints in wireless networks. *IEEE Journal on Selected Areas in Communications, 19*(10), 2053–2064.
3. Konrad, A., Zhao, B., Joseph, A., & Ludwig, R. (2001). A Markov-based channel model algorithm for wireless networks. *Proceedings of the ACM MSWiM, July*, 28–36.
4. Willig, A. (2002). A new class of packet- and bit-level models for wireless channels. *Proceedings of the IEEE PIMRC, September*, 2434–2440.
5. Yee, J. R., & Weldon, E. J. (1995). Evaluation of the performance of error-correcting codes on a Gilbert channel. *IEEE Transactions on Communications, 43*(8), 2316–2323.
6. Golestani, S. J. (1994). A self-clocked fair queueing scheme for broadband applications. *Proceedings of the IEEE INFOCOM, June*, 636–646.
7. Rom, R., & Tan, H. P. (2004). Framework for performance analysis of channel-aware wireless schedulers. *Proceedings of the IEEE IPCCC, April*, 153–158.

8. Demers, A., Keshev, S., & Shenker, S. (1990). Analysis and simulation of a fair queueing algorithm. *Journal of Internetworking: Research and Experiance, 1*, 3–26.
9. Parekh, A. K., & Gallager, R. G. (1993). A generalized processor sharing approach to flow control in integrated services networks—the single node case. *IEEE/ACM Transanctions on Networking, 1*(3), 344–357.
10. Sariowan, H., & Cruz, R. L. (1999). SCED: A generalized scheduling policy for guaranteeing quality-of-service. *IEEE/ACM Transanctions on Networking, 7*(5), 669–684.
11. Shakkottai, S., & Stolyar, A. L. (2002). Scheduling for multiple flows sharing a time-varying channel: The exponential rule. *American Mathematical Society Translations Series 2, 207*, 185–202.
12. Shakkottai, S., & Stolyar, A. L. (2001). Scheduling algorithms for a mixture of real and non-real time data in HDR. *Proceedings of the 17th International TeleTraffic Congress, September*.
13. Cao, Y., & Li, V. (2001). Scheduling algorithms in broadband wireless networks. *Proceedings of the IEEE (Invited Paper), 89*(1), 76–87.
14. Lu, S., Bharghavan, V., & Srikant, R. (1997). Fair scheduling in wireless packet networks. *Proceedings of the ACM SIGCOMM*, 63–74.
15. Lu, S., Nandagopal, T., & Bharghavan, V. (1998). A wireless fair service algorithm for packet cellular networks. *Proceedings of the ACM MOBICOM, October*, 10–20.
16. Ng, T. S., Stoica, I., & Zhang, H. (1998). Packet fair queuing algorithms for wireless networks with location-dependent errors. *Proceedings of the IEEE INFOCOM, 3*, 1103–1111.
17. Eckhardt, D., & Steenkiste, P. (2000). Effort-limited fair (ELF) scheduling for wireless networks. *Proceedings of the IEEE INFOCOM, 3*, 1097–1106.
18. Ramanathan, P., & Agrawal, P. (1998). Adapting packet fair queueing algorithms to wireless networks. *Proceedings of the ACM MOBICOM, October*, 1–9.
19. Fragouli, C., Srivastava, M., & Sivaraman, V. (1998). Controlled multimedia wireless link sharing via enhanced class-based queuing with channel-state-dependent packet scheduling. *Proceedings of the IEEE INFOCOM, 2*, 572–580.
20. Nandagopal, T., Lu S., & Bharghavan, V. (1999). A unified architecture for the design and evaluation of wireless fair queuing algorithms. *Proceedings of the ACM MOBICOM, October*, 132–142.
21. Rom, R., & Tan, H. P. (2003). Performance tradeoffs in wireless scheduling with flow aggregation. *Proceedings of the IEEE WCNC, 3*, 1633–1638.
22. Bertsekas, D., & Gallager, R. (1992). *Data Networks*. Englewood Cliffs: Prentice-Hall.
23. Rom, R., & Tan, H. P. (2004). Analysis of trade-offs between buffer and qos requirements in wireless networks. In *Proceedings of the 16th ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems, August*, (pp. 65–76).
24. Rom, R., Sidi, M., & Tan, H. P. (2006). Design and analysis of a class-aware recursive loop scheduler for class-based scheduling. *Elsevier Journal of Performance Evaluation, 63*(9–10), 839–863.

Author Biographies

Raphael Rom received the B.Sc. and M.Sc. degrees in Electrical Engineering from the Technion–Israel Institute of Technology, Haifa, Israel, and the Ph.D. degree in Computer Science from the University of Utah, Salt Lake City, UT. He was a Senior researcher on the research staff of Stanford Research Institute in California, and subsequently joined the Faculty of Electrical Engineering in the Technion, Haifa, Israel. Since 1989 he is also associated with Sun Microsystems where he lead and managed the high speed networking group of SunLabs and is engaged in modeling and analysis of communication networks. In addition he held visiting positions in IBM T.J. Watson Research Center and Stanford University and served as the Chairman of the Israeli National Research Network (IUCC) Dr. Rom is the coauthor of the book "Multiple Access Protocols: Performance and Analysis." His areas of interest are algorithms for, and performance analysis of data communication and wireless networks and the design of general data communication systems.

Hwee-Pink Tan obtained his Ph.D. from the Technion, Israel Institute of Technology in August 2004. From December 2004 to June 2006, he was a post-doctoral research fellow at EURANDOM, Eindhoven University of Technology, The Netherlands, and since July 2006, he has been a post-doctoral research fellow at the Centre for Telecommunications Value-chain Research, Trinity College Dublin, Ireland. He was awarded the A-STAR International Fellowship for his post-doctoral research from December 2004 to December 2006. His research interests are in the performance analysis of wireless networks, particularly at the link/MAC layer. He is currently involved in developing and analyzing communication etiquettes for spectrum access in cognitive radio networks and underwater acoustic networks.