Decomposition of Large Scale MDPs for Wireless Scheduling with Load- and Channel-Awareness

Huasen Wu‡, Xiaojun Lin‡, Xin Liu§, Kun Tan†, and Yongguang Zhang†
‡School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
†Microsoft Research Asia, Beijing 100080, China
§Department of Computer Science, University of California, Davis, CA 95616, USA

Abstract—While cellular networks often need to be provisioned according to the peak demand, there usually exist large fluctuations in both traffic load and channel conditions. Embracing these two dimensions of dynamics allows us to exploit the delay-tolerance of data traffic to alleviate network congestion, and thus reduce the peak. In principle, scheduling with both load- and channel-awareness can be modeled as a Markov Decision Process (MDP). However, as the system size increases, the complexity increases exponentially due to the curse of dimensionality. In this paper, we propose a distributed approach, called Coordinated Scheduling (CoSchd), for solving such a large-scale MDP problem. Under CoSchd, each user independently solves an individual MDP problem, under a limited amount of coordination signal from the BS. The complexity of the individual MDP does not increase with the system size, and hence this approach can scale to large systems. Somewhat surprisingly, we show that the CoSchd approach is asymptotically optimal when the system size becomes large. To the best of our knowledge, this paper is the first work in the literature that studies load- and channel-awareness in a unified and rigorous manner. Further, the proposed CoSchd approach could potentially be used to solve other types of large-scale MDP problems, where multiple agents are weakly coupled by sharing the resource.

I. INTRODUCTION

A grand challenge facing today’s mobile service providers is to meet the exponentially increasing demand for mobile broadband services. This problem is particularly severe at the so-called “peak”, where the network is highly loaded at specific times and locations. Currently, wireless providers invest heavily in new spectrum and infrastructure to accommodate the peak demand, but such efforts are costly and inefficient: since the network traffic at non-peak times is orders of magnitude lower than peaks, provisioning network capacity for peak demand will lead to poor utilization of network resources.

An alternative approach is to exploit the delay tolerance of mobile applications to improve the network utilization. Prior work has identified a class of applications that can tolerate some delay, ranging from a few minutes to hours [1–4]. For example, the analysis in [3] shows that more than 55% of multimedia contents in cellular networks are uploaded more than one day later after their creation time. More recently, the survey conducted in the TUBE project [1] indicates that users are actually willing to delay their data transmissions if appropriate incentives are provided, i.e. a discounted price. Motivated by these studies, in this paper we are interested in finding the best scheduling of delay-tolerant traffic to minimize the network congestion in wireless networks.

Exploiting delay-tolerance can alleviate network congestion and reduce peak capacity requirement in two directions: load-awareness and channel-awareness. First, by moving delay-tolerant traffic to the time and location (e.g., to a Wi-Fi hotspot or a different base station (BS)) where the network is less loaded, i.e., being load-aware, the network can alleviate congestion and carry more traffic overall. This idea is akin to “peak-shedding” [1], where some of the “peak” traffic is moved to the “valleys” when the demand is low. Second, mobile Internet access is highly opportunistic in nature. Due to user mobility and wireless network dynamics, both the network connectivity and signal strength can vary significantly over time. Thus, by opportunistically scheduling traffic to a later period (or location) when the channel condition is more favorable, i.e., being channel-aware, the same amount of data transmission would consume less spectrum resources, which again alleviates network congestion and reduces peak capacity requirement. This second direction can be viewed as a form of opportunistic scheduling [5], albeit in a much larger time scale.

Previously, each aforementioned direction has been explored separately. For example, the TUBE project [1] studies how to schedule data traffic to less-congested periods based on user-specific deadlines and network incentives. However, TUBE does not consider users’ time-varying wireless channels – hence we classify it as a “load-only” approach. In the other direction, a number of channel-aware scheduling schemes have been proposed at the mobile device to improve spectrum efficiency [2,4,6]. While this line of work takes advantage of the opportunistic nature of wireless networks, it has been limited to optimizing in a single mobile device. As a result, these schemes are oblivious to network congestion and hence we refer to them as “channel-only” approaches.

Clearly, existing cellular networks provide opportunities for both load-aware and channel-aware approaches. Hence, several questions could be raised. First, conceivably we can combine both load-awareness and channel-awareness. How can we design such a policy that exploits both load-variations and channel-variations in both single- and multi-cell settings? Second, how do we compare the performance and complexity of the three approaches? Under what scenarios will they perform...
well? The answers to these questions will be very important from practitioners’ point of view in deciding which approach to adopt for future cellular networks, if one ever wants to exploit the delay-tolerance of the traffic.

In this paper, we present a unified analytical framework to study various scheduling policies for delay-tolerant traffics (i.e., load-only, channel-only, and joint approaches). We consider the scenario of a cellular network with multiple BSs and WiFi hotspots. Each data transfer request has a pre-specified deadline, which is directly tied to the users’ overall experience. The network’s objective is to schedule such data transfers intelligently to alleviate the network congestion and reduce peak demand, subject to the deadline constraints of the data transfers. We define the network congestion cost as the sum of (strictly-)convex functions of the load at each BS/WiFi-hotspot and at each time. With the strict-convexity, the cost function naturally penalizes high peak demand and thus a cost-minimizing solution will tend to smooth out the traffic load across time and location.

Based on our framework, we further study how to design optimal control policies. Naturally, the design of optimal policies for the joint approach is the most difficult, because we have to jointly consider “peak-shedding” and “opportunistic scheduling”. First, compared to the channel-only approach that only considers one mobile device [2, 4], here the size of the problem is very large, as a typical network may have hundreds of thousands of requests and a large number of BSs and WiFi hotspots. In addition, if a central entity needs to know all requests and channel evolution statistics for each individual user, it raises concerns on both signaling overhead and privacy. Second, compared to the load-only approach [1], here the channel uncertainty leads to significant difficulty in determining the amount of load that can be moved under a given policy. In our setting, the amount of traffic that users are willing to delay depends on their channels and the opportunistic scheduling algorithm, and thus the functional form is hard to obtain.

We make following contributions in the paper. First, we develop a distributed solution, referred to as CoSchd, for the joint approach. CoSchd uses a duality approach that addresses both the complexity issue and the signaling/privacy issues discussed earlier. Under CoSchd, the network does not need to know the statistics of all requests before hand, but updates a set of congestion signals based on the aggregated network load. At the same time, each user executes an individual decision policy based on the congestion signal and its own channel statistics. Second, somewhat surprisingly, we show the optimality of the dual approach. This is a non-trivial result given that our problem formulation accounts for a general set of policies that are complex, and it is not obvious that the corresponding objectives and constraints are convex with respect to these policies.

Finally, we have performed extensive trace-driven simulations to evaluate the performance gains of all three approaches. Our simulation results reveal following interesting insights: First, we find that the channel-only approach outperforms the load-only approach in our settings. Note that the load-only approach still requires the network to provide congestion signals, while the channel-only approach can be implemented entirely on the mobile device (without any network support). Thus, this suggests that channel-awareness is more effective and practical than load-awareness in wireless systems. Second, we find that in a single-cell, the joint approach, i.e., CoSchd, only leads to marginal performance gains compared to the channel-only approach. In other words, given that the channel-only approach defers transmission to time-instants with good channels, there is already a “spreading” effect across time that sheds the peak load. Thus, the additional room for CoSchd to further reduce the peak becomes smaller. In contrast, we find that in multi-cell scenarios, the potential gain for CoSchd to outperform the channel-only approach can be higher. This is because in a channel-only solution, it is possible that a congestion-oblivious mobile device may defer transfer until it moves to a BS with better signal quality. However, if this BS has a heavier load, such a channel-only approach will likely cause even higher peak at this BS.

In summary, this work makes both theoretical and practical contributions. Theoretically, the joint approach provides an optimal benchmark for comparison. Practically, we propose a simple distributed algorithm for optimal joint scheduling that is implementable in real systems. Further, our comparative evaluations provide the cellular operators with operation guidelines to decide their most appropriate approaches among load-only, channel-only, and joint schemes that balance the gain and complexity.

II. System Model

We start by considering a cellular network within one BS. The proposed approach can also be generalized to include multiple BSs and WiFi-hotspots, as discussed in [7]. The problem stated here applies to both the uplink and downlink in cellular networks.

Assume that time is slotted and indexed by \( t \in \{0, 1, \ldots \} \). Let \( N \) be the number of time-slots in each day. A typical time-slot length ranges from tens of seconds to a few minutes. Because of the large time scale, we assume that a data transfer request will be completed in one time-slot when the request is accepted, as in [1].

Data Traffic. In every day, a sequence of data transfer requests enter the network with user-specified deadlines. We use the words “user” and “request” interchangeably, though a human user may have multiple requests in a day. The requests depart upon completion or deadline violation.

Consider the scheduling problem in one day, where \( t \in \{0, 1, \ldots, N - 1\} \). Let \( I = \{1, 2, \ldots, m\} \) be the index set of all users that may request transfers from the BS. For each user \( i \in I \), denote the arrival time and the file size of its request by \( A_t \) and \( B_t \), respectively. Assume that \( A_t \)'s and \( B_t \)'s are i.i.d. across users. \( A_t \) follows a distribution that reflects the typical traffic pattern of the day [1, 8]. Note that a user may not request every day and we let \( A_t = -1 \) when user \( i \) does not request.
We assume that the file size $B_i$ is bounded [9] and is available as soon as the request arrives.

Each request $i$ is associated with a user- or application-specific deadline $D_i$, i.e., the maximum delay that a user can tolerate. The deadline ranges from minutes to hours for delay-tolerant traffic [1, 3], while it is set to zero for real-time traffic. Such a deadline requirement depends on specific applications and can be set in various ways. For example, it could be a default setting in an application, e.g., syncing emails every half an hour; or, it can be learned from user preference. We assume that no user can tolerate a delay larger than one day, i.e., $D_i \leq N - 1$. To guarantee the quality of user experience, we need to constrain the deadline violation probability when scheduling, as will be discussed later.

Channel Dynamics. Each user experiences time-varying network availability (e.g., WiFi availability) and channel conditions. This is captured by a stochastic process $R_i(t)$ ($t \in \{0, 1, \ldots \}$), where $R_i(t) \geq 0$ denotes the instantaneous rate per unit spectrum resource (e.g., a time-frequency block in LTE) at which the BS can communicate with user $i$ in time-slot $t$. We assume that $R_i(t)$ is i.i.d. across users and model $R_i(t)$ as a Markov chain over a finite set of the possible transmission rates, i.e., $R_i(t) \in \{r_1, r_2, \ldots, r_J\}$, where $J$ is the number of possible rates, and $0 = r_1 < r_2 < \ldots < r_J$. Thus, when user $i$ in channel condition $R_i(t)$ ($R_i(t) > 0$) is scheduled to transmit a file of size $B_i$, it consumes $B_i / R_i(t)$ units of spectrum resource.

We assume that each user can estimate its current channel condition via measurements of received signal strength and interference levels. Further, the user can learn the transition probability of its channel dynamics based on historical measurements, as in [6, 10, 11].

Scheduling Policy and Base Station Load. Let $\Gamma$ denote a general scheduling policy that decides which users to transmit at a given time slot. We consider the set of all casual policies. Corresponding to each $\Gamma$, we let $L_i(\Gamma)$ be the aggregate amount of spectrum resource consumed by the users transmitting in time-slot $t$ under policy $\Gamma$. We express $L_i(\Gamma)$ as

$$L_i(\Gamma) = \sum_{t \in \mathcal{I}} Y_{i,t}(\Gamma), \quad t = 0, 1, \ldots, N - 1,$$

where $Y_{i,t}(\Gamma)$ is the amount of resource consumed by user $i$ in time-slot $t$. More specifically,

$$Y_{i,t}(\Gamma) = \begin{cases} \frac{B_i}{R_i(t)}, & \text{if user } i \text{ transmits in slot } t, \\ 0, & \text{otherwise.} \end{cases}$$

Objective. From the network point of view, the objective of scheduling is to minimize the total congestion cost in the horizon of $N$ time slots under the deadline violation constraints. Let $f(\cdot)$ be a general convex congestion-cost function and $v_i(\Gamma)$ be the deadline violation probability of user $i$. The scheduling problem is then

$$(\mathcal{P}_0) \quad \min_{\Gamma} \quad F = \sum_{t=0}^{N-1} \mathbb{E}[f(L_i(\Gamma))],$$

subject to $v_i(\Gamma) \leq \eta_i, \quad \forall i \in \mathcal{I}$,

where $\eta_i$ is the maximum deadline violation probability tolerated by user $i$.

In problem $\mathcal{P}_0$, the convexity of $f(\cdot)$ penalizes peaks and thus favors load that is smoothed over time, which is desirable for network operators. In our numerical results, we use the following function $f(l) = (l/C)^\nu$, where $C$ is a positive constant and $\nu > 1$ is a factor for controlling the penalty. For a sufficiently large $\nu$, e.g., $\nu = 8$ in our simulations, we can penalize the situation when the load is above $C$ during network operation. It is worth noting that even though $f(\cdot)$ is a convex function, the general optimization problem may not be convex with respect to policies because the complex coupling of resource consumptions of users and their channel characteristics.

Note that in principle, $\mathcal{P}_0$ can be viewed as an MDP by taking the waiting time and channel condition of all users as the system state; but solving such an MDP problem in a centralized manner is forbiddingly complex. First, the size of the problem is very large, as a typical network may have hundreds of thousands of users, over a time horizon of a day. In addition, deadline constraint is notoriously difficult to solve in general because of the resource coupling over time and among users. Second, the problem formulation assumes knowledge of all jobs and their detailed channel information. In practice, it is not feasible to gather such detailed information in a central entity because of both signaling overhead and privacy concerns. However, we note that all users evolve independently and are only weakly coupled in sharing the resource. According to the Law of Large Number, the scheduling problem could be simplified when the number of users is large. Therefore, we turn to the many-source regime and study distributed approaches for $\mathcal{P}_0$. We show that the proposed approach is asymptotically in the many-source regime.

III. ASYMPTOTICALLY OPTIMAL DECOMPOSITION

This section studies asymptotically optimal policies for solving the large scale MDP $\mathcal{P}_0$ in (3), whose objective is to minimize the expectation of total cost. We first propose a lower bound of $\mathcal{P}_0$ by introducing a new problem $\mathcal{P}_1$ that minimizes the total cost of expectation. We show the asymptotic optimality of the decomposition approach in the many-source regime and a distributed implementation of the approach, referred to as Coordinated Scheduling (CoSchd).

A. Lower Bound

In the original MDP problem $\mathcal{P}_0$, the cost is a function of the instantaneous load level $L_i(\Gamma)$ and the objective is to minimize the expected total cost. Because the cost function $f(\cdot)$ is convex, the optimal value of $\mathcal{P}_0$ can be lower bounded by replacing the instantaneous load level with the expected load level. Specifically, consider the following problem that minimizes the total cost of expectation (expected value of load
Because of the linearity of expectation, we can use a distributed approach to minimize the expected consumed resource of user $i$ to minimize the expected consumed resource of user $i$ such that the latter term in (6) is minimized. Therefore, for given $\beta$, the dual objective function can be obtained by solving the following subproblems:

\begin{align}
(S\mathcal{P}_0) \quad & \minimize_{h, \beta \geq 0} \sum_{t=0}^{N-1} [f(h_t) - \beta h_t] \\
(S\mathcal{P}_i) \quad & \minimize_{\beta} \sum_{t=0}^{N-1} \beta y_{i,t}(\Gamma_t) \quad \text{subject to} \quad v_i(\Gamma_t) \leq \eta_i, \, i \in \mathcal{I}.
\end{align}

The master dual problem is

\begin{align}
(D_1) \quad & \maximize_\beta \, g(\beta) \\
& \text{subject to} \quad \beta \geq 0.
\end{align}

Since $f(\cdot)$ is convex, subproblem $S\mathcal{P}_0$ can be easily solved by convex optimization algorithms [12]. For subproblem $S\mathcal{P}_i$, we can view it as a constrained sequential decision problem and obtain the optimal policy $\Gamma_t$ by backward induction [13]. Therefore, the dual problem can be solved efficiently by using (sub-)gradient approach, as will be discussed in Section III-D.

For a general optimization problem, dual decomposition only guarantees weak duality, i.e., the dual solution only provides a lower bound to the original problem. However, we show below that the duality gap between $\mathcal{P}_1$ and $\mathcal{D}_1$ is zero, and hence the algorithms $S\mathcal{P}_0$ and $S\mathcal{P}_i$ combined provides an optimal solution to $\mathcal{P}_1$.

**Proposition 2** Given that the cost function $f(\cdot)$ is convex, the dual problem $\mathcal{D}_1$ has zero duality gap, and thus the dual decomposition approach provides an optimal solution to $\mathcal{P}_1$.

**Sketch of Proof:** We prove the strong duality by reformulating $\mathcal{P}_1$ into an alternative form that exhibits a convex structure albeit with a prohibitively large number of variables. The alternative form, named $\mathcal{P}_2$, is discussed in detail in the Appendix. Roughly speaking, $\mathcal{P}_2$ assigns a decision variable for each user in each possible state (a state is a possible combination of all users’ channel conditions, which is prohibitively huge). These decisions are coupled together because of the load aggregation, the deadline constraints, and channel state evolution. On the other hand, because $\mathcal{P}_2$ is convex, its dual, called $\mathcal{D}_2$, has zero duality gap with $\mathcal{P}_2$. We can then further show that the proposed backward induction approach for $(S\mathcal{P}_i)$, when combined with $(S\mathcal{P}_0)$, is optimal for $\mathcal{D}_2$, and thus solves $\mathcal{P}_2$ optimally. Because of the equivalence of $\mathcal{P}_2$ and $\mathcal{P}_1$, our proposed duality approach is optimal for $\mathcal{P}_1$. Details of the proof are available in the Appendix.

**C. Asymptotic Optimality in the Many-Source Regime**

According to Proposition 2, $\mathcal{P}_1$ can be solved by dual decomposition approach. However, $\mathcal{P}_1$ is not equivalent to the original problem $\mathcal{P}_0$. Fortunately, if all users independently solve individual MDPs, as the number of users increases, the instantaneous load level is close to its expectation. Using this property, we can show that the decomposition approach is asymptotic optimal for $\mathcal{P}_0$ in the many-source regime.
Consider the regime of many users. To study the asymptotic properties of the approach, we consider the following \(m\)-scaled system.

**Assumption 1** All users in \(I\) can be divided into \(K\) classes. For each class \(k\),
- the number of users \(m_k\) (\(k = 1, 2, \ldots, K\)) is proportional to the total number, i.e., \(m_k = m\lambda_k\), where \(0 < \lambda_k < 1\) is the ratio of class-\(k\) users and \(\sum_{k=1}^{K} \lambda_k = 1\);
- users in class-\(k\) have the same deadline requirements, and the same statistics of arrival patterns and channel dynamics that do not change with \(m\);
- users in class-\(k\) independently implement the same policies.

Further, we make the following assumption on the cost function:

**Assumption 2** The cost function \(f(\cdot)\) is a function of the normalized load, i.e., \(f(l) = \hat{f}(\hat{l})\), where \(\hat{l} = l/m\).

Let \(\Gamma^*\) be the optimal policy for \(P_0\), and let \(F_{l|I}^{(m)}\) be the objective value of \(P_0\) under \(\Gamma^*\), and \(E_{l|I}^{(m)}\) be the optimal value of problem \(P_1\), i.e., the objective value of \(P_1\) under \(\Gamma^*\). The following proposition shows the performance of the system in the many-source regime.

**Proposition 3** Under Assumptions 1 and 2, \(F_{l|I}^{(m)}\) approaches \(\lim_{m \to \infty} F_{l|I}^{(m)} = \lim_{m \to \infty} E_{l|I}^{(m)}\).

According to Proposition 1, \(E_{l|I}^{(m)}\) provides a lower bound on the value of \(P_0\). The above proposition states that \(F_{l|I}^{(m)}\) approaches \(E_{l|I}^{(m)}\) as \(m\) increases, and thus the decomposition approach is asymptotically optimal for \(P_0\) in the many-source regime.

**Proof:** We first consider the single class system, i.e., \(K = 1\). Since \(F_{l|I}^{(m)}\) is the sum of the expected costs in each slot, i.e., \(F_{l|I}^{(m)} = \sum_{t=0}^{N-1} E[f(L_t)]\), we can prove Proposition 3 if we can show that under \(\Gamma^*\),

\[
\lim_{m \to \infty} E[f(L_t)] = E[f(L_t)],
\]

which implies that the “expectation of the cost” approaches the “cost of the expectation” as \(m\) increases. This can be verified by using the fact, that under \(\Gamma^*\), each user operates independently when the congestion signal \(\beta\) is fixed.

Specifically, fixed a time-slot \(t\). Let \(Y_i\) (\(i = 1, 2, \ldots, m\)) be the amount of resource required by the \(i\)-th user in slot \(t\). We first consider the homogenous case. Thus, \(Y_i\) are i.i.d. random variables with mean value \(\mu_Y\). The load level is \(L_t = \sum_{i=1}^{m} Y_i\) and the normalized load level is \(\tilde{L}_t = \frac{1}{m} \sum_{i=1}^{m} Y_i\) with \(E[\tilde{L}_t] = \mu_Y\).

Using the Chernoff bound, we have that for a given \(\delta > 0\),
\[
\text{Pr}\{\tilde{L}_t \leq \mu_Y - \delta\} \leq e^{-mI_Y(\delta)},
\]
\[
\text{Pr}\{\tilde{L}_t \geq \mu_Y + \delta\} \leq e^{-mI_Y(\delta)},
\]

where
\[
I_Y(\delta) = \min\{D(\mu_Y + \delta || \mu_Y), D(\mu_Y - \delta || \mu_Y)\},
\]
and \(D(x||y) = x \log \frac{x}{y} + (1-x) \log \frac{1-x}{1-y}\) is the Kullback-Leibler divergence.

Next, we bound \(E[f(L_t)]\) using the above results and the properties of the cost function. \(E[f(L_t)]\) can be calculated as follows
\[
E[f(L_t)] = \int_0^\infty \hat{f}(l)\phi_{\tilde{L}_t}(l)dl = \int_0^{\mu_Y-\delta} \hat{f}(l)\phi_{\tilde{L}_t}(l)dl + \int_{\mu_Y-\delta}^{\mu_Y+\delta} \hat{f}(l)\phi_{\tilde{L}_t}(l)dl + \int_{\mu_Y+\delta}^\infty \hat{f}(l)\phi_{\tilde{L}_t}(l)dl.
\]

Note that \(\hat{f}(l)\) is increasing in \(l\). Thus, from (10) and (11), we have
\[
E[f(L_t)] \geq (1 - 2e^{-mI_Y(\delta)}) \hat{f}(\mu_Y - \delta),
\]
and
\[
E[f(L_t)] \leq \hat{f}(\mu_Y + \delta) + 2e^{-mI_Y(\delta)} \hat{f}(\gamma_{\text{max}}).
\]

For any \(\epsilon > 0\), by the continuity of \(\hat{f}(l)\), we can choose a \(\delta > 0\) such that \(\hat{f}(\mu_Y - \delta) \geq \hat{f}(\mu_Y) - \epsilon/2\) and \(\hat{f}(\mu_Y + \delta) \leq \hat{f}(\mu_Y) + \epsilon/2\). Combining with (13) and (14), we know that there exists an \(m_1\) such that for all \(m \geq m_1\), we have
\[
|E[f(L_t)] - E[f(L_t^*)]| = |E[f(L_t)] - \hat{f}(\mu_Y)| \leq \epsilon.
\]

The conclusion then follows by adding the expected costs from time-slot 0 to \(N-1\).

**Remark:** Proposition 3 is important because the original problem \(P_0\) is exponentially complex and it is unclear whether distributed policies can achieve the global optimality. By introducing a variant \(P_1\) of the original problem, we are able to design a distributed approach and further show its asymptotic optimality with the distributed property.

**D. Distributed Implementation**

Based on the asymptotic optimality of dual decomposition, we propose the following distributed algorithm, referred to as Coordinated Scheduling (CoSchd), to implement the optimal solution to \(P_1\). In this distributed algorithm, the BS decides its congestion signal vector in an iterative fashion and each user individually decides its transmission schedule based on the congestion signal and its channel information.

We note that this framework could be applied in both an online and an offline fashion. In the offline fashion, the BS would use all users’ channel information and terminate the algorithm if certain stopping criterion is satisfied (e.g., when the duality gap is smaller than a predetermined threshold). Because the offline solution would require all statistical information to be available, it may incur high signaling overhead and privacy issues.

On the other hand, we can choose to implement an online solution, where each iteration represents one day (or weekday).
assuming traffic statistics does not change significantly across days. In this case, the BS only needs to update the congestion signal \( \beta^{(d)}_{t} \) based on the observed traffic load. In other words, the BS does not need to know the detailed information of users, and thus resolves the concerns on signaling overhead and user privacy. Note that in this case, the value of \( \beta^{(d)}_{t} \) in (18) should be replaced by its random realization in the \( d \)-th iteration. This modified version of (18) then has the flavor of stochastic approximation algorithms [5, 14]. When the step-size \( \alpha_{d} \) is small, we would expect the modified version to also converge to a small neighborhood of the optimal solution. In the rest of the paper, we focus on the online approach. Next, we discuss the operations on the mobile side and on the network side, respectively.

1) Mobile-side Operation.: On the mobile-side, each user operates independently as follows: generates policies based on its channel characteristics and the congestion signals, and then executes the policy based on the instantaneous channel condition.

For a given \( \beta \) (i.e., the congestion signal), the subproblem \( SP_{i} \) under the deadline constraint in Eq. (13) turns out to be a constrained sequential decision problem [13]. In particular, one can introduce a cost for a deadline violation. The mobile minimizes \( SP_{i} \) plus the deadline violation cost by using backward induction, where the user makes decision by comparing the transmission cost in current time-slot and the cost-to-go.

We discuss the deterministic deadline constraint case as follows and refer the readers to [13] for the probabilistic deadline constraint case.

For user \( i \) arriving at \( a_{i} \), let \( x_{a_{i},w,j} \in [0, 1] \) \((w = 0, 1, \ldots, D_{i} - 1; j = 1, 2, \ldots, J)\) be the probability that user \( i \) requests transmission when its waiting time is \( w \) and channel state is \( j \). In the deterministic deadline constraint case, i.e., \( \eta_{t} = 0 \), all data should be transmitted before expiration. Therefore, for user \( i \) arriving at \( a_{i} \), it requires that \( x_{a_{i},w,j} = 1 \). To guarantee a finite transmission cost, we assume that for each user,

\[
E\{b_{l}/R_{l}(a_{i} + D_{i} - 1)\mid \mathcal{E}_{l}, D_{l} - 1\} < +\infty, \quad i \in \mathcal{I},
\]

where \( \mathcal{E}_{l}, D_{l} - 1 \) represents the event that user \( i \) does not transmit before \( a_{i} + D_{i} - 1 \). In the case with temporally-Markovian channels, using the principle of optimality and taking the multipliers \( \beta \) into account, we can obtain the optimal decision

\[
x_{a_{i},w,j} = \begin{cases} 1, & \text{if } \frac{β_{t}^{(a_{i}+w|N)}}{R_{l}(a_{i}+w|N)} \leq E[V_{i,w+1}^{*}|r_{j}] \\ 0, & \text{otherwise,} \end{cases}
\]

where \( E[V_{i,w+1}^{*}|r_{j}] \) is the expected future cost conditioned on \( R_{l}(a_{i} + w|N) = r_{j} \), which can be calculated by backward induction:

\[
\begin{aligned}
E[V_{i,w+1}^{*}|r_{j}] &= \begin{cases} \frac{β_{t}^{(a_{i}+w|N)}}{R_{l}(a_{i}+w|N)} |r_{j}|, & \text{for } w = D_{i} - 2; \\
\min \left( \frac{β_{t}^{(a_{i}+w+1|N)}}{R_{l}(a_{i}+w+1|N)} V_{i,w+2}^{*} \right), & \text{for } w = D_{i} - 3, D_{i} - 4, \ldots, 0.
\end{cases}
\end{aligned}
\]

As a special case of Markovian channels, when the channel process is independent across time-slots, it is easy to verify that the policy becomes a threshold policy, i.e., there exists a threshold \( T_{w} \) for each \( w \), the transfer occurs if \( R_{l}(a_{i} + w) \geq T_{w} \).

To deal with the oscillation issues in the subgradient method [12], each user applies an Exponential-Moving-Averaging policy, i.e.,

\[
\bar{x}_{a_{i},w,j}^{(d)} = \vartheta \bar{x}_{a_{i},w,j}^{(d-1)} + (1 - \vartheta) x_{a_{i},w,j}^{(d)},
\]

where \( \vartheta \in [0, 1] \) is a memory factor for trading-off between the convergence speed and fluctuation, as will be discussed in Section IV.

With the policy, each user estimates its instantaneous channel condition and then makes decision using the decision table, as shown in Eq. (16).

2) Network-side Operation: The network-side operation consists of two components: serving users and updating congestion signals (for the next day). The update mechanism is described in Eq. (18), and thus we focus on serving users next.

In Algorithm 1, we follow a (sub-)gradient method to solve the dual problem \( D_{1} \):

\[
\beta^{(d+1)}_{t} = \left[ \beta^{(d)}_{t} + \alpha^{(d)}(t^{(d)}_{t} - h^{(d)}_{t}) \right]^{+}, \quad \forall t
\]

where \( d \) is the iteration index, \( \alpha^{(d)} \) is the step-size, and \( [ \cdot ]^{+} \) denotes the projection to nonnegative numbers.

IV. Evaluation

We evaluate the performance of load-only, channel-only, and CoSchd approaches via trace-driven simulations. As a baseline, we also consider ImTrans, where all users immediately transfer the data when the requests arrive.
A. Simulation Setup

We use a slot length of 10 minutes and each day is divided into 144 time-slots. We consider both a single-cell scenario and a two-cell scenario, except the load-only policy will only be evaluated in the single-cell scenario as discussed in Section ??.

1) Traffic Arrival Pattern: We assume a time-dependent Poisson arrival process, i.e., the total number of requests arriving in time-slot \( t \) is a Poisson random variable with mean value \( \lambda_i \ (t = 0, 1, \ldots, N - 1) \). For the single-cell network, the mean arrival rates are set based on the weekday traffic profile of the center BS shown in Fig. ???. For the multi-cell network, we use the weekday traffic profile of the center BS and the neighbor BS 1 (again in Fig. ??). To capture the delay-tolerance of traffic, we apply the waiting function proposed in [1], and use the patience indices for the different traffic classes estimated from the U.S. survey in [1]. Specifically, for the delay-tolerant traffic ("Time-Dependent Pricing" traffic in [1]), the probability that user \( i \) wants to wait \( D_i \) slots is proportional to \( \frac{1}{(D_i + 1)^\rho} \), where the patience index \( \rho \) is 2.0 for video traffic and 0.6 for others. In addition, the usage distribution of the different traffic classes is taken from recent estimates [15], where the proportion of video traffic is about 65%.

2) Channel Profile: We collected a set of Received Signal Strength Indication (RSSI) values from a group of anonymous mobile users to best emulate the spectrum efficiency in cellular networks. We assume that the interference strength is a constant and thus the RSSI value represents the SINR, which determines the spectrum efficiency. We follow the LTE-Advanced standards [16], and map the measured RSSI to the proper modes of Modulation and Coding Scheme (MCS). We use the 5-bit CQI and the distribution of the corresponding spectrum efficiency is shown in Fig. 1. To capture time-varying and location-dependent channel conditions, we use a Markov model where Markovian transitions between adjacent channel states (RSSI values) are assumed in each time-slot [17]. We assume that all users use the same channel model. One limitation of the model is that the parameters (e.g., transition probabilities) do not change over time while real human users may have more time-dependent behavior (e.g., 2am at home vs. 2pm at work). We hope to further collect real-life channel profile traces for a more realistic evaluation of real networks in the future.

B. Convergence of CoSchd

We first demonstrate the asymptotic behavior of the system and the convergence of CoSchd.

Fig. 2 shows the difference between the values of the original problem \( P_0 \) and its approximate version \( P_1 \). As we can see from the figure, by minimizing the total cost of expected load, the approximate problem \( P_1 \) provides a lower bound on the optimal value of the original problem. The gap between the optimal value and its lower bound becomes smaller as the network scale increases. The two values are close to each other in the medium-sized system. For example, when the average number of users in each slot is 400, the gap is about 15% of the value for the original problem.

Fig. 3 shows the evolution of the duality gap between problem \( P_1 \) and its duality. The duality gap decreases as the number of iterations increases, and the duality gap is small after several iterations. Comparing the evolutions with different memory factor \( \vartheta \), we can see that with smaller \( \vartheta \), the duality gap decreases faster, but the fluctuation is larger. By trading-off between the convergence speed and fluctuation, we set \( \vartheta = 0.9 \) for the rest of the simulations.
C. Network Load

Fig. 4 shows the network load in one day obtained by different approaches. The three subfigures represent different settings. Fig. 4(a) and Fig. 4(b) are for single-cell systems with 50% and 75% of load being delay-tolerant, respectively. In contrast, Fig. 4(c) is for a multi-cell system with 50% of load being delay-tolerant. We can make a number of interesting observations from Figs. 4(a) and 4(b). Specifically, from Fig. 4(a), we can see that by moving the delay-tolerant traffic into “valleys”, the peak load obtained by the load-only policy is about 80% of that under ImTrans. On the other hand, using the channel-only policy, the peak is reduced to about 75% of ImTrans. A similar observation can be made from Fig. 4(b), while the peak load reduction is more significant since there is more delay-tolerant traffic. This finding suggests that channel-awareness can be more effective than load-awareness in wireless systems.

Further, although CoSchd leads to even lower peak consumption by considering both load-awareness and channel-awareness, the additional gain compared to the channel-only policy is relatively marginal in the single-cell setting in Fig. 4(a) and Fig. 4(b) (about 8% reduction in both figures). We note that, under the channel-only policy, users defer their transmissions when waiting for good channels. Therefore, a “peak-shedding” effect also occurs under the channel-only approach. Since the traffic fluctuation is not large, the room for CoSchd to further move traffic is relatively small. However, the multi-cell simulation in Fig. 4(c) illustrates different behaviors. By moving the delay-tolerant traffic to the neighbor BS (i.e., BS 2), the peak of network load (corresponding to the load in BS 1 at about 18:00) is reduced by about 20% by CoSchd compared to the channel-only policy.

V. Conclusions

In this paper, we present a unified framework to study the effectiveness of load- and/or channel-awareness in deadline-constrained scheduling of delay-tolerant traffic for the purpose of alleviating cellular network congestion. Despite the high complexity of the joint scheduling problem with explicit deadline constraints, we develop an optimal solution, called CoSchd, and propose its distributed implementation.

The results in the paper are of both theoretical and practical values. Theoretically, the joint approach provides an optimal benchmark for comparing with other solutions. Practically, our proposed policy can be implemented in a distributed manner in real systems. Further, our comparative evaluations provide the cellular operators with operation guidelines to decide their most appropriate approaches. Specifically, our numerical results suggest that channel-awareness is rather important in wireless networks. For single-cell systems, channel-only may be preferred due to its simplicity and relatively good performance. For multi-cell systems with load variations, CoSchd can attain significant additional gains. Finally, we note that the relative performance of the three approaches may vary depending on the actual load and mobility patterns. Hence, for future work it will be highly desirable to use real-life traces from users within the same set of multiple cells, including their mobility and load variations, to conduct more comprehensive evaluations.
VI. APPENDIX

To prove the optimality of the proposed dual decomposition approach, we show that problem $P_1$ can be reformulated to a convex optimization problem $P_2$, albeit with an exponentially large number of decision variables. In addition, our proposed threshold policy can optimally solve the dual of $P_2$, named $D_2$, and thus our scheme is optimal.

First, we note that any policy $\Psi$ can be represented by a stochastic policy $\Psi$. Each causal policy $\Gamma$ makes decision based on the history of the arrival sequence and channel processes. To represent the history, for each user $i \in \mathcal{I}$, we introduce $A_i(t)$ to represent its present status in time-slot $t$. Namely, if user $i$ arrives in time-slot $a_i$ (we let $a_i = N$ represent the event that user $i$ does not appear), then $A_i(t) = -1$ if $a_i > t$, and $A_i(t) = a_i$ if $a_i \leq t$. Recall that $R_i(t)$ ($i = 0, 1, \ldots, N - 1$) is the channel process of user $i$. Hence, the history of the system up to time-slot $t$ is given by

$$ S_t = [A_t, R_t], $$

where $A_t = [A_t(1), A_t(2), \ldots, A_t(T(t))]^T$ and

$$ R_t = \begin{bmatrix} R_t(0) & R_t(1) & \cdots & R_t(T(t)) \\ R_{t+1}(0) & R_{t+1}(1) & \cdots & R_{t+1}(T(t)) \\ \vdots & \vdots & \ddots & \vdots \\ R_{T(t)}(0) & R_{T(t)}(1) & \cdots & R_{T(t)}(T(t)) \end{bmatrix}. $$

Let $\Omega$ be the set of possible realizations of arrival sequence and channel processes, i.e., the possible realization of $S_{N-1}$. Then each policy $\Gamma$ can be represented by a stochastic policy $\Psi$, which is a $\Omega \rightarrow [0, 1]^{|\mathcal{I}| \times N}$ mapping: for each $s \in \Omega$,

$$ \Psi(s) = \begin{bmatrix} \psi_1(s_0) & \psi_1(s_1) & \cdots & \psi_1(s_{N-1}) \\ \psi_2(s_0) & \psi_2(s_1) & \cdots & \psi_2(s_{N-1}) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{|\mathcal{I}|}(s_0) & \psi_{|\mathcal{I}|}(s_1) & \cdots & \psi_{|\mathcal{I}|}(s_{N-1}) \end{bmatrix}, $$

where $s_i$ is the history of arrival sequence and channel processes up to time-slot $t$ for the realization $s$, and each $\psi_i(s_t) \in [0, 1]$ is the transmission probability of user $i$ in time-slot $t$. Note that for the user with $a_i + D_i \geq N$, it may transmit in the following day and the transmission probability is represented by $\psi_i(s_{[a_i+w|N]})$ ($w = 0, 1, \ldots, D_i - 1$), where $[\cdot | N]$ is a mod over $N$ function. For these users, $s_i$ includes the history in the past day. Therefore, the policy $\Psi$ makes decisions based on the arrival sequence and channel conditions until the current slot and is thus causal.

Second, we study the expected resource consumed by user $i$ under $\Psi(s)$. For each $s \in \Omega$ where user $i$ arrives in time-slot $a_i$, we can calculate the probability that user $i$ transmits in slot $a_i + w$ as follows

$$ \varphi_i(s_t) = \begin{cases} \psi_i(s_{a_i}), & t = a_i \\ \psi_i(s_{[a_i+w|N]}) \sum_{w'=0}^{w-1} [1 - \psi_i(s_{[a_i+w'+N]}|N)], & t = [a_i + w|N], 0 < w \leq D_i - 1 \\ 0, & \text{otherwise}. \end{cases} $$

For given $s$, the expected consumed resource of user $i$ in time-slot $t$ is

$$ c_{i,t}(s, \Psi) = \frac{b_i \varphi_i(s_t)}{R_t(t)}. $$

In addition, note that all users should transmit before expiration. Hence,

$$ \sum_{w=0}^{D_i-1} \varphi_i(s_{[a_i+w|N]}) = 1, \ s \in \Omega, i \in \mathcal{I}. $$

Moreover, using the relationship between $\varphi_i(\cdot)$ and $\psi_i(\cdot)$, a $\varphi_i(\cdot)$ satisfying (19) can be mapped to a policy $\Psi$.

Consequently, problem $P_1$ is equivalent to

$$ \begin{array}{l} \text{minimize} \\
\quad \quad \Psi, h \end{array} \begin{array}{l} F = \sum_{t=0}^{N-1} f(h_t'), \\
\text{subject to} \\
\quad \quad \sum_{w=0}^{D_i-1} \varphi_i(s_{[a_i+w|N]}) = 1, \ s \in \Omega, i \in \mathcal{I}, \\
\quad \quad l'_t(\Psi) \leq h'_t, \ t = 0, 1, \ldots, N - 1, \end{array} $$

where

$$ l'_t(\Psi) = \sum_{w=0}^{D_i-1} \sum_{s \in \Omega} \pi(s) c_{i,t}(s, \Psi). $$

We can verify that $P_2$ is a convex optimization problem because $f(\cdot)$ is a convex function and all the constraints are linear constraints. However, we do note that it is impractical to solve $P_2$ directly because of its large number of variables. Recall that there are $|\mathcal{I}| \times N$ decision variable for each possible state. Assume the channel state of each user can be quantized to $K$ values, then there are $K^{|\mathcal{I}|} \times N$ possible states, and thus $|\mathcal{I}| \times N \times K^{|\mathcal{I}|} \times N$ decision variables, which is clearly intractable. We note that the formulation can be considered as a linear representation of a centralized Markov Decision Policy, which clearly suffers the curve of dimensionality.

Again, we resort to the dual decomposition approach to study $P_2$. Similar to the approach in Section III, we can introduce a dual variable for each time slot, and then rearrange the variables that belongs to each user. Then we have a similar format as in $SP_0$ and $SP_r$. Each user $i$ solves $SP_i$; which minimizes its transmission cost weighted by the congestion signal, under the delay constraint. It can be shown that the proposed threshold policy provides an optimal solution to the above problem. Details are omitted here because it is a standard stochastic dynamic programming problem [13].

REFERENCES


1If $\sum_{w=0}^{w'} \varphi_i(\tilde{r}_{D_i,a_i+w'}) = 1$ for some $w < D_i - 1$, then for $w' > w$, $\psi_i(\tilde{r}_{w,w'})$ can be artificially set to be 0, which will not affect the behavior of $\Psi$. 


