Utility-Optimal Dynamic Rate Allocation under Average End-to-End Delay Requirements

Mohammad H. Hajiesmaili[1], Mohammad Sadegh Talebi[2], and Ahmad Khonsari[3]

***Abstract*— QoS-aware networking applications such as real-time streaming and video surveillance systems require nearly fixed average end-to-end delay over long periods to communicate efficiently, although may tolerate some delay variations in short periods. This variability exhibits complex dynamics that makes rate control of such applications a formidable task. This paper addresses rate allocation for heterogeneous QoS-aware applications that preserves the long-term average end-to-end delay constraint while, similar to Dynamic Network Utility Maximization (DNUM), strives to achieve the maximum network utility aggregated over a fixed time interval. Since capturing temporal dynamics in QoS requirements of sources is allowed in our system model, we incorporate a novel time-coupling constraint in which delay-sensitivity of sources is considered such that a certain end-to-end average delay for each source over a pre-specified time interval is satisfied. We propose DA-DNUM algorithm, as a dual-based solution, which allocates source rates for the next time interval in a distributed fashion, given the knowledge of network parameters in advance. Through numerical experiments, we show that DA-DNUM gains higher average link utilization and a wider range of feasible scenarios in comparison with the best, to our knowledge, rate control schemes that may guarantee such constraints on delay.**

I. Introduction

Nowadays, a plethora of networking applications have emerged that are delay-sensitive, and require some guarantee on the end-to-end delay. Despite instantaneous delay sensitivity shown by some applications, one may identify several others that only concern the average end-to-end delay over some interval of interest. A notable instance is media streaming where end-to-end delay, averaged over a pre-specified interval, is obliged not to exceed some threshold to ensure continuous playback. Some other examples include real-time WSNs and delay-constrained networked control systems. In such scenarios, due to temporal variations in both source traffic and network characteristics, we face an ever increasing need to accomplish rate allocation capable of capturing such dynamicity.

As a promising framework, Network Utility Maximization (NUM) has been exploited in several network resource allocation scenarios; see, e.g., [1]–[3]. In its simplest form, NUM concerns a network that supports a set of sources and links. Each source is associated with a utility as a function of its rate and transmits its packets through a route, which is a subset of the links in the network. The fixed capacity of links and routing structure dictate a set of linear capacity constraints. The goal of the NUM problem is to find source rates that maximize the aggregate utility of the network given capacity constraints.

Several studies have thus far incorporated end-to-end delay in the basic NUM model; see, e.g., [4]–[9], and the detail mentioned in Section II. In these studies, end-to-end delay either is included in the objective function or introduced some constraints to the problem. Despite these studies, the basic NUM framework is intrinsically incapable of capturing temporal variation in network characteristics especially when they evolve with time scales comparable to those of the underlying dual-based algorithms. Generally speaking, (single-period) NUM along with delay constraints is subject to limited degrees of freedom, and consequently, one may face a broad range of infeasible problems.

The conquest of variability-aware NUM-based approaches was further followed by [10], where Dynamic NUM (DNUM), a multi-period extension of NUM, was proposed. Indeed, DNUM considers the network utility aggregated over a finite time interval and thereby takes into account temporal variations in the parameters involved in the system model. Moreover, it allows linear constraints on source rates, referred to as *delivery contracts*, which may be construed as QoS constraints over the time interval. Such delivery contracts, however, are incompetent to capture more complicated key features such as queuing delays and jitter. In contrast to single-period NUM that suffers from limited degrees of freedom, DNUM offers several flexibilities. In particular, the former may face lots of infeasible problems whereas the latter admits relatively larger set of feasible problems yet higher total aggregated utility.

In this paper, we propose a variant of DNUM that strives to allocate source rates so as to satisfy constraints on end-to-end delays as well as capacity constraints. Towards this, the main contributions of this paper are summarized as follows:

▷ Built upon DNUM framework, we characterize the average end-to-end delay requirements of sources as a set of general and well-structured constraints. Our proposed model in Section III is a generalized version of the model that is built on [7], and thereby it avoids precise knowledge of underlying packet arrival models and relies only on the derivative of the delay function. Generalization of the model of [7] to a multi-period setup provides several flexibilities.

[1]Mohammad H. Hajiesmaili is with the Institute of Network Coding, The Chinese University of Hong Kong (CUHK), Sha Tin, N.T., Hong Kong, mohammad@inc.cuhk.edu.hk

[2]Mohammad Sadegh Talebi is with the School of Electrical Engineering, KTH Royal Institute of Technology, SE-10044, Stockholm, Sweden, mstms@kth.se

[3]Ahmad Khonsari is with School of Electrical and Computer Engineering, College of Engineering, University of Tehran, and with the School of Computer Science, Institute for Research in Fundamental Sciences (IPM), Niavaran Sq., Tajrish Sq., Tehran, Iran, ak@ipm.ir

The most promising one, perhaps, is that it allows some degree of freedom to sacrifice utility in some periods so as to maintain delay while compensating for it in some other periods. Secondly, our proposed formulation endows us the ability of maintaining several delay constraints for each source, where each delay constraint concerns a particular time interval of interest (see the discussion in Section III-C.1 for an real example).

▷ We develop a distributed algorithm called *Delay-aware Dynamic Network Utility Maximization* (DA-DNUM) in Section IV that solves the problem granted the knowledge of parameters for the next time interval in advance. Our solution is based on dual decomposition approaches and since we concentrate on strongly convex delay functions and consequently cast the rate allocation as a convex optimization problem, the problem can be efficiently solved in a distributed way thanks to existing dual-based approaches.

▷ Finally in Section V, we verify the correctness of our proposed solution and DA-DNUM algorithm by a set of tractable numerical experiments and give some comparison scenarios to demonstrate its superiority against the relevant state-of-the-art rate allocation schemes. As an interesting observation, our result corroborates that the proposed temporal formulation enlarges the set of feasible scenarios in comparison with [7].

A. Basic Notations and Terminologies

Throughput the paper, we use the following notations. For any vector $\mathbf{z}$ (resp. matrix Z), $\mathbf{z} \geq 0$ (resp. $Z \geq 0$) means that all components of vector $\mathbf{z}$ (resp. matrix Z) are non-negative. The vector $\mathbf{e}_j$ denotes the j-th unit vector. The operator $\|.\|$ signifies standard Euclidean norm. The domain of a function f is denoted by $\mathbf{dom}\ f$. Moreover, $\mathbf{1}_A$ is 1 if A occurs and 0 otherwise. Finally, $[.]^+$ and $[.]_{\mathcal{P}}$ respectively define the projection onto the positive orthant and set $\mathcal{P}$. We also give some necessary definitions that can be found in, e.g., [11].

Definition 1: A function $f(.)$ is a G-Lipschitz function if

$$|f(\mathbf{x}_1) - f(\mathbf{x}_2)| \leq G\|\mathbf{x}_1 - \mathbf{x}_2\|, \quad \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbf{dom}\ f.$$

Definition 2: A convex function $f(.)$ is κ-strongly convex if and only if there exists a constant $\kappa > 0$ such that the function $f(\mathbf{x}) - \frac{\kappa}{2}\|\mathbf{x}\|^2$ is convex.

It can be seen that if $f(.)$ is convex and satisfies $\|\nabla f(.)\| \leq G$, then it is G-Lipschitz. We remark that if $f(.)$ is twice differentiable then $f(.)$ is κ-strongly convex if there exists constant κ such that $\nabla^2 f(\mathbf{x}) - \kappa I$ is positive semidefinite.

II. Related Work

A number of studies have incorporated end-to-end delay into the basic NUM framework. In these works, end-to-end delay either is included in the objective function of NUM (e.g., [4], [9], [12]) or is treated as constraint of the underlying optimization problem (e.g., [6], [7]).

Delay as objective function. In [4], delay is incorporated into the objective function and therefore, delay plays its role as a penalty to the utility function. Consequently, the goal is to simultaneously maximize the aggregated utility of all sources and reduce the end-to-end delays. Based on a delay-sensitive utility function introduced in [13], authors in [9] aim to propose some application-oriented rate allocation schemes employing an alternative utility definition. Both approaches, however, prove incompetent to provide some guarantee for delay, thereby fail to be employed in QoS-aware applications with hard long term average delay requirements.

Delay as constraint. In another set of works [6], [7], [14], the source delay is introduced as constraints of the optimization problems. By introducing Virtual Link Capacity Margin (VLCM) to characterize source delay as constraint, the authors in [7] have proposed a joint rate allocation and scheduling scheme in multi-hop wireless networks. By a different approach in [6], another variant of NUM problem is formulated to address joint power and rate control. Generally speaking, NUM along with delay constraints is subject to limited degrees of freedom, and as a result, one may face a broad range of infeasible problems. We will investigate this phenomenon in details in our experiments in Section V.

To capture dynamics in the network and sources, NUM framework has been extended to the DNUM framework [10] that supports time-varying characteristics in network model parameters such as flow utilities, links capacities, and routing matrix. The DNUM framework has been extended in different research areas [15]. In [15], the time-varying nature is utilized to consider temporal variations in modeling the utility of the sources with video streaming applications.

III. Model and Problem Formulation

A. Network Model

Our model is based on that of DNUM [10], which considers rate allocation over a discrete-time interval $\mathcal{T} = \{1, \ldots, T\}$[1]. We assume that network possesses a set $\mathcal{L} = \{1, \ldots, L\}$ of links shared among a set $\mathcal{S} = \{1, \ldots, S\}$ of sources. We represent the possibly time-varying routing in the network defined by routing matrices $R_t = [(R_t)_{ls}]_{L\times S}, t \in \mathcal{T}$, whose element $(R_t)_{ls}$ is defined as follows:

$$(R_t)_{ls} = \begin{cases} 1 & \text{if } s\text{-th source uses } l \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$$

We let c_{tl} denote the capacity of link l at period t and $\mathbf{c}_t = [c_{tl}]_{l\in\mathcal{L}}$ be the vector of link capacities at period t.

Moreover, we let $x_{st} \in \mathcal{X}_{st}$ be the transmission rate of source s at period t, where $\mathcal{X}_{st} \triangleq [w_{st}, W_{st}]$ and w_{st} and W_{st} are the minimum and the maximum rates of source s at period t, respectively. We further require $0 < w_{st} \leq W_{st}, \quad \forall s, t$. We let $X = [x_{st}]_{S\times T}$ be the *rate matrix* and define $\mathcal{X} = \{X \in \mathbb{R}^{S\times T} : x_{st} \in \mathcal{X}_{st}\}$. A feasible rate matrix X then satisfies: $X \in \mathcal{X}$.

[1]The duration of each period t and the whole time horizon T is an application-specific design parameter. As an example, in [15], video streaming is the underlying application, thus, each period is set according to the length of the video frames and the time horizon T is set according to the length of GOPs (Group Of Pictures).

B. Capacity Constraints

To model capacity constraints, we first give the definition of *link margin* variables: for each link l and time period t, link margin variable σ_{tl} is defined as the difference between capacity of link l and the maximum allowable flow passing through it [7]. Unlike [7], however, our setup does not admit schedulability constraints and hence we proceed to formulate link margin as follows. Consider conventional capacity constraint for link l at period t given by

$$\sum_{s\in\mathcal{S}}(R_t)_{ls}x_{st}+\sigma_{tl}=c_{tl} \quad \text{and} \quad \sigma_{tl}\geq 0.$$

We then relax the equality constraint above and establish the following constraints for link l at period t:

$$\sum_{s\in\mathcal{S}}(R_t)_{ls}x_{st}+\sigma_{tl}\leq c_{tl} \quad \text{and} \quad \sigma_{tl}\geq 0.$$

The relaxation above, though constricts resource usage (i.e., capacity), plays an important role in limiting the flow of link l and thereby proves essentially useful to control the queuing delay of link l. Introducing $\boldsymbol{\sigma}_t=[\sigma_{tl}]_{l\in\mathcal{L}}$ and $\boldsymbol{\sigma}=[\boldsymbol{\sigma}_t]_{t\in\mathcal{T}}$, we then represent the capacity constraints in a compact way as

$$R_tX\mathbf{e}_t+\boldsymbol{\sigma}_t\leq\mathbf{c}_t \quad \text{and} \quad \boldsymbol{\sigma}_t\geq 0, \quad \forall t\in\mathcal{T}. \tag{1}$$

These constraints constitute a set of $2T\times L$ linear inequalities.

C. Average Delay Constraints

Having defined link margin variables, we define $D(\sigma_{tl})$ as the delay of link l at period t. Clearly, the way $D(\sigma_{tl})$ depends on σ_{tl} is determined by the packet arrival model. For instance, for M/M/1 queuing model whose packet arrival is a Poisson process, we have

$$D(\sigma_{tl})=\frac{q}{\sigma_{tl}}, \qquad q>0. \tag{2}$$

Another notable instance is the case of M/G/1 queuing model whose delay function is given in [6] and [16].

In what follows, we list our assumptions on the delay function $D(.)$:

A1. $D(.)$ is twice differentiable.
A2. $D(.)$ is G-Lipschitz.
A3. $D(.)$ is κ_D-strongly convex.

A notable example that satisfies these assumptions is the delay function of (2). We also remark that these assumptions are valid for M/G/1-based arrival processes, thereby cover the majority of existing queuing models.

In the present study, we only consider queuing delays and hence, for each source s, we obtain the end-to-end delay by simply adding up all link delays along the path of s. Writing ϕ_{st} for the end-to-end queuing delay of source s at period t, we get

$$\phi_{st}=\sum_{l\in\mathcal{L}}(R_t)_{ls}D(\sigma_{tl}).$$

We further introduce $\boldsymbol{\phi}_s=[\phi_{st}]_{t\in\mathcal{T}}$. Next, we define the constraint on average end-to-end delay as follows: Assume that source s requires its average end-to-end queuing delay over some interval of interest $\mathcal{T}_\Delta\subseteq\mathcal{T}$ with length Δ be less than some constant d. This constraint is formally given by

$$\frac{1}{\Delta}\sum_{t\in\mathcal{T}_\Delta}\phi_{st}\leq d. \tag{3}$$

To model a general scenario for the introduced delay constraint, we assume that each source s requires K_s delay constraints of the form (3), indexed by $k\in\mathcal{K}_s=\{1,\ldots,K_s\}$. In Section III-C.1, we provide a real-world example as a realization of this consideration in a typical mission-oriented wireless sensor network (WSN) scenario. Each delay constraint $k\in\mathcal{K}_s$ concerns a specific time interval. Overlap between such intervals, however, is allowed. In order to encode delay constraints of the form (3), for each source s, we introduce the *delay indicator matrix* $M_s=[(M_s)_{kt}]_{K_s\times T}$ as follows

$$(M_s)_{kt}=\begin{cases}\frac{1}{G_k^s} & \text{if } k\text{-th delay constraint of } s \text{ concerns } t,\\ 0 & \text{otherwise,}\end{cases}$$

where $G_k^s=\sum_{t\in\mathcal{T}}\mathbf{1}_{\{(M_s)_{kt}\neq 0\}}$. Now, we can write the k-th delay constraint of source s as

$$\sum_{t\in\mathcal{T}}(M_s)_{kt}\phi_{st}\leq d_{sk},$$

where d_{sk} is the average delay requirement of source s for its k's delay constraint. Note that the elements of every row of M_s add up to 1 and therefore, we may interpret the left hand side of the constraint above, like that of (3), as the end-to-end queueing delay of s averaged over time interval $\{t\in\mathcal{T}:(M_s)_{kt}=1\}$. Moreover, letting $\mathbf{d}_s=[d_{sk}]_{k\in\mathcal{K}_s}$ yields the following vector representation for delay constraints:

$$M_s\boldsymbol{\phi}_s\leq\mathbf{d}_s, \qquad \forall s\in\mathcal{S}. \tag{4}$$

These constraints constitute a set of $\sum_{s\in\mathcal{S}}K_s$ inequalities that are nonlinear in $\boldsymbol{\sigma}$.

1) An Illustrative Example: Mission-Oriented WSNs: To motivate the appropriateness of the model above, we next provide a practical application of this model for WSN [17], in which there are several coexisting applications (henceforth *missions*) overlaid on a WSN. Let us look at a surveillance application that employs various types of sensors such as *video*, *motion detector*, and *thermal sensors* to provide assistive ambient intelligence in e.g., disaster recovery environments.

The naïve approach is to require each sensor to periodically transmit the data at specific time intervals. Albeit simple to implement, this approach is inefficient as each mission might possess particular QoS requirement in terms of end-to-end delay. For instance, a video mission may demand for a long-time delay constraint to work efficiently. In contrast, the thermal mission may report the temperature periodically on a regular basis and thereby declares a short-term delay requirement at certain periods.

The network designer therefore needs to select network parameters properly to achieve the best efficiency. Besides other parameters, one could set $\mathcal{T}_\Delta=\mathcal{T}$ for the real-time video mission, as it records and streams data to the sink

$$L(X,\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\mu}) = \sum_{t\in\mathcal{T}}\sum_{s\in\mathcal{S}} U_{st}(x_{st}) - \sum_{t\in\mathcal{T}} \boldsymbol{\lambda}_t^{\mathsf{T}}\left(R_t X \mathbf{e}_t - \mathbf{c}_t + \boldsymbol{\sigma}_t\right) - \sum_{s\in\mathcal{S}} \boldsymbol{\mu}_s^{\mathsf{T}}\left(M_s\boldsymbol{\phi}_s - \mathbf{d}_s\right) \tag{5}$$
$$= \sum_{t\in\mathcal{T}}\sum_{s\in\mathcal{S}} \left(U_{st}(x_{st}) - \lambda^{st}x_{st}\right) - \sum_{t\in\mathcal{T}}\sum_{l\in\mathcal{L}} \left(\mu^{tl}D(\sigma_{tl}) + \lambda_{tl}\sigma_{tl}\right) + \sum_{t\in\mathcal{T}} \boldsymbol{\lambda}_t^{\mathsf{T}}\mathbf{c}_t + \sum_{s\in\mathcal{S}} \boldsymbol{\mu}_s^{\mathsf{T}}\mathbf{d}_s.$$

continuously. The value of $\mathcal{T}_\Delta$ has a periodic shape for the thermal sensor. Say, in the case of $T = 60$, we can define $\mathcal{T}_{\Delta_1} = \{1,2,3\}$, $\mathcal{T}_{\Delta_2} = \{21,22,23\}$, and $\mathcal{T}_{\Delta_3} = \{41,42,43\}$. In this respect, this sensor reports its data in 3 different steps as mentioned above.

D. Optimization Problem

We associate a utility function $U_{st}(.)$ to each source s at period t. Assumptions on the utility functions are:

A4. For every s and t, $U_{st}(.)$ is continuous, monotonically increasing, and twice differentiable.

A5. For every s and t, $-U_{st}(.)$ is κ_U-strongly convex.

We define the network utility $U(.)$ as the sum of all utilities over time horizon $\mathcal{T}$ and sources $\mathcal{S}$ as follows:

$$U(X) = \sum_{s\in\mathcal{S}}\sum_{t\in\mathcal{T}} U_{st}(x_{st}).$$

We cast the dynamic utility maximization problem as

$$\mathsf{P1}: \quad \max_{X\in\mathcal{X},\boldsymbol{\sigma}\geq 0} \quad U(X)$$

subject to:

$$R_t X \mathbf{e}_t + \boldsymbol{\sigma}_t \leq \mathbf{c}_t, \qquad \forall t \in \mathcal{T},$$
$$M_s\boldsymbol{\phi}_s \leq \mathbf{d}_s, \qquad \forall s \in \mathcal{S},$$
$$\phi_{st} = \sum_{l\in\mathcal{L}} (R_t)_{ls} D(\sigma_{tl}), \quad \forall s \in \mathcal{S}, \forall t \in \mathcal{T}.$$

First, we highlight that constraints of P1 constitute a compact set. Hence, at least one optimal solution exists. Furthermore, P1 is a strongly convex optimization problem. An immediate consequence of this property is the uniqueness of the optimal solution. We remark that P1 is non-separable due to coupled delay constraints. It's worth noting that in the lack of average delay constraints, problem P1 degenerates to DNUM problem of [10] without delivery contracts. In the above formulation, we address QoS requirements mainly through end-to-end delay constraints and thus avoid augmenting delivery contracts, i.e. linear constraints on source rates over $\mathcal{T}$. We stress, however, that the solution procedure below permits having delivery contracts as well. We further note that for the case of $T = 1$ and $K_s = 1, \forall s$, P1 reduces to problem formulation in [7] (for the case of rate allocation only).

IV. Optimal Rate Allocation Algorithm

In this section, we solve P1 and develop a distributed rate allocation algorithm. We note that strong duality [18] holds for P1 and hence we can solve it through its dual. We let $\boldsymbol{\lambda}_t = [\lambda_{tl}]_{l\in\mathcal{L}}$ and $\boldsymbol{\mu}_s = [\mu_{sk}]_{k\in\mathcal{K}_s}$ respectively denote the Lagrange multipliers (dual variables) associated to the capacity constraints for period t and average delay constraints for source s. Moreover, we introduce $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_t]_{t\in\mathcal{T}}$ and $\boldsymbol{\mu} = [\boldsymbol{\mu}_s]_{s\in\mathcal{S}}$. Now, we give the partial Lagrangian of P1 in (5), where

$$\lambda^{st} \triangleq \sum_{l\in\mathcal{L}} (R_t)_{ls}\lambda_{tl},$$
$$\mu^{tl} \triangleq \sum_{s\in\mathcal{S}}\sum_{k\in\mathcal{K}_s} (M_s)_{kt}(R_t)_{ls}\mu_{sk}.$$

To solve problem P1, we derive the dual function $g(\boldsymbol{\lambda},\boldsymbol{\mu})$ as follows:

$$g(\boldsymbol{\lambda},\boldsymbol{\mu}) = \max_{X\in\mathcal{X},\boldsymbol{\sigma}\geq 0} L(X,\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\mu})$$
$$= \max_{X\in\mathcal{X}} \sum_{t\in\mathcal{T}}\sum_{s\in\mathcal{S}} \left(U_{st}(x_{st}) - \lambda^{st}x_{st}\right)$$
$$+ \max_{\boldsymbol{\sigma}\geq 0} \sum_{t\in\mathcal{T}}\sum_{l\in\mathcal{L}} \left(\mu^{tl}D(\sigma_{tl}) + \lambda_{tl}\sigma_{tl}\right). \tag{6}$$

We establish the dual problem associated to P1 as [11]:

$$\mathsf{D1}: \quad \min_{\boldsymbol{\lambda}\geq 0,\boldsymbol{\mu}\geq 0} g(\boldsymbol{\lambda},\boldsymbol{\mu}).$$

Given $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, let $X^\star = [x^\star_{st}]_{T\times S}$ and $\boldsymbol{\sigma}^\star_t = [\sigma^\star_{tl}]_{l\in\mathcal{L}}$ be the maximizers of the problems in 6. The maximizers are stationary point of the Lagrangian. Therefore, through preliminary manipulations we get

$$x^\star_{st}(\boldsymbol{\lambda}) = \left[U'^{-1}_{st}(\lambda^{st})\right]_{\mathcal{X}_{st}}, \quad \forall s, \forall t,$$
$$\sigma^\star_{tl}(\boldsymbol{\lambda},\boldsymbol{\mu}) = \left[D'^{-1}\left(-\frac{\lambda_{tl}}{\mu^{tl}}\right)\right]^+, \quad \forall t, \forall l.$$

One consequence of strong convexity of P1 is that the dual function $g(\boldsymbol{\lambda},\boldsymbol{\mu})$ is differentiable in its domain. Hence, we can employ the *gradient projection method* [11] to solve D1. Using Danskin's Theorem [11], for dual variable update needed for gradient projection method we get

$$\lambda^{(j+1)}_{tl} = \left[\lambda^{(j)}_{tl} + \gamma\left(\sum_{s\in\mathcal{S}} (R_t)_{ls}x^{(j)}_{st} + \sigma^{(j+1)}_{tl} - c_{tl}\right)\right]^+,$$
$$\forall l \in \mathcal{L}, \forall t \in \mathcal{T},$$
$$\mu^{(j+1)}_{sk} = \left[\mu^{(j)}_{sk} + \zeta\left(\sum_{t\in\mathcal{T}}\sum_{l\in\mathcal{L}} (R_t)_{ls}D(\sigma^{(j)}_{tl}) - d_{sk}\right)\right]^+,$$
$$\forall s \in \mathcal{S}, \forall k \in \mathcal{K}_s,$$

where $x^{(j)}_{st} = x^\star_{st}(\boldsymbol{\lambda}^{(j)})$, $\sigma^{(j+1)}_{tl} = \sigma^\star_{tl}(\boldsymbol{\lambda}^{(j)},\boldsymbol{\mu}^{(j)})$, and $\gamma > 0$ and $\zeta > 0$ are sufficiently small step sizes. Note that proper selection of γ and ζ is crucial for guaranteeing the convergence of the iterative solution above.

Given appropriate γ and ζ, update equations for dual variables converge to minimizers of D1. Strong duality then guarantees that optimal values of D1 and P1 coincide and that $X^\star$ and $\boldsymbol{\sigma}^\star$ can be obtained accordingly. Next, we give a distributed iterative algorithm, named *Delay-Aware Dynamic Network Utility Maximization (DA-DNUM)*, that is based on

a distributed implementation of the above iterative solution. Since gradient-based algorithms are not finitely convergent, in DA-DNUM algorithm we introduce a parameter th to stop the iterative procedure. DA-DNUM algorithm relies on both the knowledge of network parameters in advance of time interval $\mathcal{T}$ and ability of explicit/implicit exchange of dual variables between sources and links (more precisely, between each source s and links on the path of s). The pseudo-code of DA-DNUM is shown as Algorithm 1. For the convergence analysis of DA-DNUM we refer to [19].

Algorithm 1: DA-DNUM Algorithm

Acquire network parameters for the next time horizon $\mathcal{T}$.
Initialize $X^0, \boldsymbol{\sigma}^0, \boldsymbol{\lambda}^0$, and $\boldsymbol{\mu}^0$.
while $\max_{s,l,t} \left\{ |x_{st}^{(j+1)} - x_{st}^{(j)}|, |\sigma_{tl}^{(j+1)} - \sigma_{tl}^{(j)}| \right\} \geq th$ **do**
  At each link l, for each period t, obtain $\mu^{tl,(j)}$ and update:
    $$\sigma_{tl}^{(j+1)} = \left[D'^{-1} \left(-\frac{\lambda_{tl}^{(j)}}{\mu^{tl,(j)}} \right) \right]^+$$
    $$\lambda_{tl}^{(j+1)} = \left[\lambda_{tl}^{(j)} + \gamma \left(\sum_{s \in \mathcal{S}} (R_t)_{ls} x_{st}^{(j)} + \sigma_{tl}^{(j+1)} - c_{tl} \right) \right]^+$$
  At each source s, for each period t, obtain $\lambda^{st,(j)}$ and compute:
    $$x_{st}^{(j+1)} = \left[U_{st}'^{-1} \left(\lambda^{st,(j)} \right) \right]_{\mathcal{X}_{st}}$$
    $$\mu_{sk}^{(j+1)} = \left[\mu_{sk}^{(j)} + \gamma \left(\sum_{t \in \mathcal{T}} \sum_{l \in \mathcal{L}} (R_t)_{ls} D(\sigma_{tl}^{(j)}) - d_{sk} \right) \right]^+$$
end

V. Experimental Results

This section is devoted to the experimental results. First, we concentrate on a tractable network topology to verify the correctness of DA-DNUM. Second, by describing two comparison scenarios, we investigate the performance and scalability of DA-DNUM.

A. Experiment 1: Simple and Tractable Topology

In order to facilitate detailed discussion of results, we have chosen a network with time-invariant routing and topology shown in Fig. 1. We set $T = 10$ and c_{1t} and c_{4t} are chosen uniformly at random from $[4, 6]$, and c_{2t} and c_{3t} are drawn uniformly at random from $[4, 10]$. We choose $U_{st}(x_{st}) = \log x_{st}$ for all s and t. Moreover, we assume that $D(z) = \frac{1}{z}$ for all links that represents M/M/1 queuing model. Delay indicator matrices and $\mathbf{d}_s, s \in \mathcal{S}$ are given below:

$$M_1 = \frac{1}{3} \times \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix},$$
$$M_2 = \frac{1}{6} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$
$$M_3 = \frac{1}{6} \times \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \end{bmatrix},$$
$$M_4 = \frac{1}{4} \times \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$
$$\mathbf{d}_1 = \begin{bmatrix} 2 & 1 \end{bmatrix}^\mathsf{T}, \quad d_2 = d_3 = 2, \quad d_4 = 2.5.$$

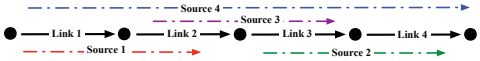


Fig. 1: Network Topology, Experiment 1

We remark that these delay indicator matrices imply that for $t = 9, 10$, P1 degenerates to DNUM [10] without delivery contracts, since there is no delay constraint in these periods.

Fig. 2(a) and Fig. 2(b) display the rate allocation result obtained from DA-DNUM algorithm with $\gamma = 0.01$ and $th = 0.01$. For the sake of comparison, Fig. 2(a) and Fig. 2(b) also show the rate allocation result of DNUM (without delivery contracts), which is obtained by solving P1 after removal of delay constraints. Fig. 2(a) shows final source rates of the two cases. As we expect, Fig. 2(a) exhibits the same values for both DA-DNUM and DNUM for $t = 9, 10$. By contrast, for $t = 1, \ldots, 8$ source rates obtained by DA-DNUM are lower than those provided by DNUM. This stems from existence of at least one delay constraint in any of these periods.

Finally, Fig. 2(b) shows link traffics, link margins, and the amount of under-utilized link capacities. Clearly, in periods $t = 9, 10$, all links possess zero link margins, since there is no delay constraint in these periods. On the other hand, for $t = 1, \ldots, 8$, positive values for link margin variables (for at least one link) evince that there is at least one active delay constraint imposed by the sources.

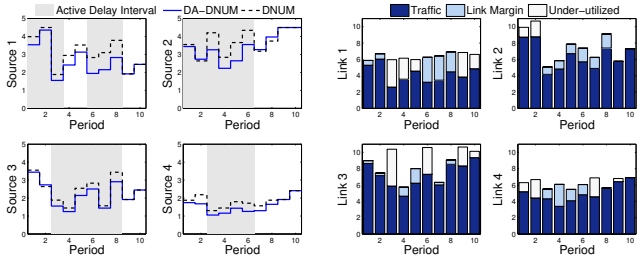
B. Experiment 2: Comparison Scenario

We next compare DA-DNUM with the algorithm proposed in [7] (by assuming fixed capacities) in a large-scale scenario. We remark that the algorithm proposed in [7] is based on the single-period version of NUM that is customized in delay-sensitive setting. Consequently, single-period NUM in algorithm of [7] persuades us to solve T separate problems for the entire $\mathcal{T}$. We consider a line topology with 200 links and 198 sources (Fig. 3) whose 200×198 (time-invariant) routing matrix is given in below:

$$R_t = \begin{bmatrix} 1 & 1 & 0 & 0 & \ldots & 0 & 0 \\ 1 & 1 & 1 & 0 & \ldots & 0 & 0 \\ 1 & 1 & 1 & 1 & \ldots & 0 & 0 \\ 1 & 1 & 1 & 1 & \ldots & 0 & 0 \\ 1 & 0 & 1 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & \ldots & 0 & 1 \end{bmatrix}, \forall t. \quad (7)$$

In addition, the other parameters are listed in Table I. To clearly exhibit the different behavior of DA-DNUM, we intentionally set up only 2 average delay constraints for source 1 (the source with all links on its path) and source 2 (the one that traverses through first 4 links).

To exhibit the flexibility of DA-DNUM, this experiment simply obliges a minimum rate demand as $x_{1,2}^{\min} = 5$. This means that the minimum rate requirement of source 1 at period 2 is 5. The aforementioned minimum rate demand is in conflict with the average delay requirement since the higher rate results in higher end-to-end delay according to the limited capacity of links. Nonetheless, DA-DNUM easily



(a) Optimal Source Rates

(b) Link Utilization

Fig. 2: Experiment 1

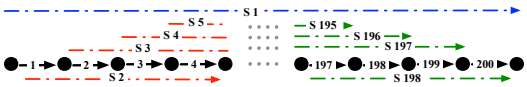


Fig. 3: Network Topology, Experiment 2

TABLE I: Parameters of Experiment 2

Parameter	Value
S	198
L	200
T	50
$c_{tl}, t \in \mathcal{T}, l \in \mathcal{L}$	[8,12] *kbps*
$k_s, s \in \{1,2\}$	1
$M_s, s \in \{1,2\}$	$[1/50]_{1\times 50}$
$k_s, s \in \{3,\ldots,198\}$	0
$M_s, s \in \{3,\ldots,198\}$	$[0]_{1\times 50}$
$\mathbf{d}_s, s \in \{1,2\}$	50

remedies this conflicting situation by assigning the declared minimum rate to s_1 at t_2, thus enduring a larger short-term delay (around 85 instead of $\mathbf{d}_1 = 50$). Thanks to supporting time-coupled system model, DA-DNUM allocates proper rates to this source in other periods, so as to maintain the average delay below 50. In contrast, the single-period algorithm of [7] fails for this scenario since the underlying NUM becomes infeasible. This experiment signifies the relatively wider set of feasible problems of DA-DNUM. One may construct several other feasible scenarios for DA-DNUM that are infeasible for the problem of [7].

VI. Conclusion

To ameliorate QoS experience in real-time networking applications in terms of guaranteeing fixed average end-to-end delay over long periods, we addressed a dynamic NUM problem with source-driven time-coupled constraints on average end-to-end delay. We proposed a delay-aware rate allocation algorithm as dual-based distributed solution of the formulated optimization problem. Our algorithm allocates source rates in a way that achieves the maximum network-wide utility aggregated over time interval while satisfying capacity and delay constraints. Numerical experiments exhibited that, compared to existing schemes DA-DNUM admits wider feasible scenarios along with higher resource utilization. This enhancement originated from multi-period problem setup that allows short-term delay fluctuations while keeps long-term value around the required one.

Acknowledgment

This work was supported by the University Grants Committee of the Hong Kong Special Administrative Region, China (Area of Excellence Grant Project No. AoE/E-02/08).

References

[1] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, 2007.

[2] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research society*, vol. 49, no. 3, pp. 237–252, 1998.

[3] S. H. Low and D. E. Lapsley, "Optimization flow control, i: Basic algorithm and convergence," *IEEE/ACM Transactions on Networking*, vol. 7, no. 6, pp. 861–874, 1999.

[4] Y. Li, A. Papachristodoulou, M. Chiang, and A. R. Calderbank, "Congestion control and its stability in networks with delay sensitive traffic," *Computer Networks*, vol. 55, no. 1, pp. 20–32, 2011.

[5] I. Hou and P. R. Kumar, "Utility maximization for delay constrained qos in wireless," in *Proc. IEEE INFOCOM*, 2010, pp. 1–9.

[6] B. M. Dogahe, M. N. Murthi, X. Fan, and K. Premaratne, "A distributed congestion and power control algorithm to achieve bounded average queuing delay in wireless networks," *Telecommunication Systems*, vol. 44, no. 3-4, pp. 307–320, Jan. 2010.

[7] F. Qiu, J. Bai, and Y. Xue, "Optimal rate allocation in wireless networks with delay constraints," *Ad Hoc Networks*, 2013.

[8] H. Xiong, R. Li, A. Eryilmaz, and E. Ekici, "Delay-aware cross-layer design for network utility maximization in multi-hop networks," *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 5, pp. 951–959, 2011.

[9] M. S. Talebi, A. Khonsari, and M. H. Hajiesmaili, "Utility-proportional bandwidth sharing for multimedia transmission supporting scalable video coding," *Computer Communications*, vol. 33, no. 13, pp. 1543–1556, 2010.

[10] N. Trichakis, A. Zymnis, and S. Boyd, "Dynamic network utility maximization with delivery contracts," in *IFAC World Congress*, 2008, pp. 2907–2912.

[11] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[12] J. Pongsajapan and S. H. Low, "Reverse engineering tcp/ip-like networks using delay-sensitive utility functions," in *Proc. IEEE INFOCOM*, 2007, pp. 418–426.

[13] S. Shenker, "Fundamental design issues for the future internet," *Selected Areas in Communications, IEEE Journal on*, vol. 13, no. 4, pp. 1176–1188, 1995.

[14] J. J. Jaramillo, R. Srikant, and L. Ying, "Scheduling for optimal rate allocation in ad hoc networks with heterogeneous delay constraints," *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 5, pp. 979–987, 2011.

[15] M. H. Hajiesmaili, A. Khonsari, A. Sehati, and M. S. Talebi, "Content-aware rate allocation for efficient video streaming via dynamic network utility maximization," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 2016–2027, 2012.

[16] M. Saad, A. Leon-Garcia, and W. Yu, "Optimal network rate allocation under end-to-end quality-of-service requirements," *Network and Service Management, IEEE Transactions on*, vol. 4, no. 3, pp. 40–49, 2007.

[17] S. Eswaran, A. Misra, F. Bergamaschi, and T. L. Porta, "Utility-based bandwidth adaptation in mission-oriented wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 8, no. 2, pp. 17–26, 2012.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[19] M. H. Hajiesmaili, M. S. Talebi, and A. Khonsari, "Utility-optimal dynamic rate allocation under average end-to-end delay requirements," *http://arxiv.org/abs/1509.03374*, 2015.