High-Order Momentum: Improving Latency and Convergence for Wireless Network Optimization

Jia Liu
Department of Computer Science
Iowa State University

***Abstract*—In recent years, the rapid growth of mobile data demands has introduced many stringent requirements on latency and convergence performance in wireless network optimization. To address these challenges, several momentum-based algorithms have been proposed to improve the classical queue-length-based algorithmic framework (QLA). By combining queue-length updates and one-slot weight changes (known as the first-order momentum), it has been shown that these algorithms dramatically improve delay and convergence compared to QLA, while maintaining the same throughput-optimality and low-complexity. These exciting attempts have sparked a lot of conjectures about whether it is useful to further exploit *high-order* momentum information to improve delay and convergence speed. In this paper, we show that the answer is *yes*. Specifically, we first propose a new weight updating scheme that enables the incorporation of high-order momentum. We then prove the throughput-optimality and queue-stability of the proposed high-order momentum-based approach and characterize its delay and convergence performances. Through these analytical results, we finally show that delay and convergence would continue to improve as more high-order momentum information is utilized.**

I. Introduction

With the proliferation of smart mobile devices (e.g., smart phones, robotic swarms, autonomous vehicles, etc.), today's wireless network infrastructures are being stretched to their limits by the massive amount of mobile data. Not only does the explosive growth of new mobile data call for an ever-increasing network capacity, but they also introduce stringent latency and convergence speed requirements in real-time network control and optimization. To design highly efficient optimization algorithms to cope with the emerging "big mobile data," a key aspect is to *efficiently* deal with the cross-interactions between congestion control at upper layers and link scheduling at lower layers, both within stack and across users. As a result, recent years have witnessed a compelling need for low-latency and fast-converging joint congestion control and scheduling algorithms.

To date, while there exists a rich literature on joint congestion control and scheduling optimization (see, e.g., [1]–[3]; and see [4] for a survey), most of these schemes are based on the queue-length-based algorithmic framework proposed by Tassiulas and Ephremides more than two decades ago [5]. The enduring popularity of the QLA framework is primarily due to its: i) low-complexity, ii) elegant cross-layer interpretation, and iii) provable throughput-optimality. However, the QLA optimization framework is increasingly unsuitable for the emerging mobile data applications. Specifically, under the standard QLA framework, it is well known that an $O(\epsilon)$ throughput-optimality gap incurs a rapidly increasing $O(1/\epsilon)$ penalty in queueing delay (see, e.g., [3], [6]). To fix this unsatisfactory delay performance, several virtual-queue-based techniques (see, e.g., [2], [7]–[11]). However, these works sacrifice extra throughput for delay reduction, which is reflected either in reduced service rates [2], [7]–[9] or packet dropping [10], [11]. Also, from optimization theory perspective, the QLA framework can be viewed as a first-order stochastic gradient descent algorithm in the Lagrangian dual domain (with queue-lengths serving as dual variables). This gradient descent nature leads to a slow convergence [12]. To address the convergence problem, several second-order approaches (see, e.g., [13], [14]) have been proposed, where the Hessian information is leveraged to increase convergence speed. However, the per-iteration Hessian matrix inversion in these second-order approaches necessitates global information exchange, which does not work well in large-scale networks.

The limitations of first- and second-order approaches have recently motivated researchers to consider a new class of *momentum-based* approaches, most notably the Heavy-Ball method [15], [16]. The basic idea of the momentum-based approaches is to compute the search direction by appropriately combining the current gradient and the previous search direction (known as "momentum") in algorithmic design. Compared to QLA, the momentum-based approaches dramatically reduce the queueing delay from $O(1/\epsilon)$ to $O(1/\sqrt{\epsilon})$ and converge two orders of magnitude faster, while maintaining the same $O(\epsilon)$ utility-optimality gap [15]. Moreover, unlike the spatial domain information exchanges required by the second-order approaches [13], [14], these momentum-based approaches only need to "backtrack" historical information in the *temporal* domain at each node locally, thus preserving the low-complexity of the standard QLA framework.

Thanks to these exciting initial attempts on momentum-based network optimization, there have been rapidly growing interests in the networking research community for a deeper understanding on the roles of momentum information. In particular, since existing works [15], [16] only exploit *first-order* momentum (i.e., only the weight change from the previous time-slot), the following fundamental question naturally arises:

• *Will delay and convergence speed continue to improve as more high-order momentum information being utilized?*

This work has been supported in part by NSF grants CNS-1758757, CCF-1758736, ECCS-1731649, CNS-1446582; and ONR grant N00014-17-1-2417.

We note that answering the above question is highly non-trivial due to several major technical challenges: i) Since the weights and high-order momentum all reside in the Lagrangian dual domain [15], it is unclear whether some strong convexity properties of the dual objective function will continue to hold when more high-order momentum information is utilized; ii) The incorporation of high-order momentum from past iterations implies multi-step dependence in the system, which significantly complicates its performance analysis; iii) As will be seen later, at the heart of the queue-stability and convergence analysis of high-order momentum-based algorithms is a *high-degree time-varying* linear system, for which the eigenvalue characteristic polynomial is notoriously hard to analyze. Indeed, the fundamental Abel-Ruffini theorem says that there is *no* algebraic solution to general polynomial equations of degree higher than four [17]. Hence, it is unclear how to analyze a momentum-based network optimization algorithm with fifth-order momentum and above.

The main contribution of this paper is that, for the first time, we develop a high-order momentum-based wireless network optimization framework to overcome the aforementioned challenges. Our work unveils the roles and characterizes the impacts of high-order momentum in delay and convergence improvements. The main technical results are as follows:

- We propose a new weight updating scheme to incorporate high-order momentum information for joint congestion control and scheduling optimization. We establish a connection between high-order momentum information and the observable queue-lengths and channel states, which enables low-complexity implementations in practice. Further, our algorithm generalizes the existing first-order momentum-based approaches [15], [16] to the high-order momentum regime, advancing the state-of-the-art of the momentum-based methods for wireless network optimization.
- By leveraging the Roché theorem [18], we show that our proposed algorithm with K-order momentum achieves a queueing delay that is $(1-\sum_{i=1}^{K}\beta_i)$–fraction of that of the QLA approach, where $\beta_i \geq 0$, $i=1,\ldots,K$ are momentum coefficients satisfying $\sum_{i=1}^{K}\beta_i<1$. Moreover, our theoretical analysis reveals that a throughput-optimality gap ϵ can be achieved with an $O((1-\sum_{i=1}^{K}\beta_i)/\epsilon)+O((1+\sum_{i=1}^{K}\beta_i)/\sqrt{\epsilon})$ cost in queueing-delay. Further, if the total momentum $\sum_n \beta_n \uparrow 1$ with a speed at or faster than $1-O(\sqrt{\epsilon})$, our algorithm achieves an $[O(\epsilon),O(1/\sqrt{\epsilon})]$ throughput-delay trade-off, which is significantly better than the $[O(\epsilon),O(1/\epsilon)]$ trade-off scaling of the QLA methods.
- With throughput-optimality gap ϵ and a K-order momentum coefficient vector $\boldsymbol{\beta}^{(K)}=\left[\beta_1^{(K)},\ldots,\beta_K^{(K)}\right]^\top$, we show that one can always construct a $(K+1)$-order momentum scheme with parameters $(\epsilon,\boldsymbol{\beta}^{(K+1)})$ such that this $(K+1)$-order scheme converges no slower than the given K-order scheme. This result combined with the previous bullet imply a key insight: Delay and convergence will continue to improve as more high-order momentum information is utilized. We note that this knowledge has *not* yet been discovered in the network optimization literature.

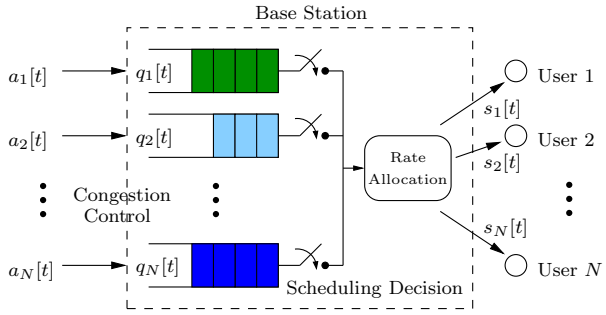

Fig. 1. A wireless cellular downlink network.

Our results in this paper collectively contribute to a comprehensive and fundamental understanding of the roles of momentum information in wireless network optimization. The remainder of this paper is organized as follows. Section II introduces the network model and problem formulation. Section III presents our proposed high-order momentum-based optimization algorithm and its performance analysis. Section IV presents numerical results and Section V concludes this paper.

II. Network Model and Problem Formulation

In this paper, we use boldface to denote matrices/vectors. We let $\mathbf{A}^\top$ denote the transpose of $\mathbf{A}$. We let $\mathbf{I}$ and $\mathbf{O}$ denote the identity and all-zero matrices, respectively, where their dimensions are conformal to the context. Also, we let $\mathbf{1}$ and $\mathbf{0}$ denote the all-one and all-zero vectors, respectively, where their dimensions are conformal to the context. We use $\|\cdot\|$ and $\|\cdot\|_1$ to denote L^2- and L^1-norms, respectively.

Network model: Consider an N-user time-slotted wireless cellular network as shown in Fig. 1, where time is indexed by $t\in\{0,1,2,\ldots\}$. To model channel fading, we use a matrix $\boldsymbol{\Pi}=[\boldsymbol{\pi}_1,\ldots,\boldsymbol{\pi}_M]\in\mathbb{R}^{N\times M}$ to represent the channel state information (CSI) between the base station and the users, where M is the total number of channel states. Each column vector $\boldsymbol{\pi}_m\in\mathbb{R}^N$ represent channel qualities of the N users under state $m\in\{1,\ldots,M\}$. Each channel state $\boldsymbol{\pi}_m$ corresponds to an achievable rate region $\mathcal{C}_m\triangleq Conv\{x_1^{(m)},\ldots,x_N^{(m)}\}$, where $Conv\{\cdot\}$ represents the convex hull operation and $x_n^{(m)}$ denotes a feasible service rate under state m that can be scheduled for user n. We assume that there exists a constant $s^{\max}>0$ such that $x_n^{(m)}\leq s^{\max}$, $\forall n,m$. For convenience, we use $\mathbf{x}^{(m)}=[x_1^{(m)},\ldots,x_N^{(m)}]^\top\in\mathbb{R}^N$ to group the service rates. We assume that the CSI process is i.i.d. across users and time-slots[1]. We let $\boldsymbol{\pi}[t]$ denote the CSI vector in time-slot t and let $p_m\triangleq\Pr\{\boldsymbol{\pi}[t]=\boldsymbol{\pi}_m\}$ denote the probability that the CSI process is in state m. We let $\bar{\mathcal{C}}\triangleq\left\{\mathbf{x}\middle|\mathbf{x}=\sum_{m=1}^{M}p_m\mathbf{x}^{(m)},\ \forall\mathbf{x}^{(m)}\in\mathcal{C}_m\right\}$ represent the mean achievable rate region. In this paper, we assume that the CSI statistics and $\bar{\mathcal{C}}$ are unknown to the base station.

Queueing dynamics: In each time-slot t, based on the current CSI observation $\boldsymbol{\pi}[t]\in\boldsymbol{\Pi}$, the scheduler chooses a service

[1]It is not difficult to generalize our results to Markovian CSI processes following similar arguments in [10], [19].

rate vector $\mathbf{s}[t] \triangleq [s_1[t], \ldots, s_N[t]]^\top \in \mathcal{C}_{\boldsymbol{\pi}[t]}$ and a congestion controlled rate vector $\mathbf{a}[t] \triangleq [a_1[t], \ldots, a_N[t]]^\top \in \mathbb{R}_+^N$. We assume that each user n is associated with a queue, whose queue-length in time-slot t is denoted as $q_n[t]$. Then, the queue-length of each user evolves as:

$$q_n[t+1] = (q_n[t] - s_n[t] + a_n[t])^+, \quad \forall n, \tag{1}$$

where $(\cdot)^+ \triangleq \max\{0, \cdot\}$. Let $\mathbf{q}[t] \triangleq [q_1[t], \ldots, q_N[t]]^\top$ be the queue-length vector in time-slot t. In this paper, we adopt the following notion of queue-stability (same as in [3], [6]): a network is said to be *stable* if the steady-state total queue-length is finite, i.e., $\limsup_{t\to\infty} \mathbb{E}\{\|\mathbf{q}[t]\|_1\} < \infty$.

Problem formulation: Let $\bar{a}_n \triangleq \lim_{T\to\infty} \frac{1}{T} \sum_{t=0}^{T-1} a_n[t]$ denote the average congestion control rate for user n. Each user n is endowed with a utility function $U_n(\bar{a}_n)$, which represents the benefits achieved by user n when its data is injected at rate $\bar{a}_n$. Each $U_n(\cdot)$, $\forall n$, is assumed to be strictly concave, increasing, and twice continuously differentiable. Also, each $U_n(\cdot)$ is assumed to be strongly concave, i.e., there exist two constants $0 < \phi \leq \Phi < \infty$ such that

$$\phi \leq -U_n''(a_n) \leq \Phi, \quad \forall n, \forall a_n \in [0, a^{\max}], \tag{2}$$

where $a^{\max}$ is the upper bound of arrival rates for burst control. For example, the well-known proportional fairness function $\log(\delta + a_n)$ with some arbitrarily small constant $\delta > 0$ satisfies (2). Our goal is to maximize $\sum_{n=1}^N U_n(\bar{a}_n)$, subject to the achievable rate region $\mathcal{C}_{\boldsymbol{\pi}[t]}$ in each time-slot and the queue-stability requirements. Putting together the models presented above, we have the following joint congestion control and scheduling (CCS) optimization problem:

$$\textbf{CCS: Max} \quad \sum_{n=1}^N U_n(\bar{a}_n)$$

$$\text{s.t. Queue-stabiltity}, s_n[t] \in \mathcal{C}_{\boldsymbol{\pi}[t]}, a_n[t] \in [0, a^{\max}], \ \forall n, t.$$

III. A Network Utility Optimization Framework Utilizing High-Order Momentum

In this section, we will first present a network utility optimization algorithmic framework utilizing high-order momentum information to solve Problem CCS in Section III-A. Then, we will summarize the main theoretical results in Section III-B, which is followed by further discussions in Section III-C on the key insights and intuition of the main theorems. Lastly, we will provide detailed performance analysis and proofs for the main theorems in Section III-D.

A. The High-Order Momentum Algorithmic Framework

Our proposed network utility optimization framework with high-order momentum is described in Algorithm 1:

Algorithm 1: A Network Utility Optimization Algorithmic Framework with High-Order Momentum Information.

Initialization:

1. Choose a throughput-optimality gap parameter $\epsilon > 0$ and a momentum coefficient vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_K]^\top \in \mathbb{R}^K$ with $\beta_k > 0$, $\forall k$, satisfying $\sum_{k=1}^K \beta_k < 1$. Set $t = 0$.
2. Let all queues be empty at the initial state: $q_n[0] = 0$, $\forall n$.
3. Associate each link n with a non-negative weight w_n and set $w_n[0] = w_n[-1] = \cdots w_n[-K] = 0$, $\forall n$.

Main Loop:

4. *MaxWeight Scheduler:* In time-slot $t \geq 0$, given the current weight vector $\mathbf{w}[t] \triangleq [w_1[t], \ldots, w_N[t]]^\top$ and the current channel state $\boldsymbol{\pi}[t]$, the scheduler determines the service rate vector $\mathbf{s}[t]$ as follows (breaking ties arbitrarily):
$$\mathbf{s}[t] = \arg\max_{\mathbf{x} \in \mathcal{C}_{\boldsymbol{\pi}[t]}} (\mathbf{w}[t])^\top \mathbf{x}. \tag{3}$$
5. *Congestion Controller:* For each user n, given its current weight $w_n[t]$, the congestion controller determines an integer-valued random data injection rate $a_n[t]$ as follows:
$$\mathbb{E}\{a_n[t] | w_n[t]\} = \min\left\{U_n'^{-1}(\epsilon w_n[t]), a^{\max}\right\}, \tag{4}$$
$$\mathbb{E}\{a_n^2[t] | w_n[t]\} \leq A < \infty, \quad \forall w_n[t], \tag{5}$$
where $U_n'^{-1}(\cdot)$ represents the inverse function of the derivative of $U_n(\cdot)$. In (4) and (5), $a^{\max}$ and A are predefined positive constants depending on the system requirements.
6. *Weight and Queue-Length Updates:* Update the queue-lengths following (1). Let $\Delta q_n[t] \triangleq q_n[t+1] - q_n[t]$ be the resultant queue-length change, $\forall n$. Next, update the weights by combining Δq_n and the weight changes from previous K time-slots (i.e., K-order momentum):
$$w_n[t+1] = \{w_n[t] + \Delta q_n[t] + \beta_1(w_n[t] - w_n[t-1]) + \cdots + \beta_K(w_n[t-K+1] - w_n[t-K])\}^+, \ \forall n. \tag{6}$$
Let $t = t+1$. Go to Step 4 and repeat the scheduling and congestion control processes.

Some comments on Algorithm 1 are in order: First, although the congestion controller in (3) and the scheduler in (4) share the same forms as those in QLA (see, e.g., [3], [6], [19]), the weights are not directly based on queue-lengths. We note that this separation of weights and queue-lengths entails significant delay reductions. Also, it can be seen that the weight update in (6) generalizes the existing momentum-based approach in [15]: It integrates a $\boldsymbol{\beta}$-parameterized *weight change directions* in the previous K time-slots (i.e., K-order momentum). By contrast, the weight updates in [15] are based on first-order momentum in the sense that the new weights only inherit the current queue and the last time-slot weight information. As will be seen later, the incorporation of high-order momentum necessitates new proof techniques to establish the main results in this paper. Lastly, note that when vector $\boldsymbol{\beta} \in \mathbb{R}_+^K$ degenerates to a scalar β (i.e., the momentum order becomes $K = 1$), the high-order momentum scheme reduces back to the heavy-ball algorithm [15]. Thus, the heavy-ball algorithm can be viewed as a special case of our high-order momentum approarch.

B. Main Theoretical Results

The first result in this paper is on the throughput-delay trade-off of our proposed high-order momentum-based algorithm:

Theorem 1 (Delay reduction and queue-stability)**.** *Let $\epsilon > 0$ be a desired throughput-optimality gap. Under momentum*

coefficients $\beta_k \geq 0$, $k = 1, \ldots, K$, *such that* $\sum_{k=1}^{K} \beta_k < 1$, *the steady-state total queue-length obtained by the* K*-order momentum algorithm can be upper-bounded as follows:*

$$\limsup_{t\to\infty} \mathbb{E}\{\|\mathbf{q}[t]\|_1\} = O\left(\left[1 - \sum_{k=1}^{K} \beta_k\right]\frac{1}{\epsilon}\right) + O\left(\left[1 + \sum_{k=1}^{K} \beta_k\right]\frac{1}{\sqrt{\epsilon}}\right). \quad (7)$$

Further, if $\sum_{k=1}^{K} \beta_k \uparrow 1$ *at a speed no slower than* $1 - O(\frac{1}{\sqrt{\epsilon}})$ *as* $\epsilon \downarrow 0$*, Eq. (7) implies that* $\limsup_{t\to\infty} \mathbb{E}\{\|\mathbf{q}[t]\|_1\} = O(\frac{1}{\sqrt{\epsilon}})$.

Several remarks on Theorem 1 are in order: i) The condition $\sum_{k=1}^{K} \beta_k < 1$ entails a profound *mathematical-physical unification*: First, $\sum_{k=1}^{K} \beta_k < 1$ makes physical sense because $(1 - \sum_{k=1}^{K} \beta_k) > 0$ in (7) implies non-negative queue-lengths, i.e., the bound in (7) is always valid. Second, as will be seen in the proof of Theorem 1, $\sum_{k=1}^{K} \beta_k < 1$ is a mathematical consequence of the Rouché theorem [18] to guarantee that the dynamic system in (6) is convergent; ii) If we fix $\boldsymbol{\beta}$ and let $\epsilon \downarrow 0$, the second term on the right-hand-side of (7) is negligible, implying a delay that is $(1 - \sum_{k=1}^{K} \beta_k)$–fraction of that of QLA. Also, if $\sum_{k=1}^{K} \beta_k \uparrow 1$ fast enough as $\epsilon \downarrow 0$, the total queue-length scales as $O(\frac{1}{\sqrt{\epsilon}})$, which grows much *slower* compared to the $O(\frac{1}{\epsilon})$ scaling of the QLA approach. Note that these significant delay gains are obtained *without* losing any extra throughput (unlike those in [10], [11]); iii) The inclusion of K high-order momentum terms in (6) can be interpreted as memorizing the weight information over the past K iterations. Thus, Theorem 1 suggests that "*more memory from distant past*" can further reduce latency in the future.

We now define $U(\mathbf{a}) \triangleq \sum_{n=1}^{N} U_n(a_n)$ as the total utility function of Problem CCS and let $\mathbf{a}^*$ be its optimal solution. We let $a_n^{\infty} \triangleq \mathbb{E}\{\min\{U_n^{'-1}(\epsilon w_n^{\infty}), a^{\max}\}\}$, $\forall n$, be the average steady-state congestion control rates achieved by our K-order momentum algorithm (we will show the existence of steady-state in Section III-D). Further, we let $\mathbf{a}^{\infty} \triangleq [a_1^{\infty}, \ldots, a_N^{\infty}]^{\top}$. Then, our second result states that the K-order momentum algorithm is *throughput-optimal* and the throughput optimality is *not* affected by choice of coefficients $(\beta_1, \ldots, \beta_K)$:

Theorem 2 (Throughput-optimality)**.** *Under Algorithm 1 and for some given* ϵ*, the mean of the stationary rate vector* $\mathbf{a}^{\infty}$ *satisfies* $\|\mathbf{a}^{\infty} - \mathbf{a}^*\| = O(\sqrt{\epsilon})$*. Also, the optimal utility objective value can be bounded as* $|U(\mathbf{a}^{\infty}) - U(\mathbf{a}^*)| = O(\epsilon)$*. Hence,* $\mathbf{a}^{\infty}$ *converges to* $\mathbf{a}^*$ *asymptotically as* $\epsilon \downarrow 0$.

In this paper, we define the notion of convergence in terms of the fewest number of time-slots such that $\{\mathbb{E}\{\mathbf{a}[t]|\mathbf{w}[t]\}\}_{t=0}^{\infty}$ reaches the $O(\sqrt{\epsilon})$-neighborhood of $\mathbf{a}^*$ indicated in Theorem 2. Let $\boldsymbol{\beta}^{(K)} = [\beta_1^{(K)}, \ldots, \beta_K^{(K)}]^{\top}$ denote a K-dimensional momentum vector. Our third result characterizes the convergence performance with respect to K in an *induction* fashion:

Theorem 3 (Convergence speed performance)**.** *Given a* K*-order momentum scheme with parameters* $(\epsilon, \boldsymbol{\beta}^{(K)})$ *that satisfy* $\sum_{k=1}^{K} \beta_k^{(K)} < 1$*. Consider a new* $(K+1)$*-order momentum scheme with parameters* $(\epsilon, \boldsymbol{\beta}^{(K+1)})$ *satisfying either one of the following two conditions: i)* $\sum_{k=1}^{K+1} \beta_k^{(K+1)} = \sum_{k=1}^{K} \beta_k^{(K)}$*; or ii)* $\beta_k^{(K+1)} = \beta_k^{(K)}$*,* $k = 1, \ldots, K$*, and* $\sum_{k=1}^{K+1} \beta_k^{(K+1)} < 1$*. Then, the linear system (cf. the proof of Theorem 1) corresponding to the* $(K+1)$*-order scheme has a smaller spectral radius than that of the* K*-order scheme.*

In plain language, Theorem 3 says that: i) Under the same amount of delay reduction (a consequence of Theorem 1 when $\sum_{k=1}^{K+1} \beta_k^{(K+1)} = \sum_{k=1}^{K} \beta_k^{(K)}$), the scheme with higher order momentum can be made to converge faster; ii) Given a K-order scheme, one can improve *both* delay (due to Theorem 1) and convergence by further adding *one more* momentum term. The proofs of Theorems 1–3 will be provided in Section III-D. Before venturing into the proof details, we would further discuss several key insights and interpretations of the theoretical results in Theorems 1–3.

C. Insights of the Theoretical Results

1) A "deep" structure in temporal domain: Combining the results in Theorems 1–3 reveals the following important insight: *As more higher-order momentum information being utilized, the delay and convergence performance of Algorithm 1 continue to improve, while throughput-optimality is not affected*. This insight bears some interesting similarity to the popular deep-learning architecture in the fields of artificial intelligence: One can interpret the momentum backtracking in temporal domain in Algorithm 1 as exploiting "deeper" layers to optimize wireless networks. More specifically, for each additional piece of momentum information, its momentum coefficient $\beta_k, k = 1, \ldots, K$, can be viewed as providing one more "degree of freedom" that allows us to tweak delay and convergence in wireless network optimization. This interesting observation shows that the use of "deep structures" is powerful and beneficial in wireless network optimization.

2) The intuition behind delay reduction: Before rigorously proving Theorem 1, we provide some high-level intuition as to why the high-order momentum approach could induce a large delay reduction. As mentioned in Section I, the QLA approach can be interpreted as using queue-lengths as dual variables to solve Problem CCS in the Lagrangian dual domain (see, e.g., [2], [3], [6]). As a result, a large number of packets have to be accumulated in each queue to maintain a "pressure" equalling to $\frac{w_n^*}{\epsilon}$, where w_n^* denotes the optimal dual variable. This is the reason that a large $O(1/\epsilon)$ queueing delay is incurred as ϵ decreases to approach throughput-optimality. However, this queue-based "dual mimicking" is unnecessary since one has the freedom to construct any desirable quantity to mimic the dual variables. Now, consider the high-order momentum weight updating in (6). A closer look reveals that the momentum terms $\beta_1(w_n[t] - w_n[t-1]) + \cdots + \beta_K(w_n[t-K+1] - w_n[t-K])$ play a similar role as the place-holder bits in [10] in the sense that they lower the required sizes of $\Delta q_n[t]$, $\forall n$, to to reach $\frac{w_n^*}{\epsilon}$. As a result, a relatively small change in $\Delta q_n[t]$ would result in a large weight variation, which allows the system to react aggressively in congestion control and scheduling even with small queues.

3) The intuition of convergence speedup: The convergence speed-up phenomenon with high-order momentum information can also be understood from an optimization theory perspective: As can be seen from (6), the dual update of weights $w_n[t]$, $\forall n, t$ is a linear combination of current queue-length updates and K weight change directions from the past. Roughly speaking, having more memory of the trajectory of the past iterations allows us to keep track of the curvature of the objective function more closely. As a result, we are better informed about the objective function with more momentum information when making decisions on search directions, thus providing a larger potential for faster convergence.

D. Proofs of the Main Theorems

Due to space limitation, in this subsection, we outline the key steps of the proofs in this paper and relegate further proof details to our online technical report [20].

Sketch of the proof of Theorem 1. For better readability, we organize the lengthy proof of Theorem 1 into three key steps:

Step 1): A ϵ-Scaled Deterministic Problem: To prove Theorem 1, it is useful to first consider an ϵ-scaled deterministic version of Problem CCS. In the deterministic problem, the CSI process is fixed at its mean level, i.e., the achievable rate region is $\bar{\mathcal{C}}$. The congestion control and scheduling variables are time-invariant and are denoted respectively as a_n and s_n, $n = 1, \ldots, N$. The ϵ-parameterized deterministic congestion control and scheduling problem (ϵ-DCCS) can be written as:

$$\epsilon\textbf{-DCCS:} \max_{a_n, s_n, \forall n} \left\{ \frac{1}{\epsilon} \sum_{n=1}^{N} U_n(a_n) \;\middle|\; \begin{array}{l} a_n - s_n \leq 0, s_n \in \bar{\mathcal{C}}, \forall n, \\ a_n \in [0, a^{\max}],\ \forall n. \end{array} \right\}.$$

Due to the strict concavity of $U_n(\cdot)$ and linear constraints, Problem ϵ-DCCS is a convex optimization problem and it is not difficult to check that the Slater condition [12] is satisfied. Hence, strong duality holds and an optimal solution to Problem ϵ-DCCS exists and is unique [12]. Next, we associate dual variables $w_n \geq 0$, $\forall n$ with the constraints $a_n - s_n \leq 0$, $\forall n$, to obtain the corresponding Lagrangian as follows:

$$\Theta_\epsilon(\mathbf{w}) \triangleq \max_{a_n, s_n, \forall n} \left\{ \frac{1}{\epsilon} \sum_{n=1}^{N} U_n(a_n) + \sum_{n=1}^{N} w_n(s_n - a_n) \right\}, \quad (8)$$

where $\mathbf{w} \triangleq [w_1, \ldots, w_N]^\top \in \mathbb{R}_+^N$ contains all dual variables. The Lagrangian dual problem of ϵ-DCCS can be written as:

$$\epsilon\textbf{-DCCS-LD:} \min_{\mathbf{w}} \left\{ \Theta_\epsilon(\mathbf{w}) \,\middle|\, \mathbf{w} \in \mathbb{R}_+^N \right\}. \quad (9)$$

Thanks to the strong duality, the optimal objective value of the Lagrangian dual problem is the same as that of Problem ϵ-DCCS. Recall that, due to the strict convexity of the Lagrangian dual problem, its optimal solution is unique. Therefore, let $\mathbf{w}^*$ be the optimal dual solution to Problem ϵ-DCCS-LD. The following result regarding $\mathbf{w}^*$ will be used in our subsequent performance analysis:

Lemma 4 (Inverse proportional scaling of optimal dual solution)**.** *For a given ϵ, the optimal dual solution satisfies $\mathbf{w}^* = (1/\epsilon)\mathbf{w}^*_{(1)}$, where $\mathbf{w}^*_{(1)}$ denotes the optimal solution to Problem ϵ-DCCS-LD with $\epsilon = 1$. That is, $\mathbf{w}^* = O(1/\epsilon)$.*

Proof. Multiplying ϵ on both sides of (8) yields:

$$\epsilon\Theta_\epsilon(\mathbf{w}) = \max_{a_n, s_n, \forall n} \left\{ \sum_{n=1}^{N} U_n(a_n) + \sum_{n=1}^{N} \widehat{w}_n(s_n - a_n) \right\}, \quad (10)$$

where $\widehat{w}_n = \epsilon w_n$. Note that the right hand side (RHS) of (10) is precisely $\Theta_1(\mathbf{w})$, for which the maximizer is $\widehat{\mathbf{w}} = \mathbf{w}^*_{(1)}$. As a result, we have $\Theta_\epsilon(\mathbf{w})$ is maximized at $(1/\epsilon)\mathbf{w}^*_{(1)}$. □

Further, we note that the optimal primal solutions $\mathbf{a}^*$ and $\mathbf{s}^*$ are independent of ϵ because ϵ is merely a scaling factor in the objective function of Problem ϵ-DCCS.

Step 2): Weight Deviation Bound: Our second step to prove Theorem 1 is to show the following weight deviation bound:

Theorem 5 (Mean weight deviation bound)**.** *Given an ϵ in Algorithm 1, there exists a constant C that depends on N, $s^{\max}$, and $a^{\max}$, such that $\mathbb{E}\{\|\mathbf{w}^\infty - \mathbf{w}^*\|\} \leq C/\sqrt{\epsilon}$, where $\mathbf{w}^\infty$ denotes the weights in steady-state.*

Proof. We start by defining several notation as follows. We let $\bar{a}_n[t] \triangleq U_n^{'-1}(\epsilon w_n[t])$ and $\bar{s}_n[t] \triangleq \mathbb{E}\{s_n[t]\}$. Also, let $\Delta\bar{q}_n[t] \triangleq \bar{a}_n[t] - \bar{s}_n[t]$. For convenience, we let $\bar{\mathbf{a}}[t] = [\bar{a}_1[t], \ldots, \bar{a}_N[t]]^\top$, $\bar{\mathbf{s}}[t] \triangleq [\bar{s}_1[t], \ldots, \bar{s}_N[t]]^\top$, and $\Delta\bar{\mathbf{q}}[t] \triangleq [\Delta q_1[t], \ldots, \Delta q_N[t]]^\top$. Moreover, we let $\mathbf{a}^*$ and $\mathbf{s}^*$ denote the optimal solution to Problem ϵ-DCCS. It is easy to see that $\mathbf{a}^* = \mathbf{s}^*$ because, if otherwise, we can keep increasing the objective value by increasing $\mathbf{a}^*$ without violating the constraints, contradicting to the fact that $\mathbf{a}^*$ is optimal.

Next, we rewrite the high-order momentum weight update equation (6) in the following vector form: $\mathbf{w}[t+1] = \{\mathbf{w}[t] + \Delta\mathbf{q}[t] + \sum_{k=1}^{K} \beta_k(\mathbf{w}[t-k+1] - \mathbf{w}[t-k])\}^+$. Subtracting $\mathbf{w}^*$ on both sides and using $\mathbf{a}^* = \mathbf{s}^*$, we can rearrange and rewrite the weight update dynamics as:

$$\begin{aligned} \mathbf{w}[t+1] - \mathbf{w}^* = &(1+\beta_1)(\mathbf{w}[t] - \mathbf{w}^*) + \sum_{k=1}^{K-1} (\beta_{k+1} - \beta_k) \\ &\times (\mathbf{w}[t-k] - \mathbf{w}^*) + (-\beta_K)(\mathbf{w}[t-K] - \mathbf{w}^*) \\ &+ [\Delta\bar{\mathbf{q}}[t] - (\mathbf{a}^* - \mathbf{s}^*)] + (\Delta\mathbf{q}[t] - \Delta\bar{\mathbf{q}}[t]) + \mathbf{u}[t], \quad (11) \end{aligned}$$

where $\mathbf{u}[t]$ represents the non-negative projection term. Note that since the weight updating in (11) depends on $K+1$ consecutive time-slots of memory $\mathbf{w}[t], \ldots, \mathbf{w}[t-K]$ from the past, traditional Markovian techniques used in [2], [10] cannot be directly applied. Our way to overcome this challenge is to define a vector $\mathbf{z}[t] \in \mathbb{R}^{(K+1)N}$ as follows:

$$\mathbf{z}[t] \triangleq \left[(\mathbf{w}[t] - \mathbf{w}^*)^\top, \cdots, (\mathbf{w}[t-K] - \mathbf{w}^*)^\top \right]^\top. \quad (12)$$

Then, it can be verified that the high-order momentum weight update can be rewritten as a *time-varying* linear system:

$$\mathbf{z}[t+1] = \mathbf{\Gamma}(\mathbf{w}[t])\mathbf{z}[t] + \Delta\tilde{\mathbf{q}}[t] + \tilde{\mathbf{u}}[t], \quad (13)$$

where $\Delta\tilde{\mathbf{q}}[t] \triangleq [(\Delta\mathbf{q}[t] - \Delta\bar{\mathbf{q}}[t])^\top, \mathbf{0}^\top, \ldots, \mathbf{0}^\top]^\top \in \mathbb{R}^{(K+1)N}$, $\tilde{\mathbf{u}}[t] \triangleq [\mathbf{u}^\top[t], \mathbf{0}^\top, \ldots, \mathbf{0}^\top]^\top \in \mathbb{R}^{(K+1)N}$, and the time-varying coefficient matrix $\mathbf{\Gamma}(\mathbf{w}[t])$ is dependent on $\mathbf{w}[t]$ and

$$\mathbf{\Gamma}(\mathbf{w}[t]) \triangleq \begin{bmatrix} (1+\beta_1)\mathbf{I}-\epsilon\mathbf{\Psi}(\mathbf{w}[t]) & (\beta_2-\beta_1)\mathbf{I} & \cdots & (\beta_K-\beta_{K-1})\mathbf{I} & -\beta_K\mathbf{I} \\ \mathbf{I} & \mathbf{O} & \cdots & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} & \cdots & \mathbf{O} & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{I} & \mathbf{O} \end{bmatrix} \in \mathbb{R}^{(K+1)N\times(K+1)N}. \tag{14}$$

defined in (14) at the top of this page. In (14), the matrix $\mathbf{\Psi}(\mathbf{w}[t]) \triangleq \text{Diag}\{\psi_1(\mathbf{w}[t]),\ldots,\psi_N(\mathbf{w}[t])\} \in \mathbb{R}^N$ is diagonal and each diagonal entry is defined as:

$$\psi_n(\mathbf{w}[t]) \triangleq \epsilon\left[\frac{g_{\Theta,n}(\hat{w}_n[t]) - g_{\Theta,n}(\hat{w}_n^*)}{\hat{w}_n[t]-\hat{w}_n^*}\right], \tag{15}$$

where $g_{\Theta,n}(\cdot)$ represents the n-th entry of the gradient of the dual function $\Theta_\epsilon(\cdot)$ and $\hat{w}_n[t] \triangleq \epsilon w_n[t]$.

Then, the following result establishes the upper and lower bounds of the quantity $\frac{g_{\Theta,n}(\hat{w}_n[t])-g_{\Theta,n}(\hat{w}_n^*)}{\hat{w}_n[t]-\hat{w}_n^*}$, which represents the curvature of the dual function $\Theta_\epsilon(\mathbf{w})$(we relegate the proof details to [20, Appendix A] due to space limitation):

Lemma 6 (Curvature of the dual)**.** *There exist constants $\psi_{(\epsilon,\boldsymbol{\beta})}, \Psi > 0$ such that $\psi_{(\epsilon,\boldsymbol{\beta})} \leq \frac{g_{\Theta,n}(\hat{w}_n[t])-g_{\Theta,n}(\hat{w}_n^*)}{\hat{w}_n[t]-\hat{w}_n^*} \leq \Psi$, $\forall w_n[t] \geq 0$, $\forall n$, where the constant $\psi_{(\epsilon,\boldsymbol{\beta})}$ depends on ϵ and $\boldsymbol{\beta}$. Moreover, $\psi_{(\epsilon,\boldsymbol{\beta})}$ increases as ϵ decreases.*

Let $\rho(\mathbf{\Gamma}(\mathbf{w}[t]))$ denote the eign-spectral radius of $\mathbf{\Gamma}(\mathbf{w}[t])$. With Lemma 6, we are in a position to show the following key result of $\rho(\mathbf{\Gamma}(\mathbf{w}[t]))$ (see the proof in Appendix A):

Lemma 7 (Eign-spectral radius of $\mathbf{\Gamma}(\cdot)$)**.** *If $\sum_{k=1}^K \beta_k < 1$ and ϵ is sufficiently small, then $\rho(\mathbf{\Gamma}(\mathbf{w}[t])) < 1$, $\forall t$.*

Now, define a non-negative Lyapunov function $V(\cdot)$ as follows: $V(\mathbf{z}[t]) \triangleq \frac{1}{2}\|\mathbf{P}^{\frac{1}{2}}(\mathbf{z}[t])\mathbf{z}[t]\|^2$, where $\mathbf{P}(\mathbf{z}[t])$ is a symmetric and positive semidefinite matrix dependent on $\mathbf{z}[t]$ and defined as follows:

$$\mathbf{P}(\mathbf{z}[t]) \triangleq \frac{1}{Z}\sum_{j=0}^{\infty}\Big(\mathbf{\Gamma}^\top(\mathbf{w}[t-1])\Big)^j\Big(\mathbf{\Gamma}(\mathbf{w}[t-1])\Big)^j, \tag{16}$$

where $Z \geq \|\sum_{j=0}^{\infty}(\mathbf{\Gamma}^\top(\mathbf{w}[t-1]))^j(\mathbf{\Gamma}(\mathbf{w}[t-1]))^j\|$, $\forall t$, is a normalization factor that will be specified shortly. Note that the infinite sum in (16) is well-defined thanks to Lemma 7. Next, we define a matrix $\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}$ such that $\|\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}\| = \max_{\mathbf{w}[t],\forall t}\|\mathbf{\Gamma}(\mathbf{w}[t])\|$. Following Lemma 7, we can choose a small ϵ and $\sum_{k=1}^K \beta_k < 1$ so that $\rho(\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}) < 1$. Then, it follows from [21, pp. 38, Lemma 1] that there exists a constant c such that $\|\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}\|^j \leq c(\rho(\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})})+\epsilon)^j \approx c(\rho(\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}))^j$. As a result, we can set the normalization factor $Z = \frac{c^2}{1-(\rho(\bar{\mathbf{\Gamma}}_{(\epsilon,\boldsymbol{\beta})}))^2}$.

Next, we evaluate the one-step mean Lyapunov drift $\mathbb{E}\big\{\Delta V(\mathbf{z}[t])\big|\mathbf{z}[t]\big\} \triangleq \mathbb{E}\big\{V(\mathbf{z}[t+1]) - V(\mathbf{z}[t])\big|\mathbf{z}[t]\big\}$. After some algebraic derivations and upper-bounding (see [20, Appendix C] for proof details), we arrive at the following result:

Lemma 8. *Let $B \triangleq \frac{N}{2}[A+(s^{\max})^2]$. There exist constants $\delta, \bar{\delta}, \eta > 0$ such that the one-step Lyapunov drift satisfies:*

$$\mathbb{E}\{\Delta V(\mathbf{z}[t])|\mathbf{z}[t]=\mathbf{z}\} \leq -\bar{\delta}\sqrt{\epsilon}\big\|\mathbf{z}\big\|\mathbb{1}_{\mathcal{B}^c_{(\epsilon)}}(\mathbf{z}) + \eta\mathbb{1}_{\mathcal{B}_{(\epsilon)}}(\mathbf{z}), \tag{17}$$

where $\mathcal{B}_{(\epsilon)} \triangleq \{\mathbf{z}: \big\|\mathbf{z}\big\| < \sqrt{B/\epsilon\delta}\}$ and $\mathcal{B}^c_{(\epsilon)}$ is its complement.

Now, consider the T-step conditional mean Lyapunov drift. To this end, we define a set $\Omega \triangleq \{\mathbf{z}: \|\mathbf{z}\| \in \mathcal{B}_\epsilon\}$. By telescoping (17) from $t=0$ to T and using Lemma 8, we can show that (see [20, Section 3.4] for detailed derivations):

$$\mathbb{E}\{V(\mathbf{z}[T])|\mathbf{z}[0]\} - V(\mathbf{z}[0]) = \sum_{t=0}^{T-1}\mathbb{E}\{\Delta V(\mathbf{z}[t])|\mathbf{z}[0]\} \leq -\bar{\delta}\sqrt{\epsilon}$$
$$\times\int_{\Omega^c}\bigg(\|\mathbf{z}[t]\|\sum_{t=0}^{T-1}p_{\mathbf{z}[t]|\mathbf{z}[0]}\bigg)d\mathbf{z} + \eta\int_{\Omega}\bigg(\sum_{t=0}^{T-1}p_{\mathbf{z}[t]|\mathbf{z}[0]}\bigg)d\mathbf{z}. \tag{18}$$

Note that $\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}p_{\mathbf{z}[t]|\mathbf{z}[0]} = p_{\mathbf{z}}^\infty$ for all $\mathbf{z}[0]$, where $p_{\mathbf{z}}^\infty$ denotes the stationary distribution of the continuous state Markov chain $\mathbf{z}[t]$. Moving $V(\mathbf{z}[0])$ to the right hand side of (18), dividing both sides by T, and letting $T\to\infty$ yields $0 \leq -\bar{\delta}\sqrt{\epsilon}\int_{\Omega^c}p_{\mathbf{z}}^\infty\|\mathbf{z}^\infty\|d\mathbf{z} + \eta\int_\Omega p_{\mathbf{z}}^\infty d\mathbf{z}$. Rearranging terms and adding $\bar{\delta}\sqrt{\epsilon}\int_\Omega p_{\mathbf{z}}^\infty\|\mathbf{z}^\infty\|$ to both sides yields:

$$\bar{\delta}\sqrt{\epsilon}\int_{\mathbb{R}^{(K+1)N}}p_{\mathbf{z}}^\infty\|\mathbf{z}^\infty\|d\mathbf{z} \leq \int_\Omega\Big(\eta+\bar{\delta}\sqrt{\epsilon}\|\mathbf{z}^\infty\|\Big)p_{\mathbf{z}}^\infty d\mathbf{z}$$
$$\overset{(a)}{\leq}(\eta+\bar{\delta}\sqrt{B/\delta})\int_\Omega p_{\mathbf{z}}^\infty d\mathbf{z} \leq \eta+\bar{\delta}\sqrt{B/\delta}, \tag{19}$$

where (a) follows from the definitions of Ω and $\mathcal{B}_\epsilon$. Note that the left-hand-side (LHS) of (19) is exactly $\bar{\delta}\sqrt{\epsilon}\mathbb{E}\big\{\|\mathbf{z}^\infty\|\big\}$. Therefore, dividing both sides of (19) by $\bar{\delta}\sqrt{\epsilon}$ yields:

$$\mathbb{E}\big\{\|\mathbf{w}^\infty-\mathbf{w}^*\|\big\} \leq \mathbb{E}\big\{\|\mathbf{z}^\infty\|\big\} \leq \bigg(\frac{\eta}{\bar{\delta}}+\sqrt{B/\delta}\bigg)\frac{1}{\sqrt{\epsilon}}. \tag{20}$$

This completes the proof of Theorem 5. □

3) Closing Steps of the Proof: Coming back to the high-order momentum weight update (in vector form): $\mathbf{w}[t+1] = \mathbf{w}[t] + \Delta\mathbf{q}[t] + \sum_{k=1}^K\beta_k(\mathbf{w}[t-k+1]-\mathbf{w}[t-k]) + \mathbf{u}[t]$. Rearranging and noting the fact that the non-negative orthant projection term $\mathbf{u}[t]$ satisfies $\mathbf{u}[t] \geq \mathbf{0}$, we have:

$$\Delta\mathbf{q}[t] \leq \big(\mathbf{w}[t+1]-\mathbf{w}[t]\big) - \sum_{k=1}^K\beta_k\big(\mathbf{w}[t-k+1]-\mathbf{w}[t-k]\big). \tag{21}$$

Telescoping the inequality in (21) from $t=0$ to $T-1$ yields:

$$\sum_{t=0}^{T-1}\Delta\mathbf{q}[t] \leq \big(\mathbf{w}[T]-\mathbf{w}[0]\big) - \sum_{k=1}^K\beta_k\big(\mathbf{w}[T-k+1]$$
$$-\mathbf{w}[-k]\big) \overset{(a)}{=} \mathbf{w}[T] - \sum_{k=1}^K\beta_k\mathbf{w}[T-k+1], \tag{22}$$

where (a) holds because $\mathbf{w}[0] = \mathbf{w}[-1] = \cdots = \mathbf{w}[-K] = \mathbf{0}$ according to our assumption. Also, since $\mathbf{q}[0] = \mathbf{0}$, we have

$$\|\mathbf{q}[T]\|_1 = \bigg\|\mathbf{q}[0] + \sum_{t=0}^{T-1}\Delta\mathbf{q}[t]\bigg\|_1 \leq \bigg\|\mathbf{w}[T] - \sum_{k=1}^K\beta_k\mathbf{w}[T-k+1]\bigg\|_1.$$

Taking expectation and "limsup" operations as $T \to \infty$ yields:

$$\limsup_{T\to\infty} \mathbb{E}\big\{\|\mathbf{q}[T]\|_1\big\} \le \mathbb{E}\bigg\{\mathbf{w}^\infty - \sum_{k=1}^{K}\beta_k \mathbf{w}^\infty\bigg\} \overset{(a)}{\le} \mathbf{w}^* + O\bigg(\frac{1}{\sqrt{\epsilon}}\bigg)$$
$$-\sum_{k=1}^{K}\beta_k\bigg[\mathbf{w}^* - O\bigg(\frac{1}{\sqrt{\epsilon}}\bigg)\bigg] = \bigg[1-\sum_{k=1}^{K}\beta_k\bigg]\mathbf{w}^* + \bigg[1+\sum_{k=1}^{K}\beta_k\bigg]O\bigg(\frac{1}{\sqrt{\epsilon}}\bigg),$$

where (a) follows from Theorem 5 and $\|\cdot\|_1 \le \sqrt{N}\|\cdot\|$. Also, as $\sum_{k=1}^{K}\beta_k \uparrow 1$ at a speed equal to or faster than $1-O(\sqrt{\epsilon})$, it then follows that $\limsup_{T\to\infty}\mathbb{E}\big\{\|\mathbf{q}[t]\|_1\big\} = O(1/\sqrt{\epsilon})$. □

Sketch of the proof of Theorem 2. We first characterize the optimality gap for $a_n^\infty \triangleq \mathbb{E}\{\min\{U_n^{'-1}(\epsilon w_n^\infty), a^{\max}\}\}$. Note that $a_n^* = U_n^{'-1}(\epsilon w_n^*)$, $\forall n$. Applying these definitions in $\big\|\mathbf{a}^\infty - \mathbf{a}^*\big\|^2$ and using Jensen's inequality and mean value theorem, we obtain (see derivations in [20, Eq. (25)]): $\|\mathbf{a}^\infty - \mathbf{a}^*\|^2 \le \frac{\epsilon^2}{\phi^2}\mathbb{E}\{\|\mathbf{w}^\infty - \mathbf{w}^*\|^2\}$. Next, consider $\mathbb{E}\{\|\mathbf{w}^\infty - \mathbf{w}^*\|^2\}$. From the proof of Lemma 8, we have the following one-slot mean Lyapunov drift bound (see [20, Appendix C] for details):

$$\mathbb{E}\{\Delta V(\mathbf{z}[t])|\mathbf{z}[t]\} \le -\frac{\epsilon}{\Phi}\big\|\mathbf{w}[t] - \mathbf{w}^*\big\|^2 + B. \tag{23}$$

Following the same arguments in the proof of Theorem 5 to telescope (23) from $t=0$ to $T-1$ yields: $\mathbb{E}\{V(\mathbf{z}[T])|\mathbf{z}[0]\} - V(\mathbf{z}[0]) \le -\frac{\epsilon}{\Phi}\sum_{t=0}^{T-1}\int_{\mathbb{R}^{2N}} p_{\mathbf{z}[t]|\mathbf{z}[0]}(\mathbf{z})\big\|\mathbf{w} - \mathbf{w}^*\big\|^2 d\mathbf{z} + TB$. Dividing both sides by $\frac{\epsilon T}{\Phi}$, rearranging terms, and letting $T \to \infty$, we have $\limsup_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\int_{\mathbb{R}^{(K+1)N}} p_{\mathbf{z}[t]|\mathbf{z}[0]}(\mathbf{z})\big\|\mathbf{w} - \mathbf{w}^*\big\|^2 d\mathbf{z} \le B\Phi/\epsilon$. Note that the left-hand-side is exactly $\mathbb{E}\big\{\|\mathbf{w}^\infty - \mathbf{w}^*\|^2\big\}$. Hence, we have

$$\|\mathbf{a}^\infty - \mathbf{a}^*\|^2 \le \frac{\epsilon^2}{\phi^2}\mathbb{E}\big\{\|\mathbf{w}^\infty - \mathbf{w}^*\|^2\big\} \overset{(a)}{\le} \frac{B\Phi}{\phi^2}\epsilon, \tag{24}$$

where (a) utilizes Theorem 5. Taking square root on (24) yields $\|\mathbf{a}^\infty - \mathbf{a}^*\| = O(\sqrt{\epsilon})$, i.e., the first half of Theorem 2.

To show that $|U(\mathbf{a}^\infty) - U(\mathbf{a}^*)| = O(\epsilon)$, we follow similar steps in the proof of Theorem 5 to define a vector $\mathbf{y}[t] \triangleq [\mathbf{w}^\top[t], \ldots, \mathbf{w}^\top[t-K]]^\top$ and a quadratic Lyapunov function $L(\mathbf{y}[t]) = \frac{1}{2}\|\mathbf{P}^{\frac{1}{2}}(\mathbf{y}[t])\mathbf{y}[t]\|^2$. Following the same arguments in the proof of Theorem 5, we can show that $\mathbf{y}[t+1] = \mathbf{\Gamma}(\mathbf{w}[t])\mathbf{y}[t] + \Delta\tilde{\mathbf{q}}[t] + \tilde{\mathbf{u}}[t]$. Then, using the same techniques as in the proof of Lemma 8, we can establish that the one-slot expected Lyapunov drift can be upper-bounded as $\mathbb{E}\big\{\Delta L(\mathbf{y}[t])\big|\mathbf{y}[t]\big\} \le -\mathbf{w}^\top[t]\mathbb{E}\{\mathbf{a}[t] - \mathbf{s}[t]\big|\mathbf{y}[t]\} + B$. Next, we note that the right-hand-side has the same structure as in [3, Eq. (24)]. Therefore, the remaining steps of the proof follow from those in [3] and the proof is complete. □

Sketch of the proof of Theorem 3. Thanks to the one-to-one mapping between $\mathbb{E}\{\mathbf{a}[t]|\mathbf{w}[t]\}$ and $\mathbf{w}[t]$, the convergence of $\{\mathbb{E}\{\mathbf{a}[t]|\mathbf{w}[t]\}\}$ can be equivalently analyzed by examining the sequence $\{\mathbf{w}[t]\}_{t=0}^\infty$. Also, from the proof of Theorem 5, we know that the distance $\|\mathbf{w}[t] - \mathbf{w}^*\|$ can be reformulated as a time-varying linear system $\mathbf{z}[t+1] = \mathbf{\Gamma}(\mathbf{w}[t])\mathbf{z}[t] + \Delta\tilde{\mathbf{q}}[t] + \tilde{\mathbf{u}}[t]$ (cf. Eq. (13)), where $\mathbf{z}[t]$ is defined in (12). As a result, the convergence speed of $\mathbf{w}[t]$ can be analyzed through $\mathbf{\Gamma}(\mathbf{w}[t])$'s eigen-spectral radius. Further, from the proof of Lemma 7 in Appendix A (cf. Eq. (29)), we know that the characteristic polynomial equation of $\mathbf{\Gamma}(\mathbf{w}[t])$ can be written as follows (omitting index "n" for brevity):

$$\lambda^{K+1} - \Big(1 + \beta_1^{(K)} - \epsilon\psi(\mathbf{w}[t])\Big)\lambda^K - \Big(\beta_2^{(K)} - \beta_1^{(K)}\Big)\lambda^{K-1} - \cdots - \Big(\beta_K^{(K)} - \beta_{K-1}^{(K)}\Big)\lambda + \beta_K^{(K)} = 0. \tag{25}$$

Let $|\bar{\lambda}^{(K)}|$ denote the maximum root magnitude of (25). For (25), the Fujiwara theorem [22] says that:

$$\big|\bar{\lambda}^{(K)}\big| \le 2\max\Big\{\big|1 + \beta_1^{(K)} - \epsilon\psi(\mathbf{w}[t])\big|,\ \big|\beta_2^{(K)} - \beta_1^{(K)}\big|^{\frac{1}{2}}, \ldots,\ \big|\beta_K^{(K)} - \beta_{K-1}^{(K)}\big|^{\frac{1}{K}},\ \frac{1}{2}\big|\beta_K^{(K)}\big|^{\frac{1}{K+1}}\Big\}. \tag{26}$$

Therefore, applying the Fujiwara theorem to a $(K+1)$-order momentum scheme, we have that:

$$\big|\bar{\lambda}^{(K+1)}\big| \le 2\max\Big\{\big|1 + \beta_1^{(K+1)} - \epsilon\psi(\mathbf{w}[t])\big|, \big|\beta_2^{(K+1)} - \beta_1^{(K+1)}\big|^{\frac{1}{2}}, \ldots, \big|\beta_{K+1}^{(K+1)} - \beta_K^{(K+1)}\big|^{\frac{1}{K+1}},\ \frac{1}{2}\big|\beta_{K+1}^{(K+1)}\big|^{\frac{1}{K+2}}\Big\}. \tag{27}$$

Now, consider the following choice of $\boldsymbol{\beta}^{(K+1)}$: $\beta_k^{(K+1)} = \beta_k^{(K)}$, $k = 1, \ldots, K-1$, $\beta_K^{(K+1)} = \beta_K^{(K)} - (\beta_K^{(K)})^{\frac{K+2}{K+1}} + \delta$, and $\beta_{K+1}^{(K+1)} = (\beta_K^{(K)})^{\frac{K+2}{K+1}} - \delta$, where $\delta > 0$ is chosen such that $\beta_k^{(K+1)} > 0$, $\forall k$. Clearly, $\sum_{k=1}^{K+1}\beta_k^{(K+1)} = \sum_{k=1}^{K}\beta_k^{(K)}$. It can be verified that if the maximizer is from the first K terms, we have $|\bar{\lambda}^{(K+1)}| = |\bar{\lambda}^{(K)}|$. Otherwise, it is easy to see that $|\bar{\lambda}^{(K+1)}| < |\bar{\lambda}^{(K)}|$. This completes first half of the proof.

Next, if $\beta_k^{(K+1)} = \beta_k^{(K)}$, $k = 1, \ldots, K$, we can choose $\beta_{K+1}^{(K+1)} = \min\{(\beta_K^{(K)})^{\frac{K+2}{K+1}}, (1 - 2^{-(K+1)})\beta_K^{(K)}\}$. It can be seen that if the maximizer is from the first K terms, we have $|\bar{\lambda}^{(K+1)}| = |\bar{\lambda}^{(K)}|$. Otherwise, we have $|\bar{\lambda}^{(K+1)}| < |\bar{\lambda}^{(K)}|$. This completes the second half of the proof. □

IV. Numerical Results

In this section, we perform numerical experiments to validate our theoretical results in Section III. For clearer illustrations and avoid being obscured by channel fading randomness, we first study a three-link deterministic cellular network, where each link has one unit capacity and only one user can be activated in each time-slot. We adopt the proportional fairness metric $\log(0.0001 + a)$ as the utility function for all users [4]. From the symmetry of the system, it is clear that the optimal congestion control rates are $\bar{a}_n^* = \frac{1}{3}$, $n = 1, 2, 3$. In all experiments, we set $\epsilon = 0.04$. We first consider a 2-order momentum scheme with $\beta_1 + \beta_2$ increasing from 0 to 0.3, 0.6, and 0.9 (note that "0" corresponds to the QLA approach). In each case, we let $\beta_2 = \frac{1}{4}\beta_1$ and the results are shown in Fig. 2. We can see that as $\beta_1 + \beta_2$ increases, the average queue-lengths are 74.5, 52.5, 30.1, and 7.5, respectively, confirming the $(1 - \sum_n \beta_n)$–fraction reduction result in Theorem 1. The throughput convergence results are shown in Fig. 3. We can see that, regardless of the value of $\sum_n \beta_n$, the congestion control rates converge to the same optimal point, confirming the throughput-optimality result of Theorem 2. Further, Fig. 3 shows that throughput converges faster as $\sum_n \beta_n$ increases.

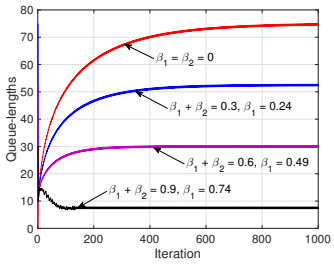

Fig. 2. Queue-lengths of 2-order momentum scheme with various (β_1,β_2).

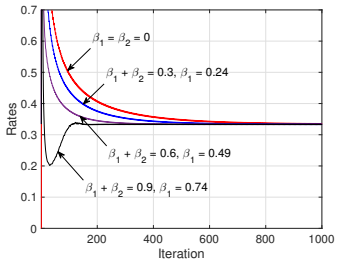

Fig. 3. Convergence of 2-order momentum scheme with various (β_1,β_2).

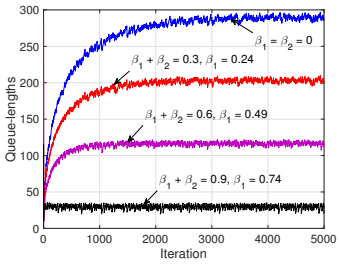

Fig. 4. Queue-lengths of 2-order momentum scheme under channel fading.

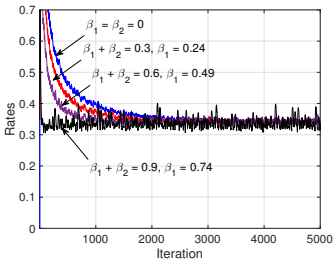

Fig. 5. Convergence of 2-order momentum scheme under channel fading.

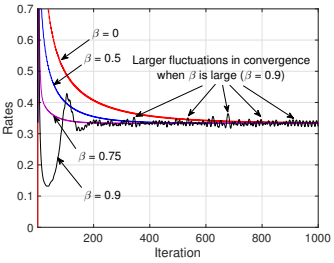

Fig. 6. The throughput convergence of 1-order momentum schemes (heavy-ball) with different β-values.

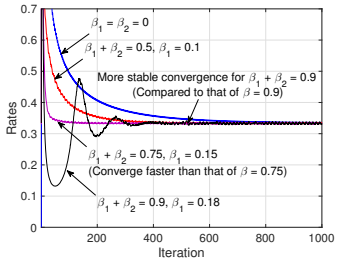

Fig. 7. The throughput convergence of 2-order momentum schemes with the same total momentum as in Fig. 6.

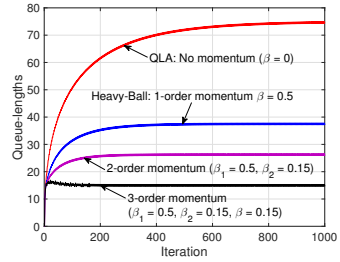

Fig. 8. Queue-lengths comparisons with increasing order of momentum $(\beta_k^{(K+1)} = \beta_k^{(K)}, k = 1, \ldots, K)$.

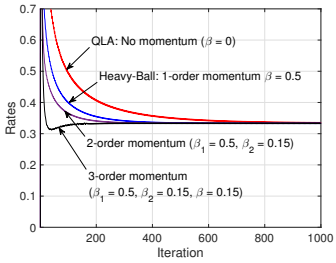

Fig. 9. Convergence comparisons with increasing order of momentum $(\beta_k^{(K+1)} = \beta_k^{(K)}, k = 1, \ldots, K)$.

Next, we test the 2-order momentum scheme in a 15-link cellular downlink with stochastic channel fading and $\epsilon = 0.01$. The variations of (β_1, β_2) are the same as in the deterministic cases. The queue-length evolution and throughput convergence are illustrated in Figs. 4 and 5, respectively. Under channel fading, we can observe the same trends in queue evolutions and throughput convergence, again confirming Theorems 1–2.

Next, we study the impacts of the degree of momentum on convergence performances. In Figs. 6 and 7, we compare 1-order (heavy-ball [15]) and 2-order momentum schemes with $\beta = \beta_1 + \beta_2$. Comparing Figs. 6 and 7, we can see that, as the total amount of momentum increases (implying better delay), the 2-order momentum scheme converges faster. Notably, when $\beta = 0.9$, the 1-order momentum scheme fluctuates more dramatically around the optimal, while the 2-order momentum scheme enjoys a much smoother convergence. This confirms the first half of Theorem 3. Lastly, we studied the effects of increasing the order of momentum in the way that $\beta_k^{(K+1)} = \beta_k^{(K)}$, $k = 1, \ldots, K$, and the results are shown in Figs. 8 and 9. We can see that as the order K increases from 0 to 3, both delay and throughput convergence speed improve, which verifies Theorem 1 and the second half of Theorem 3.

V. Conclusion

In this paper, we have developed a new cross-layer algorithmic framework that enables the use of high-order momentum information to improve delay and convergence in wireless network optimization. Compared to the well-known queue-length-based approaches, our proposed high-order momentum-based algorithmic framework offers not only throughput-optimality, but also fast-convergence and low-delay. The main contributions of this paper are three-fold: i) we have proposed a new low-complexity weight updating scheme to incorporate high-order momentum information, which can be easily implemented in practice; ii) we rigorously established the throughput-optimality of the proposed algorithmic framework and characterized its queue-stability and convergence speed with respect to the order/degree of momentum information; and iii) based on these analytical results, we are able to reveal a fundamental insight that delay and convergence speed will keep on improving as more high-order momentum information is utilized. Collectively, the findings in this paper contribute to a new and exciting research paradigm that leverages high-order momentum information in wireless network optimization. Future research topics may include to generalize the proposed high-order momentum algorithmic framework to handle potentially non-stationary wireless network settings.

Appendix A
Proof of Lemma 7

For notational convenience, we let $\mathbf{\Lambda} \triangleq (1+\beta_1)\mathbf{I} - \epsilon\mathbf{\Psi}(\mathbf{w}[t])$, $\alpha_1 \triangleq \beta_2 - \beta_1$, …, $\alpha_{K-1} \triangleq \beta_K - \beta_{K-1}$, and $\alpha_K \triangleq -\beta_K$. Then, the matrix in (14) can be written as:

$$\mathbf{\Gamma}(\mathbf{w}[t]) = \left[\begin{array}{cccc|c} \mathbf{\Lambda} & \alpha_1\mathbf{I} & \cdots & \alpha_{K-1}\mathbf{I} & \alpha_K\mathbf{I} \\ \hline \mathbf{I} & \mathbf{O} & \cdots & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} & \cdots & \mathbf{O} & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{I} & \mathbf{O} \end{array}\right]. \quad (28)$$

We first show by contradiction that the eigenvalues $\lambda_i \neq 0$, $\forall i = \{1, \ldots, (K+1)N\}$. Suppose otherwise that $\lambda_i = 0$ for some $i \in \{1, \ldots, (K+1)N\}$. In this case, it follows from (28) that $\det(\mathbf{\Gamma}(\mathbf{w}[t]) - 0\mathbf{I}) = \det(\alpha_K)\det(\mathbf{I}) = \alpha_K^{KN} \neq 0$, a

contradiction. Next, consider the determinant:

$$\det\big(\mathbf{\Gamma}(\mathbf{w}[t])-\lambda\mathbf{I}\big)=\det\left[\begin{array}{cccc|c}\mathbf{\Lambda}-\lambda\mathbf{I} & \alpha_1\mathbf{I} & \cdots & \alpha_{K-1}\mathbf{I} & \alpha_K\mathbf{I}\\ \mathbf{I} & -\lambda\mathbf{I} & \cdots & \mathbf{O} & \mathbf{O}\\ \mathbf{O} & \mathbf{I} & \cdots & \mathbf{O} & \mathbf{O}\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ \hline \mathbf{O} & \mathbf{O} & \cdots & \mathbf{I} & -\lambda\mathbf{I}\end{array}\right]$$

$$\overset{(a)}{=}\det(-\lambda\mathbf{I})\det\left[\begin{array}{cccc|c}\mathbf{\Lambda}-\lambda\mathbf{I} & \alpha_1\mathbf{I} & \cdots & \alpha_{K-1}\mathbf{I} & \alpha_{K-1}+\frac{\alpha_K}{\lambda}\\ \mathbf{I} & -\lambda\mathbf{I} & \cdots & \mathbf{O} & \mathbf{O}\\ \mathbf{O} & \mathbf{I} & \cdots & \mathbf{O} & \mathbf{O}\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ \hline \mathbf{O} & \mathbf{O} & \cdots & \mathbf{I} & -\lambda\mathbf{I}\end{array}\right],$$

where (a) follows from $\det\left[\begin{array}{cc}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{array}\right]=\det(\mathbf{D})\det(\mathbf{A}-\mathbf{B}\mathbf{D}^{-1}\mathbf{C})$ when $\mathbf{D}$ is invertible (since $\lambda\neq 0$). Repeating the above steps $K-1$ more times, we eventually arrive at the following characteristic polynomial equation for $\mathbf{\Gamma}(\mathbf{w}[t])$:

$$\begin{aligned}\det\big(\mathbf{\Gamma}(\mathbf{w}[t])-\lambda\mathbf{I}\big)&=\big(\det(-\lambda\mathbf{I})\big)^K\times\\ &\qquad\det\Big[\mathbf{\Lambda}-\lambda\mathbf{I}-\Big(\alpha_1+\frac{\alpha_2}{\lambda}+\cdots+\frac{\alpha_K}{\lambda^K}\Big)\mathbf{I}\Big]\\ &=(-\lambda)^{NK}\det\Big[\Big((1+\beta_1)+\frac{\alpha_1}{\lambda}+\cdots+\frac{\alpha_K}{\lambda^K}-\lambda\Big)\mathbf{I}-\epsilon\mathbf{\Psi}(\mathbf{w}[t])\Big]\\ &=(-1)^{NK}\prod_{n=1}^{N}\Big(\lambda^{K+1}-\big(\alpha_0-\epsilon\psi_n(\mathbf{w}[t])\big)\lambda^K-\cdots-\alpha_K\Big)\\ &=(-1)^{NK}\prod_{n=1}^{N}\Big[\lambda^{K+1}-\big(1+\beta_1-\epsilon\psi_n(\mathbf{w}[t])\big)\lambda^K-\\ &\quad(\beta_2-\beta_1)\lambda^{K-1}-\cdots-(\beta_K-\beta_{K-1})\lambda+\beta_K\Big]=0. \quad (29)\end{aligned}$$

For (29) to hold, it suffices to consider each polynomial equation $\lambda^{K+1}-\big(1+\beta_1-\epsilon\psi_n(\mathbf{w}[t])\big)\lambda^K-(\beta_2-\beta_1)\lambda^{K-1}-\cdots-(\beta_K-\beta_{K-1})\lambda+\beta_K=0$. It is easy to check that $\lambda=1$ is not one of the roots of this polynomial equation. Also, as $\epsilon\downarrow 0$, its roots converge to those of the polynomial equation $\lambda^{K+1}-\big(1+\beta_1\big)\lambda^K-(\beta_2-\beta_1)\lambda^{K-1}-\cdots-(\beta_K-\beta_{K-1})\lambda+\beta_K=0$, which can be factored as:

$$\big(\lambda-1\big)\big(\lambda^K-\beta_1\lambda^{K-1}-\beta_2\lambda^{K-2}-\cdots-\beta_K\big)=0.$$

Note that if $\sum_{k=1}^K\beta_k<1$ and $\beta_k>0,\forall k=1,\ldots,K$, then by the Roché theorem [18], we have $|\lambda|<\max\{1,\sum_{k=1}^K\beta_k\}=1$. Therefore, there exists a small enough value of ϵ such that the magnitude of the roots of (29) is upper bounded by 1. Further, note that $\epsilon\psi_n(\mathbf{w}[t])(\beta_1\lambda^{K-1}+\cdots+\beta_K)\to 0$ as $\epsilon\downarrow 0$. This is because $(\beta_1\lambda^{K-1}+\cdots+\beta_K)$ is upper bounded and independent of ϵ and $\psi_n(\mathbf{w}[t])$ is upper bounded by Lemma 6. Therefore, adding $-\epsilon\psi_n(\mathbf{w}[t])(\beta_1\lambda^{K-1}+\cdots+\beta_K)\to 0$ to both sides of (29) and after some algebraic manipulations, we have that, as $\epsilon\downarrow 0$,

$$\big[\lambda-\big(1-\epsilon\psi_n(\mathbf{w}[t])\big)\big]\big(\lambda^K-\beta_1\lambda^{K-1}-\cdots-\beta_K\big)\approx 0.$$

Therefore, when ϵ is small, we can conclude that the roots of (29) are such that $\lambda_{1,n}\approx 1-\epsilon\psi_n(\mathbf{w}[t])$, and $\lambda_{2,n},\ldots,\lambda_{K+1,n}$ are close to those of $\lambda^K-\beta_1\lambda^{K-1}-\cdots-\beta_K=0$. From the above analysis, we can conclude that $|\lambda_{k,n}|<1$, $\forall k=1,\ldots,K+1$. Also, $\lambda_{k,n}$, $\forall k$, are asymptotically independent of ϵ. Note that $\lambda_{1,n}\uparrow 1$ as $\epsilon\downarrow 0$. Therefore, for small enough ϵ, we have that $\max_k|\lambda_{k,n}|=|\lambda_{1,n}|=1-\epsilon\psi_n(\mathbf{w}[t])$. Finally, we have $\rho(\mathbf{\Gamma}(\mathbf{w}[t]))\approx\max_n|\lambda_{1,n}|=1-\epsilon\min_n\psi_n(\mathbf{w}[t])<1$ when ϵ is small enough. This completes the proof.

REFERENCES

[1] X. Lin and N. B. Shroff, "The impact of imperfect scheduling on cross-layer congestion control in wireless networks," *IEEE/ACM Trans. Netw.*, vol. 14, no. 2, pp. 302–315, Apr. 2006.

[2] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," *IEEE/ACM Trans. Netw.*, vol. 15, no. 6, pp. 1333–1344, Dec. 2007.

[3] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 396–409, Apr. 2008.

[4] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1452–1463, Aug. 2006.

[5] L. Tassiulas and A. Ephremides, "Stability properties of constrained queuing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[6] A. Eryilmaz and R. Srikant, "Joint congestion control, routing, and MAC for stability and fairness in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1514–1524, Aug. 2006.

[7] R. J. Gibbens and F. P. Kelly, "Resource pricing and the evolution of congestion control," *Automatica*, vol. 35, pp. 1969–1985, 1999.

[8] S. Kunniyur and R. Srikant, "Analysis and design of an adaptive virtual queue algorithm for active queue management," in *Proc. ACM SIGCOMM*, San Diego, CA, Aug. 2001, pp. 123–134.

[9] A. Laksmikantha, C. Beck, and R. Srikant, "Robustness of real and virtual queue-based active queue management schemes," *IEEE/ACM Trans. Netw.*, vol. 13, no. 1, pp. 81–93, Feb. 2005.

[10] L. Huang and M. J. Neely, "Delay reduction via lagrange multipliers in stochastic network optimization," *IEEE Trans. Autom. Control*, vol. 56, no. 4, pp. 842–857, Apr. 2011.

[11] M. J. Neely, "Super-fast delay tradeoffs for utility optimal fair scheduling in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1489–1501, Aug. 2006.

[12] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 3rd ed. New York, NY: John Wiley & Sons Inc., 2006.

[13] J. Liu, C. H. Xia, N. B. Shroff, and H. D. Sherali, "Distributed cross-layer optimization in wireless networks: A second-order approach," in *Proc. IEEE INFOCOM*, Turin, Italy, Apr. 14-19, 2013.

[14] J. Liu, N. B. Shroff, C. H. Xia, and H. D. Sherali, "Joint congestion control and routing optimization: An efficient second-order distributed approach," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1404–1420, Jul. 2016.

[15] J. Liu, A. Eryilmaz, N. B. Shroff, and E. Bentley, "Heavy-ball: A new approach to tame delay and convergence in wireless network optimization," in *Proc. IEEE INFOCOM*, April 10-15, 2016.

[16] E. Ghadimi, I. Shames, and M. Johansson, "Multi-step gradient methods for networked optimization," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5417–5429, Nov. 2013.

[17] J.-P. Tignol, *Galois' Theory Of Algebraic Equations*, 2nd ed. World Scientific Publishing Company, 2016.

[18] A. Beardon, *Complex Analysis: the Winding Number Principle in Analysis and Topology*. John Wiley and Sons, 1979.

[19] A. Eryilmaz, R. Srikant, and J. R. Perkins, "Stable scheduling policies for fading wireless channels," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 411–424, Apr. 2005.

[20] "High-order momentum: Improving latency and convergence for wireless network optimization," *Technical Report*, July 2017. [Online]. Available: https://1drv.ms/b/s!AqwF1oeOcX75oAMp24Bm1Q-2NxKa

[21] B. T. Polyak, *Introduction to Optimization*. New York, NY: Optimization Software, Inc., May 1987.

[22] M. Fujiwara, "Über die obere schranke des absoluten betrages der wurzeln einer algebraischen gleichung," *Tôhoku Math. J.*, vol. 10, pp. 167–171, 1916.