Computers and Operations Research 81 (2017) 67–77

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

Delay-constrained routing problems: Accurate scheduling models and admission control

Antonio Frangioni[a,*], Laura Galli[a], Giovanni Stea[b]

[a] Dipartimento di Informatica, Università di Pisa, Largo B. Pontecorvo 3, 56127 Pisa, Italy
[b] Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Largo Lucio Lazzarino 1, 56122 Pisa, Italy

ARTICLE INFO

Article history:
Received 29 December 2015
Revised 7 December 2016
Accepted 7 December 2016
Available online 9 December 2016

MSC:
90B18
90C35
90C90

Keywords:
Routing problems
Maximum delay constraints
Scheduling algorithms
Admission control
Second-order cone programs
Perspective reformulation

ABSTRACT

As shown in [1], the problem of routing a flow subject to a worst-case end-to-end delay constraint in a packed-based network can be formulated as a Mixed-Integer Second-Order Cone Program, and solved with general-p'urpose tools in real time on realistic instances. However, that result only holds for one particular class of packet schedulers, *Strictly Rate-Proportional* ones, and implicitly considering each link to be fully loaded, so that the *reserved rate* of a flow coincides with its *guaranteed rate*. These assumptions make latency expressions simpler, and enforce perfect isolation between flows, i.e., admitting a new flow cannot increase the delay of existing ones. Other commonplace schedulers both yield more complex latency formulæ and do not enforce flow isolation. Furthermore, the delay actually depends on the *guaranteed* rate of the flow, which can be significantly larger than the *reserved* rate if the network is unloaded. In this paper we extend the result to other classes of schedulers and to a more accurate representation of the latency, showing that, even when admission control needs to be factored in, the problem is still efficiently solvable for realistic instances, provided that the right modeling choices are made.

1. Introduction

The Internet already supports applications that require stringent guarantees on end-to-end delays (voice/video streaming, remote operation of industrial/medical tools, etc.). Obtaining *Quality of Service* (QoS) guarantees for a packet flow, such as a maximum delay, is thus a crucial problem, which is made nontrivial by the packed-based nature of the infrastructure. *QoS routing* is the practice of computing network paths where a suitable QoS can be guaranteed, which gives rise to Constrained Shortest Path (CSP) problems. CSPs having a single end-to-end constraint which is an additive or multiplicative *concave* function of per-link metrics admit polynomial solution algorithms, while CSPs with two or more constraints are $\mathcal{NP}$-hard (cf. [1] and the references therein). Due to the typically strict requirements on the time to deliver the solution in practice (say, some 10s or 100s of milliseconds), approximate approaches are normally employed to solve them (e.g., [2–4]). Furthermore, rather simplified network models have been traditionally employed where the relevant QoS parameters, say link delays, are considered statically known and additive. This neglects *queueing*, i.e., the delay due to the fact that the same link is shared by different flows, whose packets are transmitted sequentially. Queueing delays depend on the *packet schedulers* employed to arbitrate the flows.

A well-known paradigm for QoS scheduling is Generalized Processor Sharing (GPS) [5], that defines an ideal reference system which serves backlogged flows simultaneously at a rate proportional to their *weight*. If flow weights are chosen equal to their *reserved rates*, and their sum does not exceed the link capacity, then GPS guarantees that the flows' *guaranteed rates* will be at least as large as the reserved ones. This allows per-link and end-to-end Worst-case Delay (WCD) bounds to be computed if the traffic arrival rate at the source is constrained. Two practical implementations of GPS have been proposed, namely *Packet-by-packet Generalized Processor Sharing (PGPS)* [5] and *Worst-case Fair Weighted Fair Queueing (WF2Q)* [6]. Both exhibit tight guarantees on the *latency*, i.e., the worst-case scheduling delay at a link, which is—barring a small additive constant—inversely proportional to the guaranteed rate, thereby earning them the moniker of *Strictly Rate-Proportional* (SRP) schedulers. Since a flow's WCD depends on the guaranteed rates along its path, QoS routing problems with WCD constraints can easily be defined assuming SRP schedulers in the network. For instance, [7,8] show that the problem of finding a path with a pre-specified WCD is $\mathcal{NP}$-hard in general, unless the *same rate* is reserved at each link. Recently, [1] showed that,

* Corresponding author.
E-mail addresses: frangio@di.unipi.it (A. Frangioni), laura.galli@unipi.it (L. Galli), giovanni.stea@unipi.it (G. Stea).

http://dx.doi.org/10.1016/j.cor.2016.12.009

nonetheless, optimal solutions can be found in split-second times for realistic-sized networks even allowing different rates for each link, and that this leads to sizable performance gains in terms of flow blocking probability [9].

Unfortunately, the implementation of SRP schedulers is rather complex, which is a downside on high-speed links and/or with many simultaneous flows. In the last two decades, several schedulers have been devised which exhibit different trade-offs between latency and implementation cost, some having made their way into commercial hardware [10]. At one end of the spectrum we find GPS approximations based on *flow grouping* [11,12], and at the other end lie *frame-based* algorithms such as Deficit Round Robin [13] and its derivatives [10,14,15]. Both are simpler, but exhibit looser latency guarantees as well. To the best of our knowledge, no attempt has been made so far to devise QoS-routing schemes for these other schedulers. The only related work that we are aware of, [16], shows that non-uniform rate allocation *given a pre-specified routing plan* achieves better network utilization than uniform rate allocation in the presence of WCD constraints. This means that so far it has been impossible to estimate the impact of employing lower-complexity schedulers on the network performance (e.g., utilization or blocking probability).

Furthermore, all previous works—including [1,9]—have resorted to simplifying the latency formulæ by assuming that the *guaranteed* rate of a flow is equal to its *reserved* rate. This *bound assumption* is safe, in that the reserved rate is always no larger than the guaranteed one, but it leads to over-estimating the WCD experienced by a flow, and therefore to a more conservative resource allocation than necessary. To the best of our knowledge, the impact of the bound assumption on the network performance has not been investigated yet.

This paper provides a first, necessary step towards answering the above questions by formulating and solving the *Admissible Delay-Constrained Shortest Path* (ADCSP) problem: given the current state of the network, a set of link *reservation costs*, and a new flow to be routed between a given source and destination under a pre-specified WCD constraint, determine a feasible path and a feasible rate reservation on each link (if there exists one) minimizing the total reservation cost and ensuring that existing flows still satisfy their WCD constraints. We show that, for several classes of packet schedulers, the ADCSP problem can be formulated as a Mixed-Integer Second-Order Cone Problem (MI-SOCP) and solved by general-purpose tools in split-second times for realistically-sized networks. This paves the way to exploring the impact of employing different scheduling algorithms on network performance. We also show that, while distinguishing between *reserved* and *guaranteed* rates in the latency formulæ does increase the complexity of the models, the cost of doing so remains bearable, thus opening the way to studying the impact of these modeling choices, too, on network performance.

This paper is organized as follows. In Section 2 we present our system model and hypotheses. In Sections 3–5 we discuss models for the three main classes of latency formulæ—respectively, Strictly Rate-Proportional (and their Group-Based approximations), Weakly Rate-Proportional and Frame-Based ones—under different assumptions on the description of reserved and guaranteed rates. In Section 6 we report computational results which show the relative efficiency (and, partly, effectiveness) of the various models on real networks with realistic traffic data. Finally, in Section 7 we draw some conclusions.

2. System model

We are given a computer network represented by a directed graph $G = (N, A)$, with $n = |N|$ and $m = |A|$. Nodes are switching elements (e.g., routers), and arcs are the links interconnecting them. Henceforth, delays are in seconds, packet lengths are in bits, and rates and link speeds are in bits per second. Each node $i \in N$ is characterized by a fixed *node delay* n_i. Each arc $(i, j) \in A$ is characterized by a fixed *link delay* l_{ij}, a *physical link speed* w_{ij}, and the *maximum transmit unit* (MTU) L (assumed to be constant for simplicity). A set Q of flows is already present in the network. Each $q \in Q$ is characterized by a fixed path in G (which, for notational simplicity, we will denote by q as well), fixed reserved rates r^q_{ij} for all $(i, j) \in q$, an upper bound on the tolerable WCD—called its *deadline*—δ^q, and a *leaky-bucket arrival-curve constraint*. That is, if $F(t)$ denotes the number of bits of the flow injected at the source in $[0, t)$, $F(t+\tau) - F(t) \leq \sigma^q + \rho^q \tau$ has to hold for all t and $\tau \geq 0$, where σ^q and ρ^q are the *burst* and the *rate* of the flow, respectively.

We now introduce the *Admissible Delay-Constrained Shortest Path* (ADCSP) problem: given the current state Q of the network, the cost f_{ij} of reserving one unit of capacity on (i, j), and the data describing one "new" flow to be routed in G (its endpoints s and d, burst σ and rate ρ, and deadline δ), find one feasible $s-d$ path p and a feasible reservation of capacity at each of its arcs—if any exist—so that the flow can be routed along p and *both the new flow and all the existing ones* meet their deadline, at the minimum possible reservation cost for the new flow. ADCSP requires one to compute the WCD of a flow, which depends on several factors:

1. the selected *routing* for the flow, i.e., the $s-d$ path p in G;
2. the *reserved rate* $r_{ij} \in [0, w_{ij}]$ for each arc $(i, j) \in p$;
3. the latency guarantees of the schedulers employed to share the output links' bandwidth among the flows (for the sake of simplicity, we will always assume the schedulers to be the same at each link, but extending the models to non-uniform cases is obvious);
4. the paths and reserved rates of all the other flows $q \in Q$.

In the following, we will denote by $P(i, j) = \{\, q : (i, j) \in q \,\} \subseteq Q$ the set of existing paths (*not* counting the one just to be routed) traversing arc (i, j). We will also find it expedient to consider A partitioned into $A' \cup A''$, where A' contains the arcs (i, j) that are "empty" ($P(i, j) = \emptyset$) and A'' those that contain at least one flow.

While the natural decision variables of the problem are the reserved rates r_{ij} at each link, in general the WCD rather depends on the *guaranteed rate* g_{ij} obtained by the flow on each $(i, j) \in p$. For all the fair-queueing schedulers that we will examine, the guaranteed rate is at least as large as the reserved rate. In fact, under the assumption that the arc is not over-provisioned

$$r_{ij} \leq w_{ij} - \bar{r}_{ij}, \tag{1}$$

where $\bar{r}_{ij} = \sum_{q \in P(i,j)} r^q_{ij}$ (≥ 0) is the total reserved rate of all the *other* flows at link (i, j), the guaranteed rates are given by the expression

$$g_{ij} = (w_{ij} r_{ij}) / (\bar{r}_{ij} + r_{ij}). \tag{2}$$

It is easy to see that $g_{ij} = w_{ij}$ when $\bar{r}_{ij} = 0$ ($\equiv (i, j) \in A'$), i.e., the arc is "completely unloaded" and the new flow is the only one traversing it. Conversely, $g_{ij} = r_{ij}$ when $\bar{r}_{ij} + r_{ij} = w_{ij}$, i.e., the arc is "completely loaded". In order for the WCD to be finite, the minimum guaranteed rate among all links of p must be at least as large as the traffic injection rate of the flow, i.e.,

$$g_{ij} \geq \rho \qquad \forall (i, j) \in p. \tag{3}$$

If (3) is satisfied, the general form of the WCD of path p is

$$\frac{\sigma}{\min\{\, g_{ij} : (i, j) \in p \,\}} + \sum_{(i,j) \in p} \big(\theta_{ij} + l_{ij} + n_i \big), \tag{4}$$

where θ_{ij} is the *link latency* experienced by the flow on path p when traversing the arc (i, j), i.e. the maximum scheduling delay