On the Benefits of Traffic "Reprofiling" The Single Hop Case

Jiayi Song, Jiaming Qiu, *Member, IEEE,* Roch Guérin, *Fellow, ACM, IEEE,* and Henry Sariowan, *Member, IEEE*

Abstract

The need to guarantee hard delay bounds to traffic flows with deterministic traffic profiles, *e.g.,* token buckets, arises in a number of network settings. Of interest are solutions that offer such guarantees while minimizing network bandwidth. The paper explores a basic building block towards realizing such solutions, namely, a single hop configuration. The main results are in the form of optimal solutions for meeting local deadlines under schedulers of varying complexity and therefore cost. The results demonstrate how judiciously modifying flows' traffic profiles, *i.e., reprofiling* them, can help simple schedulers reduce the bandwidth they require, often performing nearly as well as more complex ones.

Index Terms

Latency, bandwidth, optimization, token bucket, scheduling.

I. Introduction

The provision of deterministic delay guarantees to traffic flows is emerging as an important requirement in increasingly diverse settings. They include automotive, avionics, and manufacturing applications, smart grids, and datacenters [1]–[7]. This is reflected in standards such as Time Sensitive Networking (TSN) and Deterministic Networking (DetNet) [8]–[11] and in the Service Level Objectives/Agreements (SLOs/SLAs) [12] of many service provider networks that are increasingly including latency targets, motivated in part by the rapid growth of edge computing offerings [13].

In such settings, the traffic eligible for latency guarantees is commonly controlled using a traffic regulator [14] in the form of a token bucket (r, b) that limits both the flow's long-term rate, r, and burstiness, b. A flow's token bucket parameters are typically determined using traces, and selected to ensure zero access delay [15]. The network's goal is then to ensure that the latency guarantees of all such rate-controlled flows are met, preferably with as little bandwidth as possible.

This is the environment this paper assumes, with a focus on a basic building block, namely, delivering latency guarantees on a *single link* (hop) with *the least amount of bandwidth*. The answer obviously depends on the type of scheduler controlling access to the link, and the paper considers schedulers of different levels of complexity. Of greater interest is whether, what the paper terms *reprofiling*, can be beneficial. Reprofiling amounts to modifying a flow's original (chosen by the user) token bucket parameters to make the flow "easier" to accommodate. This concept was explored in WorkloadCompactor [15] with one important difference, namely, the constraint that reprofiling should not introduce any delay. In contrast, our reprofiling solutions impose an added delay in exchange for smoother flows. This in turn calls for tighter network latency bounds to ensure that the original delay targets are still met. The outcome of this trade-off depends on the level of reprofiling applied as well as the type of scheduler in use. Investigating when and how it is positive in a single hop setting is the focus of this paper.

Specifically, the paper considers reprofiling of the form $(r, b) \xrightarrow{\text{reprofiling}} (r, b')$, where $b' \leq b$. In other words, we reduce the flow's burstiness to make it easier to handle. We note that more complex reprofiling solutions are possible. Our motivations for focusing on burst reduction are two-fold. First, we want to minimize any added complexity, and this reprofiling can be realized simply by modifying the burst parameter of the *existing* token bucket. Second, As shown in Appendix F, in simple configurations involving only two flows and a static priority scheduler, adjusting the burst size is sufficient to minimize the required bandwidth. These motivations notwithstanding, more complex reprofilers, *e.g.,* adding a second token bucket that controls the peak rate, can be of benefit in more general settings. We explore this extension in [16] in the multiple hops setting.

J. Qiu and R. Guérin are with the Computer Science and Engineering department at Washington University in St. Louis, Saint Louis, MO 63130, USA, e-mail: {qiujiaming,guerin}@wustl.edu.

J. Song was with the Computer Science and Engineering department at Washington University in St. Louis and is now with ByteDance, Mountain View, CA 94041, USA, e-mail: jiayisong@wustl.edu.

H. Sariowan is with Google, Mountain View, CA 94043, USA, e-mail hsariowan@google.com.

An early version of the paper was presented at the 33rd International Teletraffic Congress (ITC'33) in September 2021.

The work was supported by NSF grant CNS 2006530 and a gift from Google.

We note that the notion of reprofiling is closely tied to the definition of *greedy shapers* of [17, Section 1.5], with one important difference. Specifically, depending on the scheduler, a reprofiler can be either non-work-conserving, *i.e.,* as a (greedy) shaper, or work-conserving. The latter is only applicable when relying on dynamic priority schedulers such as earliest deadline first (edf) that can combine the local link deadline and the reprofiling delay when determining the order in which to send packets.

The paper makes the following contributions when it comes to meeting latency targets in the single-hop case with traffic profiles in the form of token buckets:

- Characterize the optimal (minimum bandwidth) solution, and show that a dynamic priority (edf) scheduler can realize it. The solution readily establishes that reprofiling yields no benefit with such a scheduler.
- Identify optimal reprofiling solutions for static priority and fifo schedulers, and demonstrate how they allow those schedulers to closely approximate the performance of the more complex edf scheduler across a range of scenarios.

For ease of exposition, the results are derived and presented assuming a fluid model, which, therefore, implies a preemptive behavior. As the discussion of [17, Section 1.1.1] highlights, extending the results to a packet setting is readily achievable from standard network calculus results. For illustration purposes, Appendix F derives a solution for a static priority scheduler under a packet-based model, but the results do not contribute further insight.

The paper is structured as follows. Section II introduces our traffic model and optimization framework. The next three sections present optimal solutions for schedulers of different complexity. Section III considers a general, dynamic priority scheduler, while Sections IV and V assume simpler static priority and fifo schedulers. For the latter two, the benefits of reprofiling flows are also explored. Section VI quantifies performance for each scheduler, starting with two-flow configurations that help build intuition for the results, before considering more general multi-flow scenarios. Section VII reviews related works, while Section VIII summarizes the paper's findings and their relevance to the multi-hop extension of [16]. Proofs and ancillary results are relegated to appendices.

II. Model Formulation

Consider the configuration of Fig. 1 with n flows sharing a common link[1] of rate R. The traffic generated by flow i is rate-controlled using a two-parameter token bucket (r_i, b_i) [14], its *traffic profile*, where r_i is the token rate and b_i the bucket size. Flow i also has a local packet-level deadline d_i, where w.l.o.g. we assume $d_1 > d_2 > \ldots > d_n$ with $d_1 < \infty$. Our goal is to meet the deadlines of all n flows with the lowest possible link bandwidth R. In doing so, we further assume *greedy sources* [17, Proposition 1.2.5] that fully realize the arrival curve associated with their token bucket.

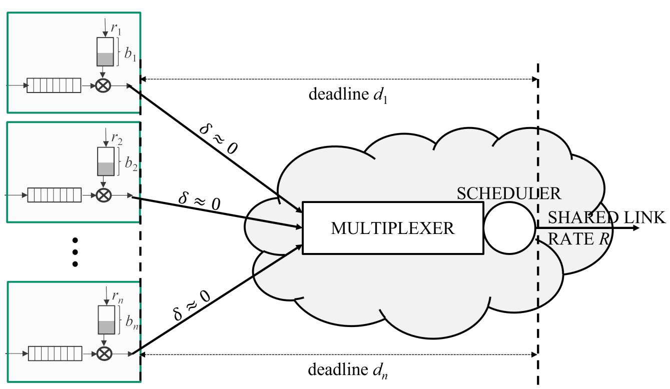


Fig. 1: A typical one-hop configuration with n flows.

In this setting, let $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$, $\boldsymbol{b} = (b_1, b_2, \ldots, b_n)$, and $\boldsymbol{d} = (d_1, d_2, \ldots, d_n)$ be the vectors of rates, burst sizes, and deadlines of the flows sharing the link, respectively, and let $D_i^*(\boldsymbol{r}, \boldsymbol{b}, R)$ denote flow i's *worst-case* delay (queueing+transmission). Our bandwidth minimization problem can then be formulated as an optimization of the form:

$$\begin{array}{ll} \textbf{OPT\_}\square & \min R \\ \text{s.t} & D_i^*(\boldsymbol{r}, \boldsymbol{b}, R) \le d_i, \quad \forall i, 1 \le i \le n \end{array}$$

where R is the optimization variable and ❑ denotes the scheduler type, which for notational simplicity has been omitted in the expression of $D_i^*(\boldsymbol{r}, \boldsymbol{b}, R)$. As mentioned in Section I, one of our goals is to evaluate the trade-off between (bandwidth) efficiency and complexity across different schedulers.

Another goal is to investigate the potential benefits of *reprofiling* flows prior to forwarding their traffic to the scheduler. Reprofiling amounts to applying a different, typically "smaller", traffic profile to each flow before forwarding them to the

[1] For simplicity, we assume that enough buffering is available and that the link capacity is such that the system is stable and lossless.

scheduler. This can[2] introduce an up-front reprofiling delay, but may lower the bandwidth required to meet overall latency goals if it makes flows "easier" to handle.

More formally, given a scheduler "❑" and n flows sharing a link, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, the goal of reprofiling is to identify smaller burst sizes $b'_i \leq b_i, 1 \leq i \leq n$, that minimize the link bandwidth R needed to meet the flows' deadlines, inclusive of any resulting reprofiling delay (the smaller burst b'_i introduces a reprofiling delay of $\frac{b_i - b'_i}{r_i}$). We note that we restrict reprofiling options to only reducing the burst size, rather than also considering adding a "peak rate" shaper. This is in part to simplify the resulting optimization **OPT_R❑**, and also because, as shown in Appendix F, this is sufficient in simple configurations with only two flows. This translates into a modified optimization problem **OPT_R❑** of the form

$$\begin{aligned} &\textbf{OPT\_R❑} \quad && \min_{\boldsymbol{b'}} R \\ &\text{s.t} && D^*_i(\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{b'}, R) \leq d_i, \quad \forall i, 1 \leq i \leq n \end{aligned}$$

where R and $\boldsymbol{b'}$ are the optimization variables. The latter denotes the vectors of updated (reprofiled) burst sizes of the n flows, and $D^*_i(\boldsymbol{r}, \boldsymbol{b}, \boldsymbol{b'}, R), 1 \leq i \leq n$, are the worst case delays, accounting for reprofiling delays, of the n flows under scheduler ❑ and a link bandwidth of R. The optimization explores the extent to which making flows smoother (smaller bursts) can facilitate meeting their delay targets with less bandwidth in spite of the access delay that reprofiling adds.

The next three sections explore solutions to **OPT_❑** (and **OPT_R❑**) for different schedulers, namely, dynamic priority, static priority, and fifo (**OPT_DP, OPT_SP**, and **OPT_F**).

III. Dynamic Priorities

We start with the most powerful but most complex scheduler, dynamic priorities, with priorities derived from service curves assigned to flows as a function of their profile (deadline and traffic envelope). We first solve **OPT_DP** by characterizing the service curves that achieve the lowest bandwidth while meeting all deadlines in the absence of any reprofiling.

To derive the result, we first specify a service-curve assignment Γ_{sc} that satisfies all deadlines, identify the minimum link bandwidth R^* required to realize Γ_{sc}, and show that any scheduler requires at least R^*. We then show that an earliest deadline first scheduler realizes Γ_{sc} and, therefore, meets all the flow deadlines under R^*. e note that this then implies that reprofiling is of no benefit when an edf scheduler is available.

Proposition 1. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has a traffic profile (r_i, b_i) and a deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Consider a service-curve assignment Γ_{sc} that allocates flow i a service curve of*

$$SC_i(t) = \begin{cases} 0 & \text{when } t < d_i, \\ b_i + r_i(t - d_i) & \text{otherwise}. \end{cases} \tag{1}$$

Then

1) *For any flow $i, 1 \leq i \leq n$, $SC_i(t)$ ensures a worst-case end-to-end delay no larger than d_i.*
2) *Realizing Γ_{sc} requires a link bandwidth of at least*

$$R^* = \max_{1 \leq h \leq n} \left\{ \sum_{i=1}^{n} r_i, \frac{\sum_{i=h}^{n} b_i + r_i(d_h - d_i)}{d_h} \right\}. \tag{2}$$

3) *Any scheduling mechanism capable of meeting all the flows' deadlines requires a bandwidth of at least R^*.*

The proof of Proposition 1 is in Appendix B-A. The optimality of Γ_{sc} is intuitive. Recall that a service curve is a lower bound on the service received by a flow. Eq. (1) assigns service to a flow at a rate exactly equal to its input rate, but delayed by its deadline, *i.e.,* provided at the latest possible time. Conversely, any mechanism $\widehat{\Gamma}$ that meets all flows' deadlines must by time t have provided flow i a cumulative service at least equal to the amount of data that flow i may have generated by time $t - d_i$, which is exactly $SC_i(t)$. Hence the mechanism must offer flow i a service curve $\widehat{SC}_i(t) \geq SC_i(t), \forall t$.

Next, we identify at least one mechanism capable of realizing the services curves of Eq. (1) under R^*, and consequently providing a solution to **OPT_DP** for schedulers that support dynamic priorities.

Proposition 2. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. The earliest deadline first (edf) scheduler realizes Γ_{sc} under a link bandwidth of R^*.*

The proof of Proposition 2 is in Appendix B-B. We note that the optimality of edf is intuitive, as minimizing the required bandwidth is the dual problem to maximizing the schedulable region for which edf's optimality is known [18].

[2]When the reprofiler operates in a non-work-conserving manner.

As previously mentioned and as the next proposition formally states, reprofiling does not reduce the minimum required bandwidth R^* of Eq. (2). Consequently, it affords no benefits with edf schedulers capable of meeting the deadlines under R^*. This is expected given the optimality of edf schedulers.

Proposition 3. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Reprofiling flows will not decrease the minimum bandwidth required to meet the flows' deadlines.*

The proof is in Appendix B-C.

Note that Γ_{sc} specifies a non-linear (piece-wise-linear) service curve for each flow. Given the popularity and simplicity of linear service curves, *i.e.,* rate-based schedulers, it is tempting to investigate whether such schedulers, *e.g.,* GPS [19], could be used instead. Unfortunately, it is easy to find scenarios where linear service curves perform worse.

Consider a link shared by two flows with traffic profiles $(r_1, b_1) = (1, 45)$ and $(r_2, b_2) = (1, 5)$, and deadlines $d_1 = 10$ and $d_2 = 1$. A rate-based scheduler must allocate a bandwidth of $\max\left\{\frac{b}{d}, r\right\}$ to a flow with traffic profile (r, b) to meet its deadline d of. Applying this to flow 2 that has the tighter deadline calls for a bandwidth of 5 to meet its deadline. After 1.25 units of time (the time to clear the initial burst of 5 and the additional data that accumulated during its transmission), flow 2's bandwidth usage drops down to $r_2 = 1$. The remaining 4 units then become available to flow 1. This means that the initial dedicated bandwidth needed by flow 1 to meet its deadline of 10 given its burst size of $b_1 = 45$ is simply its token rate $r_1 = 1$[3], for a total network bandwidth of 6 units. In contrast, Eq. (2) tells us that Γ_{sc}, only requires a bandwidth of $R^* = 5.9$.

The next two sections consider simpler static priority and fifo schedulers, and quantify the bandwidth they require to meet flows' deadlines. Both schedulers are considered either alone or with "reprofilers" that first modify the flows' traffic profiles before they are allowed to access the scheduler.

IV. Static Priorities

Though edf schedulers are efficient and increasingly realizable [20]–[22], they are expensive and may not be practical in all environments. It is, therefore, of interest to explore simpler alternatives while quantifying the trade-off they entail between efficacy and complexity. For that purpose, we consider next a static priority scheduler where each flow is assigned a fixed priority as a function of its deadline.

As before, we consider n flows with traffic profiles (r_i, b_i) and deadlines $d_i, 1 \leq i \leq n$, sharing a common link. The question we first address is how to assign (static) priorities to each flow given their deadlines and **OPT_SP**'s goal of minimizing link bandwidth? The next proposition offers a partial and somewhat intuitive answer to this question by establishing that the minimum link bandwidth can be achieved by giving flows with shorter deadlines a higher priority. Formally,

Proposition 4. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Under a static-priority scheduler, there exists an assignment of flows to priorities that minimizes link bandwidth while meeting all flows deadlines such that flow i is assigned a priority strictly greater than that of flow j only if $d_i < d_j$.*

The proof is in Appendix C-A. We note that while Proposition 4 states that link bandwidth can be minimized by assigning flows to priorities in the order of their deadline, it neither rules out other mappings nor does it imply that flows with different deadlines should always be mapped to distinct priorities. For example, large enough deadlines can all be met by a link bandwidth equal to the sum of the flows' average rates, *i.e.,* $R^* = \sum_{i=1}^n r_i$. In this case, priorities and their ordering are irrelevant. More generally, grouping flows with different deadlines in the same priority class can often result in a lower bandwidth than mapping them to distinct priority classes[4]. Nevertheless, motivated by Proposition 4, we propose a simple assignment rule that strictly maps lower deadline flows to higher priorities, and evaluate its performance.

A. Static Priorities without Reprofiling

From [17, Proposition 1.3.4] we know that when n flows with traffic profiles $(r_i, b_i), 1 \leq i \leq n$, share a link of bandwidth $R \geq \sum_{i=1}^n r_i$ with flow i assigned to priority i (priority n is the highest), then, under a static-priority scheduler, the worst case delay of flow h is upper-bounded by $\frac{\sum_{i=h}^n b_i}{R - \sum_{i=h+1}^n r_i}$ (recall that under our notation, priority n is the highest). As a result, the minimum link bandwidth $\widetilde{R}^*$ to ensure that flow h's deadline d_h is met for all h, *i.e.,* solving **OPT_SP**, is given by:

$$\widetilde{R}^* = \max_{1 \leq h \leq n} \left\{ \sum_{i=1}^n r_i, \frac{\sum_{i=h}^n b_i}{d_h} + \sum_{i=h+1}^n r_i \right\} \tag{3}$$

[3]Clearing the burst of flow 1 by its deadline $d_1 = 10$ calls for a bandwidth x such that $45 - \frac{5}{4}x - (x+4)\left(10 - \frac{5}{4}\right) \leq 0$, which yields $x \geq 1$.

[4]We illustrate this in Appendix E for the case of two flows sharing a static priority scheduler.

Towards evaluating the performance of a static priority scheduler versus that of an edf scheduler, we compare $\widetilde{R}^*$ with R^* through their relative difference, *i.e.,* $\frac{\widetilde{R}^*-R^*}{R^*}$. For ease of comparison, we rewrite R^* as

$$R^* = \max_{1\leq h\leq n}\left\{\sum_{i=1}^{n} r_i, \frac{\sum_{i=h}^{n} b_i}{d_h} + \sum_{i=h+1}^{n} r_i\left(1-\frac{d_i}{d_h}\right)\right\} \tag{4}$$

Comparing Eqs. (3) and (4) shows that $R^* = \widetilde{R}^*$ iff $\widetilde{R}^* = \sum_{i=1}^{n} r_i$, *i.e.,* $\frac{\sum_{i=h}^{n} b_i}{d_h} \leq \sum_{i=1}^{h} r_i, \forall\ 1 \leq h \leq n$. In other words, static priority and edf schedulers perform equally well (yield the same minimum bandwidth), when flow bursts are small and deadlines relatively large so that they can be met with a link bandwidth equal to the sum of the token rates. However, when $\widetilde{R}^* \neq \sum_{i=1}^{n} r_i$, a static priority scheduler can require a much larger bandwidth.

Consider a scenario where R^* is achieved at h^*, *i.e.,* $R^* = \frac{\sum_{i=h^*}^{n} b_i}{d_{h^*}} + \sum_{i=h^*+1}^{n} r_i\left(1-\frac{d_i}{d_{h^*}}\right)$. Though $\widetilde{R}^*$ may not be realized at the same h^* value, this still provides a lower bound for $\widetilde{R}^*$, namely, $\widetilde{R}^* \geq \frac{\sum_{i=h^*}^{n} b_i}{d_{h^*}} + \sum_{i=h^*+1}^{n} r_i$. Thus, the relative difference between $\widetilde{R}^*$ and R^* is no less than

$$\begin{aligned}&\frac{\frac{\sum_{i=h^*}^{n} b_i}{d_{h^*}} + \sum_{i=h^*+1}^{n} r_i}{\frac{\sum_{i=h^*}^{n} b_i}{d_{h^*}} + \sum_{i=h^*+1}^{n} r_i\left(1-\frac{d_i}{d_{h^*}}\right)} - 1\\ &= \frac{\sum_{i=h^*+1}^{n} d_i r_i}{\sum_{i=h^*}^{n} b_i\ + \sum_{i=h^*+1}^{n} r_i\left(d_{h^*}-d_i\right)}\end{aligned} \tag{5}$$

As the right-hand-side of Eq. (5) increases with d_i for all $i \geq h^*$, it is maximized for $d_i = d_{h^*} - \epsilon_i, \forall i > h^*$, for arbitrarily small $\epsilon_{h^*+1} < \ldots < \epsilon_n$, so that its supremum is equal to $\frac{\sum_{i=h^*+1}^{n} r_i d_{h^*}}{\sum_{i=h^*}^{n} b_i}$. Note that this is intuitive, as when flows have arbitrarily close deadlines, they should receive equal service shares, which is in direct conflict with a strict priority ordering.

Under certain flow profiles, the above supremum can be large. In a two-flow scenario, basic algebraic manipulations give a supremum of $\frac{r_2}{r_1+r_2}$, which is achieved at $d_2 = d_1 = \frac{b_2+b_1}{r_1+r_2}$. Since $\frac{r_2}{r_1+r_2} \to 1$ as $\frac{r_1}{r_2} \to 0$, the optimal static priority scheduler in the two-flow case could require twice as much bandwidth as the optimal edf scheduler.

B. Static Priorities with Reprofiling

Static priorities can require significantly more bandwidth than R^* mostly because they are a rather blunt instrument when it comes to fine-tuning the allocation of transmission opportunities as a function of packet deadlines. In particular, they often result in some flows experiencing a delay much lower than their target deadline.

This is intrinsic to the static structure of the scheduler and to our choice of an assignment that maps distinct deadlines to different priorities, but can be mitigated by anticipating and leveraging the "slack" in the delay of some flows. One such option is to use this slack towards reprofiling those flows, *i.e.,* make them "smoother". Of interest then, is how to reprofile flows to maximize any resulting link bandwidth reduction?

Consider the trivial example of a single link shared by two flows with traffic profiles $(r_1, b_1) = (1, 5)$ and $(r_2, b_2) = (4, 5)$ and deadlines $d_1 = 1.4, d_2 = 1.25$. A strict static-priority scheduler requires a bandwidth $\widetilde{R}^* = 11.14$. Assume next that we reprofile flow 2 to $(r_2, b_2') = (4, 0)$ before it enters the scheduler. The added reprofiling delay of $(b_2 - b_2')/r_2 = 1.25$ reduces the scheduling delay budget down to 0, but eliminates all burstiness. As a result, we only need a bandwidth of 7.57 (under a fluid model) to meet both flows' deadlines (a bandwidth of $4 = r_2$ is still consumed by flow 2, but the remaining 3.57 is sufficient to allow flow 1 to meet its deadline). In other words, reprofiling flow 2 yields a bandwidth decrease of more than 30%. This simple example illustrates the benefits that judicious reprofiling can afford.

The next few propositions characterize the optimal reprofiling solution and the resulting bandwidth gains for a static priority scheduler and a set of flows and deadlines. We first derive expressions for flows' reprofiling and scheduling delays under static priorities, before obtaining the optimal reprofiling solution and the resulting minimum link bandwidth $\widetilde{R}_R^*$.

Specifically, given n flows with initial traffic profiles $(r_i, b_i), 1 \leq i \leq n$, deadlines $d_1 > d_2 > \ldots > d_n$, a reprofiling solution $(r_i, b_i'), 1 \leq i \leq n$, and a link of bandwidth R, Proposition 5 characterizes the worst case delay (reprofiling plus scheduling) of each flow, when a static priority scheduler assigns flow i priority i (shorter deadlines have higher priority). The result is used to formulate an optimization problem, **OPT_RSP**, that seeks to minimize the link bandwidth required to meet individual flows' deadlines. The variables of the optimization are the reprofiling solution and the link bandwidth. Proposition 7 characterizes the minimum bandwidth $\widetilde{R}_R^*$ that **OPT_RSP** can achieve, while Proposition 8 provides the optimal reprofiling solution.

Let $\boldsymbol{b}' = (b_1', b_2', b_3', ..., b_n')$ be the vector of reprofiled flow bursts, with $B_i' = \sum_{j=i}^{n} b_j'$ and $R_i = \sum_{j=i}^{n} r_j$, the sum of the reprofiled bursts and rates of flows with priority greater than or equal to $i, 1 \leq i \leq n$, where $B_i' = 0$ and $R_i = 0$ for $i > n$. Flow i's worst-case end-to-end delay is characterized next.

Proposition 5. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i). Assume a static priority scheduler that assigns flow i a priority of i, where priority n is the highest priority, and reprofiles flow i to (r_i, b'_i), where $0 \leq b'_i \leq b_i$. Given a link bandwidth of $R \geq \sum_{j=1}^{n} r_j$, the worst-case delay for flow i is*

$$D_i^* = \max\left\{\frac{b_i + B'_{i+1}}{R - R_{i+1}},\ \frac{b_i - b'_i}{r_i} + \frac{B'_{i+1}}{R - R_{i+1}}\right\}. \tag{6}$$

The proof is in Appendix C-B. Note that Eq. (6) states that flow i's worst-case delay is realized by the last bit of its burst. The two terms of Eq. (6) capture the cases when this bit arrives before or after the end of flow i's last busy period at the link, respectively, as this determines the extent to which it is affected by the reprofiling delay.

Observe also that D_i^* is independent of b'_1 for $2 \leq i \leq n$, and decreases with b'_1 when $i = 1$. This is intuitive as flow 1 has the lowest priority so that reprofiling it can neither decrease the worst-case end-to-end delay of other flows, nor consequently reduce the minimum link bandwidth required to meet specific deadlines for each flow. Formally,

Corollary 6. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > \ldots > d_n$ and $d_1 < \infty$. Assume a static priority scheduler that assigns flow i a priority of i, where priority n is the highest priority, and reprofiles flow i to (r_i, b'_i), where $0 \leq b'_i \leq b_i$. Given a link bandwidth of $R \geq \sum_{j=1}^{n} r_j$, reprofiling flow 1 cannot reduce the minimum required bandwidth.*

Combining Proposition 5 and Corollary 6 with **OPT_SP** gives the following optimization **OPT_RSP** for a link shared by n flows and relying on a static priority scheduler preceded by reprofiling. Note that since the minimum link bandwidth needs to satisfy $R \geq \sum_{i=1}^{n} r_i$, combining this condition with R_i's definition gives $\sum_{i=1}^{n} r_i = R_1 \leq R$.

$$\textbf{OPT\_RSP} \quad \min_{\boldsymbol{b'}} R \quad \text{s.t}$$
$$\max\left\{\frac{b_i + B'_{i+1}}{R - R_{i+1}}, \frac{b_i - b'_i}{r_i} + \frac{B'_{i+1}}{R - R_{i+1}}\right\} \leq d_i,\ \forall\ 1 \leq i \leq n,$$
$$R_1 \leq R, \quad b'_1 = b_1, \quad 0 \leq b'_i \leq b_i,\ \forall\ 2 \leq i \leq n.$$

The solution of **OPT_RSP** is characterized in Propositions 7 and 8 whose proofs are in Appendix C-C. Proposition 7 gives the optimal bandwidth $\widetilde{R}_R^*$ based only on flow profiles, and while it is too complex to yield a closed-form expression, it offers a feasible numerical procedure to compute $\widetilde{R}_R^*$.

Proposition 7. *For $1 \leq i \leq n$, denote $H_i = b_i - d_i r_i$, $\Pi_i(R) = \frac{r_i + R - R_{i+1}}{R - R_{i+1}}$ and $V_i(R) = d_i(R - R_{i+1}) - b_i$. Define $\mathbb{S}_1(R) = \{V_1(R)\}$, and $\mathbb{S}_i(R) = \mathbb{S}_{i-1}(R) \bigcup \{V_i(R)\} \bigcup \left\{\frac{s - H_i}{\Pi_i(R)} \mid s \in \mathbb{S}_{i-1}(R)\right\}$ for $2 \leq i \leq n$. Then we have $\widetilde{R}_R^* = \max\{R_1, \inf\{R \mid \forall s \in \mathbb{S}_n(R), s \geq 0\}\}$.*

Computing $\widetilde{R}_R^*$ requires solving polynomial inequalities of degree $(n-1)$, so that a closed-form expression is not feasible except for small n. However, as $\mathbb{S}_i(R)$ relies only on flow profiles and $\mathbb{S}_j(R)$, $\forall j < i$, we can recursively construct $\mathbb{S}_n(R)$ from $\mathbb{S}_1(R)$. Hence, since $R_1 \leq \widetilde{R}_R^* \leq \widetilde{R}^*$, we can use a binary search to compute $\widetilde{R}_R^*$ from the relation $\widetilde{R}_R^* = \max\{R_1, \inf\{R \mid \forall s \in \mathbb{S}_n(R), s \geq 0\}\}$ in Proposition 7.

Next, Proposition 8 gives a constructive procedure to obtain the optimal reprofiling burst sizes $\boldsymbol{b}'^*$ given $\widetilde{R}_R^*$ and the original flow profiles.

Proposition 8. *The optimal reprofiling solution $\boldsymbol{b}'^*$ satisfies*

$$b_i'^* = \begin{cases} \max\{0, b_n - r_n d_n\}, & i = n; \\ \max\left\{0, b_i - r_i d_i + \dfrac{r_i B_{i+1}'^*}{\widetilde{R}_R^* - R_{i+1}}\right\}, & 2 \leq i \leq n-1. \end{cases} \tag{7}$$

where we recall that $b_1'^* = b_1$ and $B_i'^* = \sum_{j=i}^{n} b_j'^*$.

Note that the optimal reprofiling burst size $b_i'^*$ of flow $i, 1 < i < n$ relies only on the optimal link bandwidth $\widetilde{R}_R^*$ and the reprofiling burst sizes of higher priority flows. Hence, we can recursively characterize $b_i'^*$ from $b_n'^*$ given $\widetilde{R}_R^*$.

V. Basic FIFO with Reprofiling

In this section, we consider a simple first-in-first-out (fifo) scheduler that serves data in the order in which it arrives. For conciseness and given the benefits of reprofiling demonstrated in Section IV-B, we directly assume that flows are reprofiled prior to being scheduled. Considering again a link shared by n flows with traffic profiles $(r_i, b_i), 1 \leq i \leq n$, and deadlines $d_1 > d_2 > \ldots > d_n$, our goal is to find a reprofiling solution (r_i, b'_i), $1 \leq i \leq n$, to minimize the link bandwidth required to meet the flows' deadlines.

Towards answering this question, we first proceed to characterize the worst case delay across n flows sharing a link of bandwidth R equipped with a fifo scheduler when the flows have initial traffic profiles $(r_i, b_i), 1 \leq i \leq n$, and are reprofiled to $(r_i, b'_i), 1 \le i \le n$, prior to being scheduled. Using this result, we then identify the reprofiled burst sizes $b'_i, 1 \le i \le n$, that minimize the link bandwidth required to ensure that all deadlines $d_1 > d_2 > \ldots > d_n$, and $d_1 < \infty$ are met. As with other configurations, we only state the results with proofs relegated to Appendix D.

Proposition 9. *Consider a system with n token bucket controlled flows with traffic profiles $(r_i, b_i), 1 \le i \le n$, sharing a fifo link with bandwidth $R \ge R_1 = \sum_{j=1}^n r_j$. Assume that the system reprofiles flow i to (r_i, b'_i). The worst-case delay for flow i is then*

$$\widehat{D}_i^* = \max\left\{\frac{b_i - b'_i}{r_i} + \frac{\sum_{j\neq i} b'_j}{R}, \frac{\sum_{j=1}^n b'_j}{R} + \frac{(b_i - b'_i)R_1}{r_i R}\right\}. \tag{8}$$

The proof of Proposition 9 is in Appendix D-A.

With the result of Proposition 9 in hand, we can formulate a corresponding optimization problem, **OPT_RF**, for computing the optimal reprofiling solution that minimizes the link bandwidth required to meet the deadlines $d_1 > d_2 > \ldots > d_n$, and $d_1 < \infty$ of the n flows. Specifically, combining Proposition 9 with **OPT_F** gives the following optimization **OPT_RF** for a link shared by n flows when relying on a fifo scheduler preceded by reprofiling. As before, $\sum_{i=1}^n r_i = R_1 \le R$.

$$\begin{aligned} &\textbf{OPT\_RF} \quad \min_{\boldsymbol{b}'} R \quad \text{s.t} \quad \forall\, 1 \le i \le n \\ &\max\left\{\frac{b_i - b'_i}{r_i} + \frac{\sum_{j\neq i} b'_j}{R}, \frac{\sum_{j=1}^n b'_j}{R} + \frac{(b_i - b'_i)R_1}{r_i R}\right\} \le d_i, \\ &R_1 \le R, \quad 0 \le b'_i \le b_i, \forall\, 1 \le i \le n. \end{aligned} \tag{9}$$

The solution of **OPT_RF** is characterized in Propositions 10 and 11 with proofs in Appendix D-B. As with a static priority scheduler, Proposition 10 gives a numerical procedure to compute the optimal bandwidth $\widehat{R}_R^*$ given the flows' profiles, while Proposition 11 gives the optimal reprofiling solution $\widehat{\boldsymbol{b}}'^*$ given $\widehat{R}_R^*$ and the original flows' profiles.

Proposition 10. *For $1 \le i \le n$, define $H_i = b_i - d_i r_i$, $\widehat{B}_i = \sum_{j=1}^i b_j$, and $\mathbb{Z}_i = \{1 \le j \le i \mid j \in \mathbb{Z}\}$. Denote*

$$X_F(R) = \max_{\substack{P_1, P_2 \subseteq \mathbb{Z}_n,\\ P_2 \neq \mathbb{Z}_n, P_1 \bigcap P_2 = \emptyset}} \frac{\sum_{i\in P_1} \frac{RH_i}{R+r_i} + \sum_{i\in P_2}\left(b_i - \frac{r_i d_i R}{R_1}\right)}{1 - \sum_{i\in P_1}\frac{r_i}{R+r_i} - \sum_{i\in P_2}\frac{r_i}{R_1}}$$

and

$$Y_F(R) = \min_{1\le i\le n-1}\left\{\widehat{B}_n, Rd_n, \min_{\substack{P_1, P_2 \subseteq \mathbb{Z}_i,\\ P_1 \bigcap P_2 = \emptyset,\\ P_1 \bigcup P_2 \neq \emptyset}}\left\{\frac{\widehat{B}_i - \sum_{j\in P_1}\frac{RH_j}{R+r_j} - \sum_{j\in P_2}\left(b_j - \frac{r_j d_j R}{R_1}\right)}{\sum_{j\in P_1}\frac{r_j}{R+r_j} + \sum_{j\in P_2}\frac{r_j}{R_1}}\right\}\right\}.$$

*Then the optimal solution for **OPT_RF** is*

$$\widehat{R}_R^* = \max\left\{R_1, \frac{\widehat{B}_n R_1}{\sum_{i=1}^n r_i d_i}, \min\{R \mid X_F(R) \le Y_F(R)\}\right\}.$$

As $\max\left\{R_1, \frac{\widehat{B}_n R_1}{\sum_{i=1}^n r_i d_i}\right\} \le \widehat{R}_R^* \le \widehat{R}^* = \max\left\{R_1, \frac{\widehat{B}_n}{d_n}\right\}$, where $\widehat{R}^*$ is the minimum required bandwidth achieved by a base (no reprofiling) fifo system, we can use Proposition 10 and a binary search to compute $\widehat{R}_R^*$. Once $\widehat{R}_R^*$ is known, Proposition 11 gives the optimal reprofiling solution.

Proposition 11. *For $1 \le i \le n$, define $T_i(\widehat{B}'_n, R) = \max\left\{0, \frac{R}{R+r_i}\left(H_i + \frac{r_i}{R}\widehat{B}'_n\right), b_i + \frac{r_i(\widehat{B}'_n - Rd_i)}{R_1}\right\}$. The optimal reprofiling solution $\widehat{\boldsymbol{b}}'^*$ of **OPT_RF**'s is given by $\widehat{b}'^*_1 = \widehat{B}'^*_1$, and $\widehat{b}'^*_i = \widehat{B}'^*_i - \widehat{B}'^*_{i-1}$, $2 \le i \le n$, where $\widehat{\boldsymbol{B}}'^*_i$ satisfy*

$$\begin{cases} \widehat{B}'^*_n = X_F(\widehat{R}_R^*), \\ \widehat{B}'^*_i = \max\left\{\sum_{j=1}^i T_j(\widehat{B}'^*_n, \widehat{R}_R^*), \widehat{B}'^*_{i+1} - b_{i+1}\right\}, \\ \textit{when } 1 \le i \le n-1. \end{cases} \tag{10}$$

Note that $\widehat{B}'^*_n$ relies only on $\widehat{R}_R^*$ and flows' profiles. Whereas when $1 \le i \le n-1$, $\widehat{B}'^*_i$ relies only on $\widehat{R}_R^*$, $\widehat{B}'^*_n$, $\widehat{B}'^*_{i+1}$ and flows' profiles. Hence, we can recursively characterize $\widehat{B}'^*_i$ from $\widehat{B}'^*_n$ given $\widehat{R}_R^*$.

VI. EVALUATION

In this section, we explore the relative benefits of the solutions developed in the previous three sections. Of interest is assessing the "cost of simplicity," namely, the amount of additional bandwidth required by simpler schedulers such as static priority or fifo compared to an edf scheduler. Also of interest is the magnitude of the improvements that reprofiling affords with static priority and fifo schedulers. To that end, the evaluation proceeds with a number of *pairwise comparisons* to quantify the relative (bandwidth) cost of each alternative.

The evaluation first focuses (Section VI-A) on scenarios with just two flows. Closed-form expressions are then available for the minimum bandwidth of each configuration, which make formal comparisons possible. Section VI-B extends this to more "general" scenarios involving multiple flows with different combinations of deadlines and traffic profiles.

In the initial two-flow comparisons of Section VI-A, we first select a pair of representative traffic profiles (token buckets), and then vary the flows' respective deadlines over a wide range of values. For each such combination, we explicitly compute the relative differences in bandwidth required by the different schedulers (with and without reprofiling, as applicable) using expressions derived from the propositions obtained in the previous sections. The results are presented in the form of "heat-maps" across the range of deadline combinations.

For the more general scenarios involving multiple flows (Section VI-B), we first generate a set of flow profiles, *i.e.,* token buckets and deadlines, by randomly selecting them from within specified ranges. For each such combination, the amount of bandwidth required to meet the flows' deadlines are then computed using again results from the propositions derived in the previous sections. Finally, for each pair of schedulers, we report statistics (means, standard deviations and the 95% confidence intervals of the means) of the relative bandwidth differences across those random selections.

A. Basic Two-Flow Configurations

Recalling our earlier notation for the minimum bandwidth in each configuration, *i.e.,* R^* (edf); $\widetilde{R}^*$ (static priority); $\widetilde{R}^*_R$ (static priority w/ reprofiling); $\widehat{R}^*$ (fifo); and $\widehat{R}^*_R$ (fifo w/ reprofiling), and specializing Eq. (2) to a configuration with two flows, (r_1, b_1) and (r_2, b_2), the absolute minimum bandwidth to meet the flows' deadlines d_1 and d_2 is given by

$$R^* = \max\left\{r_1 + r_2,\ \frac{b_2}{d_2},\ \frac{b_1 + b_2 - r_2 d_2}{d_1} + r_2\right\}, \tag{11}$$

which is then also the bandwidth required by the edf scheduler.

Similarly, if we consider a static priority scheduler, from Eq. (3), its bandwidth requirement $\widetilde{R}^*$ (in the absence of any reprofiling) for the same two-flow configuration is of the form

$$\widetilde{R}^* = \max\left\{r_1 + r_2,\ \frac{b_2}{d_2},\ \frac{b_1 + b_2}{d_1} + r_2\right\}; \tag{12}$$

If (optimal) reprofiling is introduced, specializing Proposition 7 to two flows, the minimum bandwidth $\widetilde{R}^*_R$ reduces to

$$\begin{cases} \max\left\{r_1 + r_2, \dfrac{b_2}{d_2}, \dfrac{b_1 + b_2 - r_2 d_2}{d_1} + r_2\right\}, \text{ when } \dfrac{b_2}{r_2} \geq \dfrac{b_1}{r_1} \\ \max\left\{r_1 + r_2, \dfrac{b_2}{d_2}, \dfrac{b_1 + \max\{b_2 - r_2 d_2, 0\}}{d_1} + r_2\right\}, \\ \text{otherwise;} \end{cases} \tag{13}$$

Finally, specializing the results of Propositions 10 and 11 to two flows, we find that the minimum required bandwidth $\widehat{R}^*$ under fifo without reprofiling is

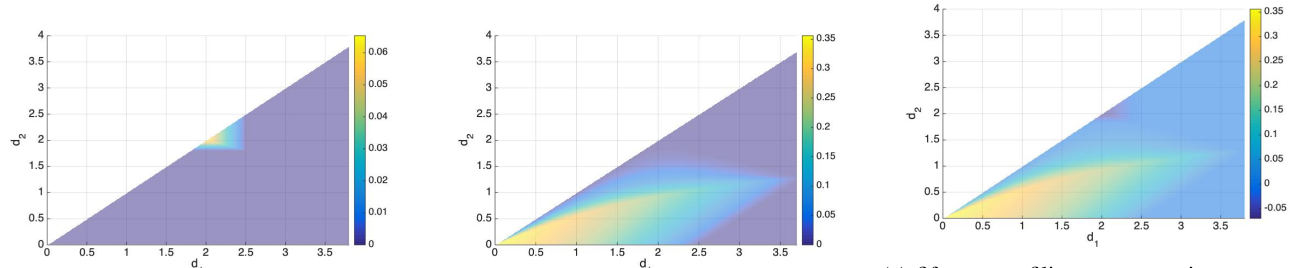
$$\widehat{R}^* = \max\left\{r_1 + r_2,\ \frac{b_1 + b_2}{d_2}\right\}; \tag{14}$$

and that when (optimal) reprofiling is used, $\widehat{R}^*_R$ is given by Eq. (15). With these expressions in hand, we can now assess the

$$\widehat{R}^*_R = \max\left\{r_1 + r_2, \frac{b_2}{d_2}, \frac{(b_1 + b_2)(r_1 + r_2)}{d_1 r_1 + d_2 r_2}, \frac{b_1 + b_2 - d_1 r_1 + \sqrt{(b_1 + b_2 - d_1 r_1)^2 + 4 r_1 d_2 b_2}}{2 d_2}\right\}. \tag{15}$$

relative benefits of each option in this two-flow scenario.

Specifically, we consider next combinations consisting of two flows with representative token bucket parameters $(r_1, b_1) = (4, 10)$ and $(r_2, b_2) = (10, 18)$, and systematically vary their respective deadlines $d_1 \geq d_2$ over a range of values. The bandwidth required to meet the deadlines is then compared for different pairs of schedulers using the expressions reported in Eq. (11), Eq. (13), and Eq. (15).



(a) Dyn. prio. vs. stat. prio. + reprofiling
(b) Dyn. prio. vs. fifo + reprofiling
(c) fifo + reprofiling vs. stat. prio. + reprofiling

Fig. 2: Relative bandwidth increases for $(r_1, b_1) = (4, 10)$ and $(r_2, b_2) = (10, 18)$, as a function of d_1 and $d_2 < d_1$. The figure is in the form of a heat-map. Darker colors (purple) correspond to smaller increases than lighter ones (yellow).

1) The Impact of Scheduler Complexity: We first evaluate the impact of relying on schedulers of decreasing complexity, when those schedulers are coupled with an optimal reprofiling solution. In other words, we compare the bandwidth requirements of an edf scheduler to those of static priority and fifo schedulers combined with an optimal reprofiler. The comparison is in the form of relative differences (improvements realizable from more complex schedulers), *i.e.,* $\frac{\widetilde{R}_R^* - R^*}{\widetilde{R}_R^*}$, $\frac{\widehat{R}_R^* - R^*}{\widehat{R}_R^*}$, and $\frac{\widehat{R}_R^* - \widetilde{R}_R^*}{\widehat{R}^*}$.

edf vs. static priority w/ optimal reprofiling.

We start with comparing an edf scheduler with a static priority scheduler plus optimal reprofiling. Eqs. (11) and (13) then state that $R^* < \widetilde{R}_R^*$ iff $\frac{b_2}{r_2} < d_2 \leq d_1 < \frac{b_1}{r_1}$.

The results are reported in Fig. 2a, and, as mentioned, are in the form of a heat-map of the relative bandwidth differences as the flows' respective deadlines vary. As shown in the figure, a static priority scheduler, when combined with reprofiling, performs as well as an edf scheduler, except for a relatively small (triangular) region where d_1 and d_2 are close to each other and both of intermediate values[5]. Towards better characterizing this range, *i.e.,* $d_2 > \frac{b_2}{r_2}$ and $d_1 < \frac{b_1}{r_1}$, we see that the supremum of $\frac{\widetilde{R}_R^* - R^*}{\widetilde{R}_R^*}$ is achieved at $d_1 = d_2 = \frac{b_1+b_2}{r_1+r_2}$, with $\widetilde{R}_R^* = \frac{b_1}{d_1} + r_2$, and $R^* = r_1 + r_2$. The relative difference in bandwidth between the two schemes is then of the form

$$\frac{\widetilde{R}_R^* - R^*}{\widetilde{R}_R^*} = 1 - \frac{1}{\frac{b_1}{b_1+b_2} + \frac{r_2}{r_1+r_2}},$$

which can be shown to be upper-bounded by 0.5. In other words, in the two-flow case, the (optimal) edf scheduler can result in a bandwidth saving of at most 50% when compared to a static priority scheduler with (optimal) reprofiling. This happens when the deadlines of the two flows are very close to each other, a scenario unlikely in practice.

edf vs. fifo w/ optimal reprofiling

Next, we compare an edf scheduler and a fifo scheduler plus optimal reprofiling. Eqs. (11) and (15) state that $\widehat{R}_R^* > R^*$ iff $d_1 - \frac{b_1}{r_1} < d_2 < \frac{b_1+b_2-d_1 r_1}{r_2}$. We illustrate the corresponding relative differences in Fig. 2b using the same two-flow combination as before. From the figure, we see that fifo + reprofiling performs poorly relative to an edf scheduler when neither d_1 nor d_2 are large. As with static priorities, such configurations may not be common in practice.

We note that the supremum of $\frac{\widehat{R}_R^* - R^*}{\widehat{R}_R^*}$ is achieved when $0 < d_2 < \frac{b_1+b_2+r_2 d_1 - \sqrt{(b_1+b_2+r_2 d_1)^2 - 4 r_2 b_2 d_1}}{2 r_2}$, with Eq. (11) defaulting to $R^* = \frac{b_2}{d_2}$ and Eq. (15) to $\widehat{R}_R^* = \frac{b_1+b_2-d_1 r_1 + \sqrt{(b_1+b_2-d_1 r_1)^2 + 4 r_1 d_2 b_2}}{2 d_2}$. Hence, the relative difference becomes

$$\frac{\widehat{R}_R^* - R^*}{\widehat{R}_R^*} = 1 - \frac{2 b_2}{b_1 + b_2 - d_1 r_1 + \sqrt{(b_1 + b_2 - d_1 r_1)^2 + 4 r_1 d_2 b_2}},$$

which increases with d_2. Thus, its supremum is achieved as $d_2 \to d_1$. Similarly, one easily shows that $1 - \frac{2b_2}{b_1+b_2-d_1 r_1 + \sqrt{(b_1+b_2-d_1 r_1)^2 + 4 r_1 d_2 b_2}}$ decreases with d_1. Hence, the supremum of the relative difference is achieved as $d_1 \to 0$, and is of the form $\frac{b_1}{b_1+b_2}$, which goes

[5](i) When d_2 and d_1 are close and small, the bandwidth required to meet the deadlines is very large under either edf or static priority schedulers. This ensures that both produce similar transmissions' orders. Consider, for example, a low-priority (larger deadline) burst that arrives $(d_1 - d_2)$ before a high-priority (smaller deadline) one. It has higher priority under edf, and the speed of the link ensures it is transmitted before the arrival of the high-priority burst, which ensures no difference between edf and a static priority scheduler.
(ii) When d_2 and d_1 are close but large, both schedulers meet their deadlines with the same bandwidth, *i.e.,* the sum of the flows' average rates.

to 1 as $\frac{b_1}{b_2} \to \infty$. In other words, an edf scheduler can yield a 100% improvement over a fifo scheduler with optimal reprofiling.

Fifo vs. static priority both w/ optimal reprofiling

Finally, we compare fifo and static priority schedulers when both rely on optimal reprofiling. Eqs. (13) and (15) give that $\widehat{R}_R^* > \widetilde{R}_R^*$ iff $\max\left\{\frac{b_2}{r_2}, \frac{(b_1+b_2)(r_1+r_2)}{r_2(b_1/d_1+r_2)}\right\} < d_1 < \frac{b_1}{r_1}$. Fig. 2c illustrates the difference, again relying on a heat-map for the same two-flow combination as the two previous scenarios.

The figure shows that the benefits of priority are maximum when d_2 is small and d_1 is not too large. This is intuitive in that a small d_2 calls for affording maximum protection to flow 2, which a priority structure offers more readily than a fifo. Conversely, when d_1 is large, flow 1 can be reprofiled to eliminate all burstiness, which limits its impact on flow 2 even when both flows compete in a fifo scheduler.

The figure also reveals that a small region exists (when d_1 and d_2 are close to each other and both are of intermediate value) where fifo outperforms static priority. As alluded to in the discussion following Proposition 4 and as expanded in Appendix E, this is because a *strict* priority ordering of flows as a function of their deadlines needs not always be optimal. For instance, it is easy to see that two otherwise identical flows that only differ infinitesimally in their deadlines should be treated "identically." This is more readily accomplished by having them share a common fifo queue than assigned to two distinct priorities.

To better understand differences in performance between the two schemes, we characterize the supremum and the infimum of $\frac{\widehat{R}_R^*-\widetilde{R}_R^*}{\widehat{R}_R^*}$. Basic algebraic manipulations show that the supremum is achieved as $d_1 = d_2 \to 0$, where Eq. (15) defaults to $\widehat{R}_R^* = \frac{b_1+b_2-d_1r_1+\sqrt{(b_1+b_2-d_1r_1)^2+4r_1d_2b_2}}{2d_2}$ and Eq. (13) to $\widetilde{R}_R^* = \frac{b_2}{d_2}$, so that their relative difference is ultimately of the form

$$\frac{\widehat{R}_R^* - \widetilde{R}_R^*}{\widehat{R}_R^*} = \frac{b_1}{b_1+b_2},$$

which goes to 1 as $\frac{b_1}{b_2} \to \infty$, *i.e.,* a maximum penalty of 100% for fifo with reprofiling over static priorities with reprofiling.

Conversely, the infimum is achieved at $d_1 = d_2 = \frac{b_1+b_2}{r_1+r_2}$, with Eqs. (13) and (15) defaulting to $\widetilde{R}_R^* = \frac{b_1}{d_1} + r_2$ and $\widehat{R}_R^* = r_1 + r_2$, and a relative difference of the form

$$\frac{\widehat{R}_R^* - \widetilde{R}_R^*}{\widehat{R}_R^*} = \frac{r_1}{r_1+r_2} - \frac{b_1}{b_1+b_2},$$

which increases with $\frac{r_1}{r_2}$ and decreases with $\frac{b_1}{b_2}$. When $\frac{r_1}{r_2} \to 0$ and $\frac{b_1}{b_2} \to \infty$, it achieves an infimum of -1, *i.e.,* a maximum penalty of 100% but now for static priorities with reprofiling over fifo with reprofiling. In other words, when used with reprofiling, both fifo and static priority can require twice as much bandwidth as the other.

Ensuring that static priority always outperforms fifo calls for determining when flows should be grouped in the same priority class rather than assigned to separate classes. Such grouping can be identified in simple scenarios with two or three flows, *e.g.,* see Appendix E, but a general solution appears challenging. However, as we shall see in Section VI-B, the simple strict priority assignment on which we rely performs well in practice across a broad range of flow configurations.

2) The Benefits of Reprofiling: In this section, we evaluate the benefits afforded by (optimally) reprofiling flows with static priority and fifo schedulers. This is done by computing for both schedulers the minimum bandwidth required to meet flows' deadlines without and with reprofiling, and evaluating the resulting relative differences, *i.e.,* $\frac{\widetilde{R}^*-\widetilde{R}_R^*}{\widetilde{R}^*}$ and $\frac{\widehat{R}^*-\widehat{R}_R^*}{\widehat{R}^*}$.

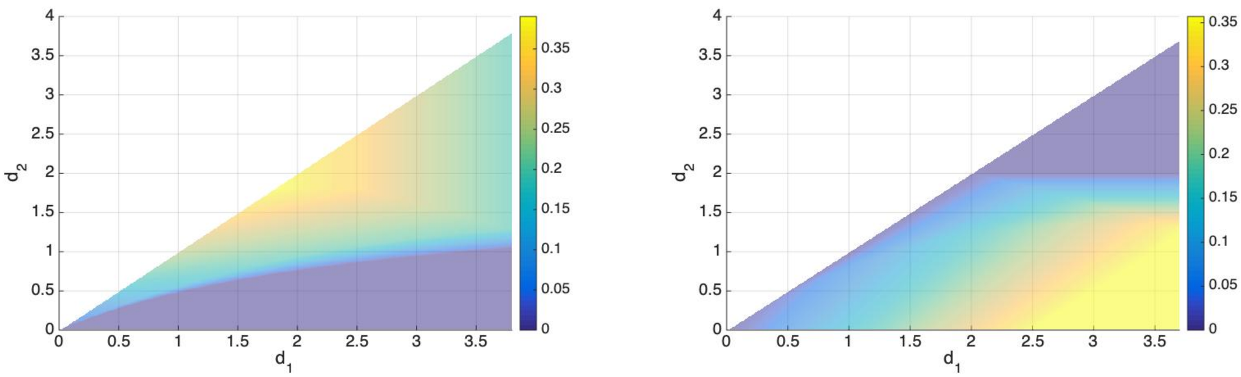
For a static priority scheduler, Eqs. (12) and (13) indicate that $\widetilde{R}_R^* < \widetilde{R}^*$ iff $\widetilde{R}^* = \frac{b_1+b_2}{d_1} + r_2 > \max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$, *i.e.,* for a static priority scheduler, reprofiling[6] decreases the required bandwidth only when d_1, the larger deadline, is not too large and d_2, the smaller deadline, is not too small. This is intuitive. When d_1 is large, the low-priority flow 1 can meet its deadline even without any mitigation of the impact of flow 2. Conversely, a small d_2 offers little to no opportunity for reprofiling flow 2 as the added delay it introduces would need to be compensated by an even higher link bandwidth. This is illustrated in Fig. 3a for the same two-flow combination as in Fig. 2, *i.e.,* $(r_1, b_1) = (4, 10)$ and $(r_2, b_2) = (10, 18)$. The intermediate region where "d_1 is not too large and d_2 is not too small" corresponds to the yellow triangular region where the benefits of reprofiling can reach 40%.

Similarly, Eqs. (14) and (15) indicate that $\widehat{R}_R^* < \widehat{R}^*$ iff $d_2 < \frac{b_1+b_2}{r_1+r_2}$, *i.e.,* for a fifo scheduler, reprofiling decreases the required bandwidth only when d_2, the smaller deadline, is small. This is again intuitive as a large d_2 means that the deadline can be met even without reprofiling flow 1[7]. Fig. 3b presents the relative gain in bandwidth for again the same 2-flow combination. As in the static priority case, the figure shows that for a fifo scheduler the benefits of reprofiling can reach about 40% in the example under consideration.

The next section explores scenarios involving combinations of multiple flow profiles. Based on those results it appears that, unsurprisingly, a fifo scheduler stands to generally benefit more from reprofiling than a static priority one.

[6]Recall from Corollary 6 that the flow with the largest deadline, flow 1 in the two-flow case, is never reprofiled.

[7]Note that under static priority the smaller deadline flow is reprofiled, while in the fifo case it is the larger deadline flow that is reprofiled to minimize its impact on the one with the tighter deadline.



(a) Stat. prio. without vs. with reprofiling

(b) fifo without vs. with reprofiling

Fig. 3: Relative bandwidth increases for $(r_1, b_1) = (4, 10)$ and $(r_2, b_2) = (10, 18)$ as a function of d_1 and $d_2 < d_1$. The figure is in the form of a heat-map. Darker colors (purple) correspond to smaller increases than lighter ones (yellow).

B. Relative Performance – Multiple Flows

In this section, we extend the investigation to configurations with more than two flows, using both synthetic flow profiles and profiles derived from datacenter traffic traces. The evaluation relies on generating a set of flow profiles, *i.e.,* token buckets plus deadlines, and for each combination compute the bandwidth required to meet the deadlines using the results derived in the paper. The main difference with the 2-flow configurations of Section VI-A is that, as described next, we now consider a wider range of token bucket parameters with different possible combinations of deadlines. Additionally, unlike the 2-flow configurations for which the amount of bandwidth required could be obtained from explicit expressions, *i.e.,* Eq. (11), Eq. (13), and Eq. (15), computing the required bandwidth now typically involves numerical procedures, as documented in the propositions derived in the paper.

1) Synthetic Flow Profiles: We assign flows to *ten* different deadline classes with a dynamic range of 10, *i.e.,* with minimum and maximum deadlines of 0.1 and 1, respectively, and consider different spreads in that range for the 10 deadlines classes. Specifically, we select three different possible types of spreads for deadline classes, namely,

Even deadline spread:

1) $\boldsymbol{d}_{11} = (1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1)$;

Bi-modal deadline spread:

2) $\boldsymbol{d}_{21} = (1, 0.95, 0.9, 0.85, 0.8 \| 0.3, 0.25, 0.2, 0.15, 0.1)$,
3) $\boldsymbol{d}_{22} = (1, 0.96, 0.93, 0.9, 0.86, 0.83, 0.8 \| 0.2, 0.15, 0.1)$,
4) $\boldsymbol{d}_{23} = (1, 0.95, 0.9 \| 0.3, 0.26, 0.23, 0.2, 0.16, 0.13, 0.1)$;

Tri-modal deadline spread:

5) $\boldsymbol{d}_{31} = (1, 0.95, 0.9 \| 0.6, 0.55, 0.5, 0.45 \| 0.2, 0.15, 0.1)$,
6) $\boldsymbol{d}_{32} = (1 \| 0.68, 0.65, 0.62, 0.6, 0.57, 0.55, 0.53, 0.5 \| 0.1)$,
7) $\boldsymbol{d}_{33} = (1 \| 0.6 \| 0.28, 0.25, 0.23, 0.2, 0.17, 0.15, 0.12, 0.1)$,
8) $\boldsymbol{d}_{34} = (1, 0.97, 0.95, 0.93, 0.9, 0.88, 0.85, 0.82 \| 0.6 \| 0.1)$.

Those three types of spreads translate into different groupings of deadlines, which affect the relative numbers of deadlines in close proximity to each others.

Each of the above eight groupings is used across $1,000$ experiments, where an experiment consists of randomly selecting a "flow's" traffic profile for each of the ten deadline classes. Note that what we denote by a flow, in practice maps to the aggregate of all individual flows assigned to the corresponding deadline class (individual flow profiles add up). Flow profiles are generated by independently drawing ten (aggregate) flow burst sizes b_1 to b_{10} from $U(1, 10)$, and ten (aggregate) rates r_1 to r_{10} from $U(0, r_{\max})$. The upper bound $r_{\max}$ corresponds to a rate value beyond which a fifo scheduler always performs as well as the optimal solution even without reprofiling[8].

The primary purpose of those synthetic experiments is to allow a systematic exploration of the performance of the different schemes across a broad range of configurations. The results can then be used to assess the expected performance of each scheme for individual configurations of practical interest.

The results of the experiments are summarized in Table I, which gives the mean, standard deviation, and the mean's 95% confidence interval for the *relative savings* in link bandwidth, first for edf over static priority with reprofiling, followed by edf over fifo with reprofiling, and then static priority over fifo both with reprofiling. As mentioned, bandwidth values are computed numerically for each configuration using results from the previously derived propositions.

[8]This happens when the sum of the rates is large enough to alone clear the aggregate burst before the smallest deadline, *i.e.,* $10r_{\max} = \frac{\sum_{i=1}^{10} b_i}{0.1}$.

TABLE I: Bandwidth savings from scheduler choice.
Synthetic flow profiles.

Comparisons	**Scenario**	**Mean**	**Std. Dev.**	95% **Conf.**
edf vs. static+reprofiling R^* vs. $\widetilde{R}^*_R$ $\left(\frac{\widetilde{R}^*_R - R^*}{\widetilde{R}^*_R}\right)$	$\boldsymbol{d}_{11}$	1.2%	2.3%	$[1.02\%, 1.31\%]$
	$\boldsymbol{d}_{21}$	1.5%	2.7%	$[1.35\%, 1.69\%]$
	$\boldsymbol{d}_{22}$	1.1%	2.7%	$[1.01\%, 1.28\%]$
	$\boldsymbol{d}_{23}$	2.9%	4.2%	$[2.59\%, 3.12\%]$
	$\boldsymbol{d}_{31}$	1.4%	2.5%	$[1.2\%, 1.51\%]$
	$\boldsymbol{d}_{32}$	1.0%	2.1%	$[0.84\%, 1.1\%]$
	$\boldsymbol{d}_{33}$	6.2%	6.5%	$[5.76\%, 6.58\%]$
	$\boldsymbol{d}_{34}$	0.7%	1.7%	$[0.6\%, 0.81\%]$
edf vs. fifo+reprofiling R^* vs. $\widehat{R}^*_R$ $\left(\frac{\widehat{R}^*_R - R^*}{\widehat{R}^*_R}\right)$	$\boldsymbol{d}_{11}$	1.7%	6.5%	$[1.13\%, 2.11\%]$
	$\boldsymbol{d}_{21}$	3.2%	8.7%	$[2.68\%, 3.76\%]$
	$\boldsymbol{d}_{22}$	1.7%	6.2%	$[1.26\%, 2.03\%]$
	$\boldsymbol{d}_{23}$	8.0%	12.8%	$[7.24\%, 8.82\%]$
	$\boldsymbol{d}_{31}$	2.5%	7.8%	$[2.06\%, 3.03\%]$
	$\boldsymbol{d}_{32}$	0.8%	4.6%	$[0.54\%, 1.11\%]]$
	$\boldsymbol{d}_{33}$	12.0%	14.1%	$[11.15\%, 12.9\%]$
	$\boldsymbol{d}_{34}$	0.4%	3.2%	$[0.2\%, 0.6\%]$
static vs. fifo both w/ reprofiling $\widetilde{R}^*_R$ vs. $\widehat{R}^*_R$ $\left(\frac{\widehat{R}^*_R - \widetilde{R}^*_R}{\widehat{R}^*_R}\right)$	$\boldsymbol{d}_{11}$	0.6%	6.5%	$[0.16\%, 0.95\%]$
	$\boldsymbol{d}_{21}$	1.8%	8.3%	$[1.26\%, 2.28\%]$
	$\boldsymbol{d}_{22}$	0.5%	6.1%	$[0.12\%, 0.88\%]$
	$\boldsymbol{d}_{23}$	5.5%	11.3%	$[4.84\%, 6.24\%]$
	$\boldsymbol{d}_{31}$	1.2%	7.5%	$[0.76\%, 1.69\%]$
	$\boldsymbol{d}_{32}$	-0.2%	4.5%	[-0.43%, 0.13%]
	$\boldsymbol{d}_{33}$	6.6%	11.2%	$[5.92\%, 7.3\%]$
	$\boldsymbol{d}_{34}$	-0.3%	3.3%	[-0.53%, -0.12%]

The first conclusion one can draw from Table I is that while an edf scheduler affords some benefits, they are on average smaller than the maximum values of Section VI-A. Average improvements over static priority with reprofiling hover around 1% and did not exceed about 6%. Improvements are a little higher when considering fifo with reprofiling, where they reach 12%, but those values are still significantly less than the worst case scenarios of Section VI-A.

Table I also reveals that, somewhat surprisingly, static priority and fifo perform similarly when both are afforded the benefit of reprofiling (the largest difference observed in the experiments is 5.5%). Static priority has an edge on average even if, as discussed in Section VI-A, a few scenarios exist where a fifo scheduler outperforms static priority when both are combined with reprofiling, *e.g.,* d_{32} and d_{34}. Recall that this is because we strictly map smaller deadlines to higher priority. The differences are, however, small, *i.e.,* 0.2% and 0.3%, respectively, for the two scenarios where fifo outperforms static priority on average.

TABLE II: Benefits of reprofiling for static priority & fifo.
Synthetic flow profiles.

Comparisons	**Scenario**	**Mean**	**Std. Dev.**	95% **Conf.**
static w/ & w/o reprofiling $\widetilde{R}^*_R$ vs. $\widetilde{R}^*$ $\left(\frac{\widetilde{R}^* - \widetilde{R}^*_R}{\widetilde{R}^*}\right)$	$\boldsymbol{d}_{1}$	8.43%	4.50%	$[8.15\%, 8.71\%]$
	$\boldsymbol{d}_{21}$	8.11%	4.19%	$[7.85\%, 8.37\%]$
	$\boldsymbol{d}_{22}$	8.42%	4.52%	$[8.14\%, 8.71\%]$
	$\boldsymbol{d}_{23}$	9.38%	4.80%	$[9.08\%, 9.67\%]$
	$\boldsymbol{d}_{31}$	8.24%	4.33%	$[7.97\%, 8.51\%]$
	$\boldsymbol{d}_{32}$	9.49%	5.07%	$[9.18\%, 9.81\%]$
	$\boldsymbol{d}_{33}$	15.97%	4.78%	$[15.67\%, 16.27\%]$
	$\boldsymbol{d}_{34}$	8.83%	4.94%	$[8.53\%, 9.14\%]$
fifo w/ & w/o reprofiling $\widehat{R}^*_R$ vs. $\widehat{R}^*$ $\left(\frac{\widehat{R}^* - \widehat{R}^*_R}{\widehat{R}^*}\right)$	$\boldsymbol{d}_{1}$	49.52%	8.17%	$[49.01\%, 50.03\%]$
	$\boldsymbol{d}_{21}$	48.71%	7.62%	$[48.24\%, 49.18\%]$
	$\boldsymbol{d}_{22}$	49.53%	8.27%	$[49.02\%, 50.05\%]$
	$\boldsymbol{d}_{23}$	45.78%	6.52%	$[45.37\%, 46.18\%]$
	$\boldsymbol{d}_{31}$	49.08%	7.88%	$[48.59\%, 49.57\%]$
	$\boldsymbol{d}_{32}$	49.95%	8.59%	$[49.42\%, 50.49\%]$
	$\boldsymbol{d}_{33}$	42.47%	6.19%	$[42.08\%, 42.85\%]$
	$\boldsymbol{d}_{34}$	50.13%	8.84%	$[49.59\%, 50.68\%]$

Towards gaining a better understanding of reprofiling and the extent to which it is behind the somewhat unexpected good performance of fifo, Table II reports its impact for both static priority and fifo. As Table I, it gives the mean, standard deviation, and the mean's 95% confidence interval, but now of the relative gains in bandwidth that reprofiling affords over no reprofiling for both fifo and static priority schedulers.

The data from Table II highlights that while both static priority and fifo benefit from reprofiling, the magnitude of the improvements is significantly higher for fifo. Specifically, improvements from reprofiling are systematically above 40% and often close to 50% for fifo, while they exceed 10% only once for static priority (at 15% for scenario d_{33}) and are typically

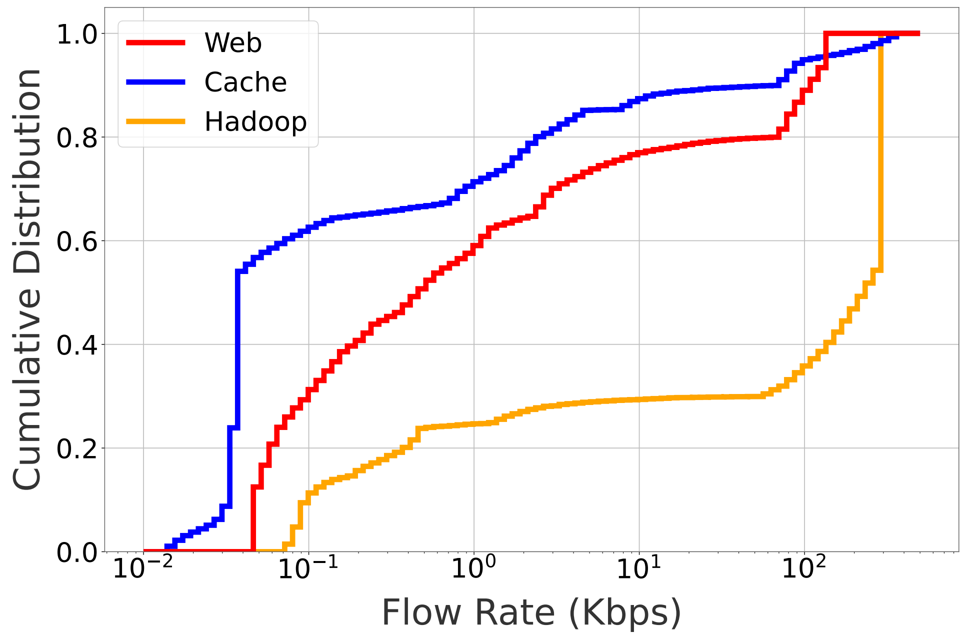


Fig. 4: CDF of flow rates for Web, Cache, & Hadoop applications from [23] assuming correlated flow durations and sizes.

around 8%. As alluded to earlier, this is not surprising given that static priority offers some ability to discriminate flows based on their deadlines, while fifo does not.

2) Application Derived Flow Profiles: The benefits of synthetic flow profiles in allowing a systematic investigation notwithstanding, it is also of interest to target configurations more directly representative of traffic mixes as they arise in practice. To that end, we rely on a methodology similar to that used in [16, Section VIII-B2], and construct a set of flow profiles derived from traffic data reported in [23].

Specifically, [23] investigates the traffic flowing through the network of one of Facebook's large datacenter, and reports, among other things, the distribution of flow sizes and durations (Figs. 6 & 7 of [23]) for three representative applications: Web (W), Cache read and replacement (C), and Hadoop (H). We rely on these data to generate sample traffic profiles (r, b) for flows from those three applications as follows:

1) For a given application, we generate flow size+duration tuples by sampling the corresponding distributions assuming they are perfectly positively correlated. In other words, we assume that larger flows last longer.
2) A flow's token rate r is then obtained by dividing the flow size by its duration. Fig. 4 shows the resulting cumulative distributions of flow rates for all three applications.
3) Generating token bucket sizes b involves an additional step and associated assumption:
   a) The smallest flow sizes from Fig. 6 of [23] are assumed representative of a single transmission burst. This yields burst sizes $S_W = 0.15$Kbytes, $S_C = 0.4$Kbytes, and $S_H = 0.3$Kbytes for our three sample applications.
   b) As bucket sizes are typically chosen to accommodate consecutive bursts, we leverage the claim in [23] that all three applications are "internally bursty" with Cache significantly burstier than Hadoop, and Web in between, to randomly select bucket sizes in $[0, 20S_C]$, $[0, 10S_W]$ and $[0, 2S_H]$, respectively. We note that these values yield relatively small buckets, and, therefore, maximum burst sizes.

The resulting profiles have relatively low rates and burstiness, at least when it comes to individual flows, with Hadoop's profile typical of bandwidth hungry applications, and Web and Cache representative of more interactive applications. This maps to the types of services that [23] mentions as relying on those three applications, *i.e.,* Web search, user data query, and offline analysis (*e.g.,* data mining). As a result, we assign deadlines to each application that broadly reflect those services, with three deadline classes set to 10ms, 50ms, and 200ms for Web, Cache, and Hadoop respectively.

In evaluating performance, we consider four "traffic mixes" that differ in their relative proportion of flows from each application. Specifically, the corresponding four scenarios sample our W, C, H applications in the proportions: 1:1:1, 3:9:1, 9:3:1, and 9:9:1, respectively. For each scenario, we randomly sample 100 flow profiles in those proportions. The 100 flows are then grouped according to their deadline class, which yields a set of three aggregates to schedule on the shared link according to their deadlines. This procedure is repeated 1000 times, and the relative link bandwidth requirements across schedulers and reprofiling options are given in Table III.

The first conclusion from Table III is that static priority with reprofiling performs just as well as edf for all four scenarios (top four rows). This is not surprising. The large gaps between the deadlines of the three classes of applications ensure that once

TABLE III: Bandwidth savings & benefits of reprofiling. Application-derived flow profiles.

Comparisons	Scenario	Mean	Std. Dev.	95% Conf.
R^* vs. $\widetilde{R}^*_R$ $\left(\frac{\widetilde{R}^*_R - R^*}{\widetilde{R}^*_R}\right)$	1:1:1	0.0%	0.0%	[0%, 0%]
	3:9:1	0.0%	0.0%	[0%, 0%]
	9:3:1	0.0%	0.0%	[0%, 0%]
	9:9:1	0.0%	0.0%	[0%, 0%]
R^* vs. $\widehat{R}^*_R$ $\left(\frac{\widehat{R}^*_R - R^*}{\widehat{R}^*_R}\right)$	1:1:1	41.82%	17.04%	[40.76%, 42.88%]
	3:9:1	72.36%	3.43%	[72.15%, 72.58%]
	9:3:1	56.84%	8.43%	[56.32%, 57.37%]
	9:9:1	69.70%	4.10%	[69.45%, 69.96%]
$\widetilde{R}^*_R$ vs. $\widehat{R}^*_R$	See above			
$\widetilde{R}^*_R$ vs. $\widetilde{R}^*$ $\left(\frac{\widetilde{R}^* - \widetilde{R}^*_R}{\widetilde{R}^*}\right)$	1:1:1	0.01%	0.22%	[0.00%, 0.03%]
	3:9:1	1.74%	0.73%	[1.70%, 1.79%]
	9:3:1	1.02%	2.34%	[0.87%, 1.16%]
	9:9:1	4.06%	1.19%	[3.98%, 4.13%]
$\widehat{R}^*_R$ vs. $\widehat{R}^*$ $\left(\frac{\widehat{R}^* - \widehat{R}^*_R}{\widehat{R}^*}\right)$	1:1:1	22.74%	9.17%	[22.17%, 23.31%]
	3:9:1	21.61%	8.48%	[21.09%, 22.14%]
	9:3:1	14.93%	8.93%	[14.37%, 15.48%]
	9:9:1	19.55%	8.62%	[19.02%, 20.08%]

high-priority bursts are cleared, the residual bandwidth is sufficient to transmit lower priority bursts before their deadline. As with synthetic profiles, reprofiling is instrumental in realizing this outcome, even if the wide gaps between deadlines together with the limited burstiness of the applications produce a smaller gain ($\widetilde{R}^*_R$ vs. $\widetilde{R}^*$).

The benefits of reprofiling are again more apparent with fifo, even if it significantly under-performs both edf and static priority. The lack of discrimination across flows that fifo suffers from is exacerbated by the large gaps between deadlines, and compensating for it calls for an average of about 50% more bandwidth across all four scenarios. However, without reprofiling this bandwidth increase ($\widehat{R}^*_R$ vs. $\widehat{R}^*$) is around 20% larger. This again demonstrates the extent to which reprofiling can help simpler schedulers.

C. Summary Discussion

Several common themes emerge between the evaluations of Sections VI-B1 and VI-B2. The first is that reprofiling can help a static priority scheduler perform nearly as well as an edf scheduler. Second, while its inability to discriminate between flows puts fifo at a clear disadvantage, reprofiling is again capable of partially mitigating its handicap. Finally, while with static priority the benefits of reprofiling are realized by reprofiling high-priority flows to limit their impact on low-priority ones, the opposite holds for fifo.

The differences between the results of Sections VI-B1 and VI-B2 also revealed a number of intuitive findings brought about by the differences in deadline spreads in the two scenarios. In particular, the large gaps between deadlines present in Section VI-B2 make it easier for a static priority scheduler to perform nearly as well as an edf scheduler. This holds with and without reprofiling, even if reprofiling remains useful. Conversely, more closely packed deadlines offer additional opportunities for reprofiling to be useful, as closer deadlines amplify the need for fine tuning of a flow's profile relative to its deadline and impact on other flows.

VII. Related Works

The question of meeting deadlines for a set of rate-limited flows is one that has received much attention in the scheduling literature. It is not our intent to provide an exhaustive review of those works. Instead, we limit ourselves to highlighting works whose results are closest to ours or that offered early insight into the problem, including the benefits of adjusting flows' profiles (reprofiling) that is one of the foci of this paper.

a) Packet-level shaping and scheduling: Scheduling flows with deterministic traffic profiles was investigated in [18] that considered both buffer and delay requirements. In particular, the paper established[9] the optimality of the edf policy in terms of maximizing the schedulable region. This is the "dual" of the bandwidth minimization problem investigated in this paper, and the result parallels that of Proposition 2. Static priority and fifo schedulers were, however, not investigated, and neither was the impact of reprofiling flows.

The aspect of minimizing the resources required to meet the latency targets of token bucket-controlled flows was explored in [25]. The paper relied on service curves with high and low rates and sought to identify the earliest possible time for switching to the lower rate. The focus was, however, on minimizing resources required by each flow individually rather than in aggregate, as in this paper. In addition, the potential impact of reprofiling flows was not addressed.

[9]A similar result was reported in [24].

b) Shaping bulk data transfers: Minimizing bandwidth (cost) through reprofiling (reshaping) flows has been investigated for bulk data transfers where transfer completion times rather than packet-level deadlines are the targets [26]–[31]. The problem stems from non-linear bandwidth costs, *e.g.,* based on the 95^{th} percentile, so that judiciously adjusting (shaping) the transmission rates of bulk transfers can yield significant savings. Rate shaping is, however, at a time-granularity of minutes rather than at the packet-level. The optimization frameworks of those papers are, therefore, not applicable to our problem. Their solutions, can, however, complement ours by leveraging the fluctuations in link utilization inherent in delivering hard, packet-level delay bounds, as we do.

c) Deterministic networking: The deterministic traffic profiles and delay bounds of the TSN and DetNet standards have also given rise to related investigations as documented in recent surveys [32], [33]. In particular, the optimization framework that underlies many of those studies have connections to the problem we address. However, like most prior similar works, traffic profiles are assumed fixed and the impact of reprofiling is not considered.

d) Datacenter solutions: The emergence of traffic profiles and latency targets in datacenter networks motivated [34]. It targets a multi-hop network, but calls on topological properties of typical datacenter networks to collapse its model to a single hop, thereby aligning with the scope of this paper. Similarities extend to considering a static priority scheduler, but traffic profiles differ. Rather than a token bucket, [34] relies on the notion of network "epochs" to bound packet bursts. Delay bounds are then expressed as a function of the network fan-in and a "throughput factor" that reflects the number of transmission opportunities sources can have per network epoch. Also absent from the paper are exploring bandwidth minimization and the potential benefits of reprofiling flows.

e) Reprofiling investigations: Meeting packet-level latency constraints with a static priority scheduler while minimizing costs through reprofiling of flows is the focus of WorkloadCompactor [15]. The reprofiling decisions of [15] are, however, focused on selecting token bucket parameters from among a family of feasible regulators[10] that do not introduce additional delay. In contrast, our reprofiling allows for an added delay that the scheduler must then compensate for. Exploring when and how this trade-off is of benefit is our main contribution and what makes this work *complementary* to the approach from WorkloadCompactor.

Specifically, WorkloadCompactor considers traffic/workload traces for which it seeks to first identify *feasible* token bucket parameter pairs $\langle r, b\rangle$ that result in *zero* access delay for those traces. WorkloadCompactor's main contribution is in realizing that multiple such $\langle r, b\rangle$ pairs are possible (the r-b curve of [15]), and that *"jointly optimizing the choice of* $\langle r, b\rangle$ *rate limit parameters for each workload to better compact workloads onto servers"* can reduce the required server capacity. This is where the contribution of WorkloadCompactor ends, and where that of this paper actually starts.

More precisely, once the optimization of WorkloadCompactor completes, the set of $\langle r, b\rangle$ pairs it produces can be used, together with the associated target latency bounds, as inputs to Proposition 8. Proposition 8 explores, for a static priority scheduler, how to best reprofile token bucket-controlled flows to meet their deadlines with the least bandwidth. This reprofiling is beyond that suggested by WorkloadCompactor[11], and explores how trading-off access delay to further smooth flows can yield additional benefits. In other words, the approach proposed in the paper complements that of WorkloadCompactor, in that it can be applied to any set of token bucket profiles produced by WorkloadCompactor, and modify them to yield further reductions in system resources (bandwidth or server capacity) while still meeting latency targets.

We also note that, because WorkloadCompactor considers the problem of selecting token bucket parameters for traffic traces (to ensure zero access delay), it addresses an aspect that this paper does not consider since we assume that token bucket profiles are given. This is yet another aspect in which the two papers are complementary.

f) Early works: Finally, we note that exploring the trade-off between making traffic smoother and end-to-end performance is not unique to packet networks. It is present in the early "fluctuation smoothing" scheduling policies of [35] that sought to reduce processing time in manufacturing plants, and more recently in the reshaping of parallel I/O requests to improve the scalability of database systems [36].

VIII. Conclusion and Future Work

The paper investigated the question of minimizing the bandwidth needed to guarantee worst case latencies to a set of token bucket-controlled flows sharing a single link. The investigation was carried for schedulers of different complexity.

The paper first characterized the minimum required bandwidth independent of schedulers, and showed that an edf scheduler could realize all flows' deadlines under such bandwidth. Motivated by the need for lower complexity solutions, the paper then explored simpler static priority and fifo schedulers. It derived the minimum required bandwidth for both, but more interestingly established how to optimally reprofile flows to reduce the bandwidth needed while still meeting all deadlines. The relative benefits of such an approach were illustrated numerically for a number of different flow combinations, which showed how reprofiling can enable simpler schedulers to perform nearly as well a more complex ones across a range of configurations.

The obvious direction in which to extend the paper is to a multi-hop setting. In [16], we build on the results of Proposition 1 and provide initial results for the multi-hop case under the assumption that (service curve) edf schedulers are available at each hop. Extending the investigation to static priority and fifo schedulers is under way.

[10]Regulators above the r-b curve using the terminology of [15].

[11]Again [15] focuses on regulators that ensure zero access delay for a given trace, while we investigate how a non-zero access delay can be of benefit.

Another aspect of interest with static priority schedulers is relaxing the assumption that flows with different deadlines map to distinct priority classes, and allow multiple deadlines to be assigned to the same class. Not only does it enhance scalability, but it can also improve performance[12]. Last but not least, extensions to statistical rather than deterministic delay guarantees are also of practical relevance.

REFERENCES

[1] M. Ashjaei, L. L. Bello, M. Daneshtalab, G. Patti, S. Saponara, and S. Mubeen, "Time-sensitive networking in automotive embedded systems: State of the art and research opportunities," *Journal of Systems Architecture*, vol. 117, p. 102137, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1383762121001028

[2] *Avionics Full Duplex Switched Ethernet (AFDX) Network*, Airlines Electronic Engineering Committee, Aircraft Data Network Part 7, ARINC Specification 664, Aeronautical Radio, Annapolis, MD, USA, 2002.

[3] C. Zunino, A. Valenzano, R. Obermaisser, and S. Petersen, "Factory communications at the dawn of the fourth industrial revolution," *Computer Standards & Interfaces*, vol. 71, p. 103433, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0920548919300868

[4] T. Docquier, Y.-Q. Song, V. Chevrier, L. Pontnau, and A. Ahmed-Nacer, "Performance evaluation methodologies for smart grid substation communication networks: A survey," *Computer Communications*, vol. 198, pp. 228–246, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366422004285

[5] (2022) AWS global network. [Online]. Available: https://aws.amazon.com/about-aws/global-infrastructure/global_network/

[6] (2022) Google cloud networking overview. [Online]. Available: https://cloud.google.com/blog/topics/developers-practitioners/google-cloud-networking-overview

[7] (2021, January) Microsoft global network. [Online]. Available: https://docs.microsoft.com/en-us/azure/networking/microsoft-global-network

[8] J. Farkas, L. L. Bello, and C. Gunther, "Time-sensitive networking standards," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 20–21, June 2018.

[9] G. Parsons, "The rise of time-sensitive networking (TSN) in automobiles, industrial automation, and aviation," In Compliance - Electronic Design, Testing & Standards, January 2022, https://incompliancemag.com/article/the-rise-of-time-sensitive-networking-tsn-in-automobiles-industrial-automation-and-aviation/.

[10] Y. Seol, D. Hyeon, J. Min, M. Kim, and J. Paek, "Timely survey of time-sensitive networking: Past and future directions," *IEEE Access*, vol. 9, pp. 142 506–142 527, 2021.

[11] N. Finn, P. Thubert, B. Varga, and J. Farkas, "Deterministic Networking Architecture," RFC 8655, October 2019. [Online]. Available: https://www.rfc-editor.org/info/rfc8655

[12] C. Jones, J. Wilkes, N. Murphy, and C. Smith, *Site Reliability Engineering: How Google Runs Production Systems*, B. Beyer, C. Jones, J. Petoff, and N. Murphy, Eds. O'Reilly Media, 2016, accessible at https://sre.google/sre-book/service-level-objectives/.

[13] (2022) Enterprises are willing to pay for an SLA with guaranteed latency, says new 5G MEC report. [Online]. Available: https://www.vanillaplus.com/2022/06/08/70101-enterprises-are-willing-to-pay-for-an-sla-with-guaranteed-latency-says-new-5g-mec-report/

[14] J.-Y. Le Boudec, "A theory of traffic regulators for deterministic networks with application to interleaved regulators," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, p. 2721–2733, December 2018. [Online]. Available: https://doi.org/10.1109/TNET.2018.2875191

[15] T. Zhu, M. A. Kozuch, and M. Harchol-Balter, *WorkloadCompactor: Reducing Datacenter Cost While Providing Tail Latency SLO Guarantees*. Santa Clara, CA: Association for Computing Machinery, September 2017, p. 598–610. [Online]. Available: https://doi.org/10.1145/3127479.3132245

[16] J. Qiu, J. Song, R. Guérin, and H. Sariowan, "On the benefits of traffic "reprofiling." the multiple hops case – Part I," 2023, under submission.

[17] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Springer, 2001.

[18] L. Georgiadis, R. Guerin, and A. Parekh, "Optimal multiplexing on a single link: delay and buffer requirements," *IEEE Transactions on Information Theory*, vol. 43, no. 5, pp. 1518–1535, September 1997.

[19] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case," *IEEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344–357, June 1993.

[20] A. Sivaraman, S. Subramanian, M. Alizadeh, S. Chole, S.-T. Chuang, A. Agrawal, H. Balakrishnan, T. Edsall, S. Katti, and N. McKeown, "Programmable packet scheduling at line rate," in *Proc. ACM SIGCOMM*, Florianopolis, Brazil, August 2016.

[21] A. Saeed, Y. Zhao, N. Dukkipati, M. Ammar, E. Zegura, K. Harras, and A. Vahdat, "Eiffel: Efficient and flexible software packet scheduling," in *Proc. USENIX NSDI*, Boston, MA, 2019.

[22] N. K. Sharma, C. Zhao, M. Liu, P. G. Kannan, C. Kim, A. Krishnamurthy, and A. Sivaraman, "Programmable calendar queues for high-speed packet scheduling," in *Proc. USENIX NSDI*, Santa Clara, CA, February 2020.

[23] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, "Inside the social network's (datacenter) network," in *Proc. ACM SIGCOMM*, London, United Kingdom, August 2015. [Online]. Available: https://doi.org/10.1145/2785956.2787472

[24] J. Liebeherr, D. E. Wrege, and D. Ferrari, "Exact admission control for networks with a bounded delay service," *IEEE/ACM Transactions on Networking*, vol. 4, no. 6, pp. 885–901, 1996.

[25] J. B. Schmitt, "On the allocation of network service curves for bandwidth/delay-decoupled scheduling disciplines," in *Proc. IEEE GLOBECOM*, vol. 2, Taipei, Taiwan, November 2002.

[26] T. Nandagopal and K. P. Puttaswamy, "Lowering inter-datacenter bandwidth costs via bulk data scheduling," in *Proc. IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, Ottawa, ON, Canada, May 2012.

[27] H. Zhang, K. Chen, W. Bai, D. Han, C. Tian, H. Wang, H. Guan, and M. Zhang, "Guaranteeing deadlines for inter-datacenter transfers," in *Proc. European Conference on Computer Systems (EuroSys)*, Bordeaux, France, 2015. [Online]. Available: https://doi.org/10.1145/2741948.2741957

[28] V. Jalaparti, I. Bliznets, S. Kandula, B. Lucier, and I. Menache, "Dynamic pricing and traffic engineering for timely inter-datacenter transfers," in *Proc. ACM SIGCOMM*, Florianopolis, Brazil, August 2016. [Online]. Available: https://doi.org/10.1145/2934872.2934893

[29] W. Li, X. Zhou, K. Li, H. Qi, and D. Guo, "TrafficShaper: Shaping inter-datacenter traffic to reduce the transmission cost," *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1193–1206, June 2018.

[30] Z. Yang, Y. Cui, X. Wang, Y. Liu, M. Li, S. Xiao, and C. Li, "Cost-efficient scheduling of bulk transfers in inter-datacenter WANs," *IEEE/ACM Transactions on Networking*, vol. 27, no. 5, pp. 1973–1986, 2019.

[31] R. Singh, S. Agarwal, M. Calder, and P. Bahl, "Cost-effective cloud edge traffic engineering with cascara," in *Proc. USENIX NSDI*, virtual, 2021. [Online]. Available: https://www.usenix.org/conference/nsdi21/presentation/singh

[32] T. Stüber, L. Osswald, S. Lindner, and M. Menth, "A survey of scheduling in time-sensitive networking (TSN)," 2022. [Online]. Available: https://arxiv.org/abs/2211.10954

[33] J. Walrand, "A concise tutorial on traffic shaping and scheduling in time-sensitive networks," *IEEE Communications Surveys & Tutorials*, pp. 1–1, May 2023.

[12] See Appendix E for the simple case of a two-flow configuration.

[34] M. P. Grosvenor, M. Schwarzkopf, I. Gog, R. N. M. Watson, A. W. Moore, S. Hand, and J. Crowcroft, “Queues Don’t matter when you can JUMP them!” in *Proc. USENIX NSDI*, Oakland, CA, May 2015. [Online]. Available: https://www.usenix.org/conference/nsdi15/technical-sessions/presentation/grosvenor

[35] S. Lu, D. Ramaswamy, and P. Kumar, “Efficient scheduling policies to reduce mean and variance of cycle-time in semiconductor manufacturing plants,” *IEEE Transactions on Semiconductor Manufacturing*, vol. 7, no. 3, pp. 374–388, 1994.

[36] N. Li, H. Jiang, H. Che, Z. Wang, and M. Q. Nguyen, “Improving scalability of database systems by reshaping user parallel I/O,” in *Proc. EuroSys*, Rennes, France, April 2022. [Online]. Available: https://doi.org/10.1145/3492321.3519570

Appendix A
Summary of Notation Used in the Paper

Notation	**Definition**
n	number of flows inside the network
b_i	token bucket size of flow i
$\boldsymbol{b}$	vector of token bucket sizes across n flows: $(b_1, b_2, \ldots, b_n)$
b'_i	reprofiled token bucket size of flow i
$\boldsymbol{b}'$	vector for all reprofiled token bucket sizes across n flows: $(b'_1, b'_2, \ldots, b'_n)$
b'^*_i	optimal bucket size of flow i's reprofiler under static priority
$\boldsymbol{b}'^*$	vector for all optimal reprofiled token bucket sizes across n flows: $(b'^*_1, b'^*_2, \ldots, b'^*_n)$ under static priority
$\widehat{b}'^*_i$	optimal bucket size of flow i's reprofiler under fifo
$\widehat{\boldsymbol{b}}'^*$	vector for all optimal reprofiled token bucket sizes across n flows: $(\hat{b}'^*_1, \hat{b}'^*_2, \ldots, \hat{b}'^*_n)$ under fifo
B'_i	cumulative reprofiled token bucket size for flows with a priority no smaller than i, *i.e.,* $\sum_{j=i}^{n} b'_j$
$\widehat{B}_i$	cumulative token bucket burst size for flows 1 to i, *i.e.,* $\sum_{j=1}^{i} b_j$
$\widehat{B}'_i$	cumulative reprofiled token bucket size for flows from 1 to i, *i.e.,* $\sum_{j=1}^{i} b'_j$
d_i	end-to-end deadline for flow i
$\boldsymbol{d}$	vector of end-to-end deadlines for n flows: $(d_1, d_2, \ldots, d_n)$
D^*_i	worst-case end-to-end delay for flow i under priority+reprofiling
$\widehat{D}^*_i$	worst-case end-to-end delay for flow i under fifo+reprofiling
r_i	token bucket rate of flow i
$\boldsymbol{r}$	vector for all token bucket rates across n flows: $(r_1, r_2, \ldots, r_n)$
(r_i, b_i)	traffic profile of flow i
R_i	cumulative token bucket rates for flows with a priority no smaller than i, *i.e.,* $\sum_{j=i}^{n} r_j$
R	shared link bandwidth
R^*	optimal minimum required bandwidth
$\widetilde{R}^*$	minimum required link bandwidth in the absence of reprofiling under static priority
$\widetilde{R}^*_R$	minimum required link bandwidth with reprofiling under static priority
$\widehat{R}^*$	minimum required link bandwidth in the absence of reprofiling under fifo
$\widehat{R}^*_R$	minimum required link bandwidth with reprofiling under fifo
t	time
H_i	$b_i - d_i r_i$
$\Pi_i(R)$	$\frac{r_i + R - R_{i+1}}{R - R_{i+1}}$
$V_i(R)$	$d_i(R - R_{i+1} - b_i)$
$\mathbb{S}_1(R)$	$V_1(R)$
$\mathbb{S}_i(r)$	$\mathbb{S}_{i-1}(R) \bigcup V_i(R) \bigcup \left\{ \frac{s - H_i}{\Pi_i(R)} \mid s \in \mathbb{S}_{i-1}(R) \right\}$
$\mathbb{Z}_i$	the set of integers from 1 to i, *i.e.,* $\{1 \leq i \leq j \mid j \in \mathbb{Z}\}$
$X_F(R)$	$\max_{P_1, P_2 \subseteq \mathbb{Z}_n, P_2 \neq \mathbb{Z}_n, P_1 \bigcap P_2 = \emptyset} \frac{\sum_{i \in P_1} \frac{RH_i}{R + r_i} + \sum_{i \in P_2} \left(b_i - \frac{r_i d_i R}{R_1} \right)}{1 - \sum_{i \in P_1} \frac{r_i}{R + r_i} - \sum_{i \in P_2} \frac{r_i}{R_1}}$
$Y_F(R)$	$\min_{1 \leq i \leq n-1} \left\{ \widehat{B}_n, Rd_n, \min_{P_1, P_2 \subseteq \mathbb{Z}_i, P_1 \bigcap P_2 = \emptyset, P_1 \bigcup P_2 \neq \emptyset} \left\{ \frac{\widehat{B}_i - \sum_{j \in P_1} \frac{RH_j}{R + r_j} - \sum_{j \in P_2} \left(b_j - \frac{r_j d_j R}{R_1} \right)}{\sum_{j \in P_1} \frac{r_j}{R + r_j} + \sum_{j \in P_2} \frac{r_j}{R_1}} \right\} \right\}$
$T_i(\widehat{B}'_n, R)$	$\max \left\{ 0, \frac{R}{R + r_i} \left(H_i + \frac{r_i}{R} \widehat{B}'_n \right), b_i + \frac{r_i (\widehat{B}'_n - Rd_i)}{R_1} \right\}$
Γ_{sc}	a service curve assignment that gives each flow i a service curve of $SC_i(t)$
OPT_❑	general optimization where **❑=S,F** for static priority and fifo
OPT_R❑	optimization with reprofiling where **❑=S,F** for static priority and fifo

APPENDIX B
PROOFS FOR DYNAMIC PRIORITY SCHEDULER

A. Proof for Proposition 1

For the reader's convenience, we restate Proposition 1.

PROPOSITION 1. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has a traffic profile (r_i, b_i) and a deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Consider a service-curve assignment Γ_{sc} that allocates flow i a service curve of*

$$SC_i(t) = \begin{cases} 0 & \text{when } t < d_i, \\ b_i + r_i(t - d_i) & \text{otherwise}. \end{cases} \tag{1}$$

Then

1) *For any flow $i, 1 \leq i \leq n$, $SC_i(t)$ ensures a worst-case end-to-end delay no larger than d_i.*
2) *Realizing Γ_{sc} requires a link bandwidth of at least*

$$R^* = \max_{1 \leq h \leq n} \left\{ \sum_{i=1}^{n} r_i, \frac{\sum_{i=h}^{n} b_i + r_i(d_h - d_i)}{d_h} \right\}. \tag{2}$$

3) *Any scheduling mechanism capable of meeting all the flows' deadlines requires a bandwidth of at least R^*.*

Proof. We first show that under Γ_{sc} each flow meets its deadline, and then show that a bandwidth of R^* is enough to accommodate all the service curves defined in Γ_{sc}. Next, we show that no mechanism exists than can meet the deadlines with a bandwidth strictly smaller than R^*.

- For any flow $1 \leq i \leq n$, its token bucket constrained arrival curve is of the form

$$AC_i(t) = \begin{cases} 0 & \text{when } t = 0 \\ b_i + r_i t & \text{otherwise}. \end{cases}$$

  Combining it with flow i's service curve $SC_i(t)$, the worst-case end-to-end delay for flow i is of the form[13],

$$D_i^* = \sup_{t \geq 0} \inf_{\tau \geq 0} \{AC_i(t) \leq SC_i(t + \tau)\} = d_i.$$

- To accommodate all the service curves in Γ_{sc}, the system needs a bandwidth R such that $Rt \geq \sum_{i=1}^{n} SC_i(t)$ for all $t > 0$, *i.e.,*

$$R \geq \sup_{t > 0} \frac{\sum_{i=1}^{n} SC_i(t)}{t}. \tag{16}$$

  Towards establishing that the minimum link bandwidth $R^* = \sup_{t>0} \frac{\sum_{i=1}^{n} SC_i(t)}{t}$ is captured by Eq. (2), we first introduce another proposition:

  Proposition 12. *Assume that $PL(t)$ is a wide-sense increasing, piecewise linear, and right continuous function defined by a finite set of k linear segments:*

$$pl_i(t) = c_i + s_i t, \; t \in [t_i, t_{i+1}), \; 1 \leq i \leq k$$

  where $s_i \geq 0$ and c_i are the slopes and intercepts of segment $pl_i(t)$ respectively, and $t_i < t_{i+1}$, $\forall i, 1 \leq i \leq k$, with $t_1 = 0$ and $t_{k+1} = \infty$. Furthermore, $PL(0) = 0$ so that $c_1 = 0$.
  Then, to compute $\sup_{t>0} \frac{PL(t)}{t}$, it is sufficient to consider values of $\frac{PL(t)}{t}$ at the following times t:
  1) *$t \to t_{k+1} = \infty$*
  2) *interval boundaries $t_i, \; 2 \leq i \leq k$, when one of the following conditions is met:*
     a) *$PL(t)$ is continuous and the slope of $PL(t)$ decreases,* i.e., $s_{i-1} > s_i$.
     b) *$PL(t)$ is discontinuous.*

  Proof. We readily know that to find $\sup_{t>0} \frac{PL(t)}{t}$, it is sufficient to consider interval boundaries t_is, since $\frac{pl_i(t)}{t} = \frac{c_i}{t} + s_i$, $\frac{PL(t)}{t}$ is a decreasing (increasing) function of t within any interval $[t_i, t_{i+1})$ when $c_i \geq 0$ ($c_i < 0$).
  Consider first an intermediate boundary $t_i, 2 \leq i \leq k$, we first argue that if $PL(t)$ is *continuous*, then $\sup_{t>0} \frac{PL(t)}{t}$ cannot be achieved at any boundary t_i for which $s_{i-1} < s_i$, *i.e.,* a boundary where the slope of $PL(t)$ increases. Towards establishing the result, consider the following two cases:

[13] See THEOREM 1.4.2 in [17], page 23.

1) $c_i < 0$. In this case, irrespective of s_i, $\frac{PL(t)}{t}$ is increasing in $[t_i, t_{i+1})$ and $\sup_{t\in[t_i,t_{i+1})} \frac{PL(t)}{t} = \frac{PL(t_{i+1}^-)}{t_{i+1}^-}$, so that $\sup_{t>0} \frac{PL(t)}{t}$ cannot be achieved at t_i.
2) $c_i \geq 0$. Since we have assumed that $PL(t)$ is continuous, we have that $pl_{i-1}(t_i) = c_{i-1} + s_{i-1}t_i = pl_i(t_i) = c_i + s_i t_i$. This implies $c_{i-1} = c_i + (s_i - s_{i-1})t_i > c_i \geq 0$ since $s_i > s_{i-1}$. Since both c_i and c_{i-1} are non-negative, $\frac{PL(t)}{t}$ is a decreasing function throughout $[t_{i-1}, t_{i+1})$. Hence, $\sup_{t>0} \frac{PL(t)}{t}$ cannot be realized at t_i.

Turning next to the case where $PL(t)$ is discontinuous at some t_i, then $\lim_{t\to t_i^-} \frac{PL(t)}{t} < \lim_{t\to t_i^+} \frac{PL(t)}{t}$ since $PL(t)$ is wide-sense increasing. Since $PL(t)$ is right continuous, $\frac{PL(t_i)}{t_i} = \lim_{t\to t_i^+} \frac{PL(t)}{t}$, which implies it is sufficient to consider t_i for computing the supremum.

Finally, we consider the two extreme boundaries $t_1 = 0$ and $t_{k+1} = \infty$. For $t_1 = 0$, since $c_1 = 0$, we have $\frac{PL(t)}{t} = s_1, \forall\, t \in [t_1 = 0, t_2)$. Since t_2 is one of the intermediate boundaries covered by the first part of our proof, we do not need to consider t_1 when computing the supremum of $\frac{PL(t)}{t}$ For $t_{k+1} = \infty$, $\lim_{t\to\infty} \frac{PL(t)}{t} = s_k$.

Combining the fact that $\sup_{t>0} \frac{PL(t)}{t}$ can only be realized at an interval boundary with the above establishes that it is realized at either an interval boundary t_i where $PL(t)$ is continuous and experiencing a slope decrease, *i.e.,* $s_{i-1} > s_i$, or at an interval boundary where $PL(t)$ is discontinuous, or at t_{k+1}, *i.e.,* $t \to \infty$. □

Returning to the proof of Proposition 1, the aggregate service curve $\sum_{i=1}^n SC_i(t)$ is a wide-sense increasing, piecewise linear function since each $SC_i(t)$ is, by definition, wide-sense increasing and piecewise linear. From Proposition 12 we know that the supremum of $\frac{\sum_{i=1}^n SC_i(t)}{t}$ is either $\lim_{t\to\infty} \frac{\sum_{i=1}^n SC_i(t)}{t} = \sum_{i=1}^n r_i$ or is achieved at a boundary value $d_k \in \{d_1, d_2, \ldots, d_n\}$. Combining this with the expressions for the individual service curves $SC_i(t), i = 1, \ldots, n$, gives

$$R \geq \max_{1\leq h\leq n} \left\{ \sum_{i=1}^n r_i,\ \frac{\sum_{i=h}^n b_i + r_i(d_h - d_i)}{d_h} \right\} = R^*.$$

Thus, R^* is sufficient to accommodates all service curves in the service curves assignment Γ_{sc}.

- Next, we show that R^* is a lower bound for the minimum required bandwidth. Note that $R^* \geq \sum_{i=1}^n r_i$, so that if the deadlines can be met with $R^* = \sum_{i=1}^n r_i$ no improvement is feasible. Below we consider the case when $R^* > \sum_{i=1}^n r_i$, *i.e.,* there exists $1 \leq \hat{h} \leq n$, such that $R^* = \frac{\sum_{i=\hat{h}}^n b_i + r_i(d_{\hat{h}} - d_i)}{d_{\hat{h}}} > \sum_{i=1}^n r_i$

  Suppose there exists a mechanism achieving a minimum required bandwidth $R' < R^*$. Next we construct an arrival pattern consistent with each flow's token bucket arrival constraints, such that R' cannot satisfy all flows' deadlines.

  Consider the arrival pattern such that for all $1 \leq i \leq n$, flow i sends b_i at $t = 0$ (where the system restarts the clock), and then constantly sends at a rate of r_i. By time $d_{\hat{h}}$, to satisfy the deadlines of all flows i with $d_i \leq d_{\hat{h}}$, *i.e.,* $i \geq \hat{h}$, the link must have transmitted at least $b_i + r_i(d_{\hat{h}} - d_i)$ amount of data for each such flow. Consequently, by $d_{\hat{h}}$ the link should have cumulatively transmitted at least $\sum_{i=\hat{h}}^n b_i + r_i(d_{\hat{h}} - d_i)$. As $\sum_{i=\hat{h}}^n b_i + r_i(d_{\hat{h}} - d_i) = d_{\hat{h}}R^* > d_{\hat{h}}R'$, a bandwidth of R' must violate some flows' deadlines.

□

B. *Proof for Proposition 2*

Proposition 2. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > \ldots > d_n$ and $d_1 < \infty$. The earliest deadline first (EDF) scheduler realizes Γ_{sc} under a link bandwidth of R^*.*

Proof. We first show that EDF satisfies Γ_{sc}, and then shows that EDF requires a minimum required bandwidth of R^*.

We show that EDF satisfies Γ_{sc} by contradiction. Suppose EDF cannot achieve Γ_{sc}. Then there exists $\hat{t} \geq d_i$ and $1 \leq i \leq n$, such that $\widehat{SC}_i(\hat{t}) < SC_i(\hat{t})$, where $\widehat{SC}_i$ is the service curve that EDF assigns to flow i. To satisfy flow i's deadline, at $\hat{t} - d_i$ EDF should yield a virtual delay no large than d_i, *i.e.,*

$$\inf_{\tau\geq 0} \left\{ b_i + r_i(\hat{t} - d_i) \leq \widehat{SC}_i(\hat{t} - d_i + \tau) \right\} \leq d_i,$$

which then gives $\widehat{SC}_i(\hat{t} - d_i + d_i) \geq b_i + r_i(\hat{t} - d_i)$. As $b_i + r_i(\hat{t} - d_i) = SC_i(\hat{t})$, this contradicts the assumption that $\widehat{SC}_i(\hat{t}) < SC_i(\hat{t})$.

Next, we show that EDF requires a minimum bandwidth of R^*. Suppose flow i's data sent at t has a deadline of $(t + d_i)$. We show that EDF satisfies all flows' deadlines with a bandwidth of $R^* = \max_{1\leq h\leq n} \left\{ \sum_{i=1}^n r_i, \frac{\sum_{i=h}^n b_i + r_i(d_h - d_i)}{d_h} \right\}$. Based on the utilization of the shared link, we consider two cases separately, where in both cases we prove the result by contradiction. Specifically, suppose that under EDF, R^* cannot satisfy all flows' latency requirements. Then there exists $1 \leq \hat{h} \leq n$ and $\hat{t} \geq 0$, such that EDF processes at least one bit sent by flow $\hat{h}$ at time $\hat{t}$ after time $(\hat{t} + d_{\hat{h}})$. We consider first the case where the shared link uses up all its bandwidth during the period $[0, \hat{t} + d_{\hat{h}}]$ to transmit data with absolute deadlines no larger than $(\hat{t} + d_{\hat{h}})$, and then consider the case where there exists $t_0 \in [0, \hat{t} + d_{\hat{h}}]$, such that at t_0 the shared link is not busy with data whose absolute deadline is no larger than $(\hat{t}+d_{\hat{h}})$. Showing that R^* is enough for EDF to meet all flows' deadlines, establishes the result.

1) Consider the case where for all $t \in [0, \hat{t}+d_{\hat{h}}]$ the shared link uses up all its bandwidth to send bits with an absolute deadline no larger than $(\hat{t}+d_{\hat{t}})$. Then by $(\hat{t}+d_{\hat{h}})$ the shared link cumulatively has processed $R^*(\hat{t}+d_{\hat{h}})$ amount of data that all have deadlines no larger than $(\hat{t}+d_{\hat{t}})$. From the fact that EDF violates an absolute deadline of $(\hat{t}+d_{\hat{h}})$, we know that there exists an arrival pattern consistent with the token bucket constraints, such that cumulatively flows send more than $R^*(\hat{t}+d_{\hat{h}})$ amount of data with absolute deadlines no larger than $(\hat{t}+d_{\hat{h}})$. From the token bucket constraints, we know that by $(\hat{t}+d_{\hat{h}})$, flows can send at most $\sum_{i=1}^n \left[b_i + r_i(\hat{t}+d_{\hat{h}}-d_i)\right] \mathbf{I}_{\hat{\mathbf{t}}+\mathbf{d}_{\hat{\mathbf{h}}}-\mathbf{d_i}\geq \mathbf{0}}$ amount of data whose absolute deadline is at most $(\hat{t}+d_{\hat{h}})$. Therefore, we have $R^*(\hat{t}+d_{\hat{h}}) < \sum_{i=1}^n \left[b_i + r_i(\hat{t}+d_{\hat{h}}-d_i)\right] \mathbf{I}_{\hat{\mathbf{t}}+\mathbf{d}_{\hat{\mathbf{h}}}-\mathbf{d_i}\geq \mathbf{0}}$. Define $d_0 = \infty$. There exists $1 \leq \hat{n} \leq n$ such that $\hat{t}+d_{\hat{h}} \in [d_{\hat{n}}, d_{\hat{n}-1})$ so that
$$R^* < \frac{\sum_{i=1}^n \left[b_i + r_i(\hat{t}+d_{\hat{h}}-d_i)\right] \mathbf{I}_{\hat{\mathbf{t}}+\mathbf{d}_{\hat{\mathbf{h}}}-\mathbf{d_i}\geq \mathbf{0}}}{\hat{t}+d_{\hat{h}}} = \sum_{i=\hat{n}}^n \left(r_i + \frac{b_i - r_i d_i}{\hat{t}+d_{\hat{h}}} \right) := R'.$$
If $\sum_{i=\hat{n}}^n b_i - r_i d_i \leq 0$, we have $R^* < R' \leq \sum_{i=1}^n r_i$, which contradicts to $R^* > \sum_{i=1}^n r_i$. Hence we consider only $\sum_{i=\hat{n}}^n b_i - r_i d_i > 0$, where R' decreases with $(\hat{t}+d_{\hat{h}})$. Define $\hat{R}(u) = \sum_{i=\hat{n}}^n \left(r_i + \frac{b_i - r_i d_i}{u}\right)$. We then have
$$R^* < R' \leq \hat{R}(d_{\hat{n}}) = \frac{\sum_{i=\hat{n}}^n \left[b_i + r_i(d_{\hat{n}} - d_i)\right]}{d_{\hat{n}}},$$
which contradicts to the definition of R^*.
2) Otherwise, the shared link uses less than all its bandwidth at $t_0 \in [0, \hat{t}+d_{\hat{h}}]$, and uses up all its bandwidth for all $t \in (t_0, \hat{t}+d_{\hat{h}}]$ to send bits with an absolute deadline no larger than $(\hat{t}+d_{\hat{t}})$. Then during $(t_0, \hat{t}+d_{\hat{h}}]$ the shared link processes $R^*(\hat{t}+d_{\hat{h}}-t_0)$ amount of data with absolute deadlines no larger than $(\hat{t}+d_{\hat{h}})$, and flows send strictly more than $R^*(\hat{t}+d_{\hat{h}}-t_0)$ amount of data with absolute deadlines no larger than $(\hat{t}+d_{\hat{h}})$. From the token bucket constraints, we know that during $(t_0, \hat{t}+d_{\hat{h}}]$ flows can send at most $\sum_{i=1}^n \left[b_i + r_i(\hat{t}+d_{\hat{h}}-t_0-d_i)\right] \mathbf{I}_{\hat{\mathbf{t}}+\mathbf{d}_{\hat{\mathbf{h}}}-\mathbf{t_0}-\mathbf{d_i}\geq \mathbf{0}}$ amount of data whose absolute deadlines are no larger than $(\hat{t}+d_{\hat{h}})$. Thus we have $R^* < \frac{\sum_{i=1}^n \left[b_i + r_i(\hat{t}+d_{\hat{h}}-t_0-d_i)\right] \mathbf{I}_{\hat{\mathbf{t}}+\mathbf{d}_{\hat{\mathbf{h}}}-\mathbf{t_0}-\mathbf{d_i}\geq \mathbf{0}}}{d_{\hat{h}}+\hat{t}-t_0}$, which, as before, contradicts the definition of R^*.

□

C. Proof for Proposition 3

PROPOSITION 3. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Reprofiling flows will not decrease the minimum bandwidth required to meet the flows' deadlines.*

Proof. We show that adding a reprofiler to any of the flow does not decrease the optimal minimum required bandwidth.

Suppose we apply a reprofiler b_i' to flow i, where $1 \leq i \leq n$ and $b_i' \geq b_i - d_i r_i$. Then flow i incurs a reshaping delay of $\frac{b_i - b_i'}{r_i}$. This then leaves a maximum in-network delay of $d_i - \frac{b_i - b_i'}{r_i}$, which is greater than 0 since $b_i' \geq b_i - d_i r_i$. Hence, to meet its deadline flow i requires a service curve of at least
$$SC_i'(t) = \begin{cases} 0, & \text{when } t < d_i - \dfrac{b_i - b_i'}{r_i} \\ b_i' + r_i \left(t - d_i + \dfrac{b_i - b_i'}{r_i} \right) = b_i + r_i(t - d_i), & \text{otherwise} \end{cases}$$
which is greater than $SC_i(t) = \begin{cases} 0, & \text{when } t < d_i \\ b_i + r_i(t-d_i), & \text{otherwise} \end{cases}$. Consequently, according to Proposition 1 the system needs a bandwidth of at least
$$R = \max\left\{ \sum_{i=1}^n r_i, \sup_{t\geq 0} \frac{\sum_{j\neq i} SC_j(t) + SC_i'(t)}{t} \right\} \geq \max\left\{ \sum_{i=1}^n r_i, \sup_{t\geq 0} \frac{\sum_{1\leq j\leq n} SC_j(t)}{t} \right\} = R^*_{sc}, \tag{17}$$
to meet each flow's deadline.

□

Appendix C
Proofs for Static Priority Scheduler

A. *Proof for Proposition 4*

We actually prove Proposition 4 under the more general packet-based model. By assuming that all packets have a length of 0, the packet-based model defaults to the fluid model.

Proposition 4. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i) and deadline d_i, with $d_1 > d_2 > ... > d_n$ and $d_1 < \infty$. Under a static-priority scheduler, there exists an assignment of flows to priorities that minimizes link bandwidth while meeting all flows deadlines such that flow i is assigned a priority strictly greater than that of flow j only if $d_i < d_j$.*

Proof. For a mechanism Γ, denote flows with priority h under Γ as $G_h(\Gamma)$. Define $d_h^{(max)}(\Gamma) = \max_{i \in G_h(\Gamma)} d_i$ and $d_h^{(min)}(\Gamma) = \min_{i \in G_h(\Gamma)} d_i$. Assume priority class $h+1$ has a higher priority than priority class h. Then we will prove the proposition by induction on the number k of priority classes. Our induction hypothesis $S(k)$ is expressed in the following statement:

$S(k)$: For link shared by any number of flows, there exists an optimal k-priority mechanism Γ_k such that $\forall s < l \leq k$, $d_l^{(max)}(\Gamma_k) < d_s^{(min)}(\Gamma_k)$.

- **Base case:** consider the case when $k = 2$. We show that for any mechanism Γ_2, if $d_1^{(min)}(\Gamma_2) < d_2^{(max)}(\Gamma_2)$, then there exists a 2-priority mechanism Γ_2' such that $d_1^{(min)}(\Gamma_2') > d_2^{(max)}(\Gamma_2')$ and $R^*(\Gamma_2') \leq R^*(\Gamma_2)$.
For mechanism Γ_2, denote $l_1^{(max)}(\Gamma_2)$ to be the maximum packet size for flows in $G_1(\Gamma_2)$. To satisfy each flow's deadline, it requires a bandwidth R such that
$$\begin{cases} \dfrac{\sum_{i \in G_2(\Gamma_2)} b_i + l_1^{(max)}(\Gamma_2)}{R} \leq d_2^{(min)}(\Gamma_2), \\ \dfrac{\sum_i b_i}{R - \sum_{i \in G_2(\Gamma_2)} r_i} \leq d_1^{(min)}(\Gamma_2), \\ \displaystyle\sum_{i=1}^{n} r_i \leq R, \end{cases} \tag{18}$$
which gives
$$R^*(\Gamma_2) = \max\left\{ \sum_{i=1}^{n} r_i, \frac{\sum_{i \in G_2(\Gamma_2)} b_i + l_1^{(max)}(\Gamma_2)}{d_2^{(min)}(\Gamma_2)}, \frac{\sum_i b_i}{d_1^{(min)}(\Gamma_2)} + \sum_{i \in G_2(\Gamma_2)} r_i \right\}.$$
Define $G_1'(\Gamma_2) = \{i \in G_2(\Gamma_2) \mid d_i > d_1^{(max)}(\Gamma_2)\}$ and $G_2'(\Gamma_2) = G_2(\Gamma_2) - G_1'(\Gamma_2)$. Consider the mechanism Γ_2' such that $G_2(\Gamma_2') = G_2'(\Gamma_2)$, and $G_1(\Gamma_2') = G_1'(\Gamma_2) \cup G_1(\Gamma_2)$. Note that $d_1^{(min)}(\Gamma_2') > d_2^{(max)}(\Gamma_2')$, and note also that $d_i^{(min)}(\Gamma_2') = d_i^{(min)}(\Gamma_2), i = 1, 2$. Similar as Eq. (18), we have
$$R^*(\Gamma_2') = \max\left\{ \sum_{i=1}^{n} r_i, \frac{\sum_{i \in G_2(\Gamma_2')} b_i + l_1^{(max)}(\Gamma_2')}{d_2^{(min)}(\Gamma_2)}, \frac{\sum_i b_i}{d_1^{(min)}(\Gamma_2)} + \sum_{i \in G_2(\Gamma_2) - G_1'(\Gamma_2)} r_i \right\}$$
Note that $\frac{\sum_i b_i}{d_1^{(min)}(\Gamma_2)} + \sum_{i \in G_2(\Gamma_2)} r_i \geq \frac{\sum_i b_i}{d_1^{(min)}(\Gamma_2)} + \sum_{i \in G_2(\Gamma_2) - G_1'(\Gamma_2)} r_i$. Next we show $\sum_{i \in G_2(\Gamma_2)} b_i + l_1^{(max)}(\Gamma_2) \geq \sum_{i \in G_2(\Gamma_2')} b_i + l_1^{(max)}(\Gamma_2')$, from which we then have $R^*(\Gamma_2) \geq R^*(\Gamma_2')$, and therefore $S(2)$. Since $G_1(\Gamma_2) \subsetneqq G_1(\Gamma_2')$, it has $l_1^{(max)}(\Gamma_2') \geq l_1^{(max)}(\Gamma_2)$.
  - When $l_1^{(max)}(\Gamma_2') = l_1^{(max)}(\Gamma_2)$, from $G_2(\Gamma_2') \subsetneqq G_2(\Gamma_2)$ we have $\sum_{i \in G_2(\Gamma_2)} b_i \geq \sum_{i \in G_2(\Gamma_2')} b_i$, and therefore have $\sum_{i \in G_2(\Gamma_2)} b_i + l_1^{(max)}(\Gamma_2) \geq \sum_{i \in G_2(\Gamma_2')} b_i + l_1^{(max)}(\Gamma_2')$.
  - When $l_1^{(max)}(\Gamma_2') > l_1^{(max)}(\Gamma_2)$, *i.e.,* there exists a flow $\hat{i} \in G_1'(\Gamma_2)$ such that $l_{\hat{i}} > l_1^{(max)}(\Gamma_2)$. From $b_{\hat{i}} \geq l_{\hat{i}}$ and $G_2(\Gamma_2') \subseteq G_2(\Gamma_2) - \hat{i}$, we have
$$\begin{aligned} \sum_{i \in G_2(\Gamma_2)} b_i + l_1^{(max)}(\Gamma_2) &= \sum_{i \in G_2(\Gamma_2) - \hat{i}} b_i + b_{\hat{i}} + l_1^{(max)}(\Gamma_2) \geq \sum_{i \in G_2(\Gamma_2')} b_i + l_{\hat{i}} + l_1^{(max)}(\Gamma_2) \\ &> \sum_{i \in G_2(\Gamma_2')} b_i + l_1^{(max)}(\Gamma_2') \end{aligned}$$
- **Induction Step:** Let $k \geq 2$ and suppose $S(k)$ holds. Next, we show that $S(k+1)$ holds.
Consider any (k+1)-priority mechanism Γ_{k+1}. For $1 \leq h \leq (1+k)$, denote $l_h^{(max)}(\Gamma_{k+1})$ to be the maximum packet size for all flows with priority strictly smaller than h, and define $l_h^{(max)}(\Gamma_{k+1})$ to be 0 if no flow has a priority strictly smaller

than h. Define $B_h^{(k)}(\Gamma_{k+1}) = \sum_{j=h}^{k}\sum_{i\in G_j(\Gamma_{k+1})} b_i$, *i.e.,* the sum of bucket sizes for flows with priority $h \le j \le k$, and $R_h^{(k)}(\Gamma_{k+1}) = \sum_{j=h}^{k}\sum_{i\in G_j(\Gamma_{k+1})} r_i$, *i.e.,* the sum of rates for flows with priority in $h \le j \le k$. Then to satisfy each flow's deadline Γ_{k+1} requires a bandwidth R satisfying

$$\begin{cases} \frac{B_h^{(k+1)}(\Gamma_{k+1})+l_h^{(max)}(\Gamma_{k+1})}{R-R_{h+1}^{(k+1)}(\Gamma_{k+1})} \le d_h^{(min)}(\Gamma_{k+1}), \quad \forall h \in [1, k+1]; \\ \\ \sum_{i=1}^{n} r_i \le R; \end{cases} \tag{19}$$

which gives a minimum required bandwidth of

$$R^*(\Gamma_{k+1}) = \max_{1\le h\le k+1}\left\{\sum_{i=1}^{n} r_i,\ \frac{B_h^{(k+1)}(\Gamma_{k+1})+l_h^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})} + R_{h+1}^{(k+1)}(\Gamma_{k+1})\right\}. \tag{20}$$

Afterwards, we first show that there exists a $(k+1)$-priority mechanism Γ'_{k+1} satisfying

– **Condition 1**: $G_{k+1}(\Gamma'_{k+1}) = \{i \mid d_n \le d_i < d_{\hat{n}}\}$ where $1 \le \hat{n} \le n$, and $R^*(\Gamma'_{k+1}) \le R^*(\Gamma_{k+1})$, where $G_{k+1}(\Gamma'_{k+1}) = \emptyset$ if $\hat{n} = n$.

After that, under a slight abuse of notation, we show that for any Γ_{k+1} satisfying Condition 1, there exists a $(k+1)$-priority mechanism Γ'_{k+1} satisfying

– **Condition 2**: $R^*(\Gamma'_{k+1}) \le R^*(\Gamma_{k+1})$, and $d_i^{(max)}(\Gamma'_{k+1}) < d_j^{(min)}(\Gamma'_{k+1})$ for all $j < i \le k+1$.

Combining them gives $S(k+1)$.

1) We first show the existence of a mechanism Γ'_{k+1} satisfying condition 1. If Γ_{k+1} satisfies Condition 1, then $\Gamma_{k+1} = \Gamma'_{k+1}$. Otherwise, for all $1 \le \hat{n} < n$, $G_{k+1}(\Gamma_{k+1}) \ne \{i \mid d_n \le d_i < d_{\hat{n}}\}$. Define $\hat{i} = \max\{1 \le i \le n \mid i \notin G_{k+1}(\Gamma_{k+1})\}$ and suppose $\hat{i} \in G_{\hat{h}}(\Gamma_{k+1})$. Further define $G'_{k+1} = \{i \in G_{k+1}(\Gamma_{k+1}) \mid d_i < d_{\hat{h}}^{(min)}(\Gamma_{k+1})\}$ and $G'_{\hat{h}} = G_{k+1}(\Gamma_{k+1}) - G'_{k+1}$.
Consider the mechanism Γ'_{k+1} such that 1) $G_{k+1}(\Gamma'_{k+1}) = G'_{k+1}$, 2) $G_{\hat{h}}(\Gamma'_{k+1}) = G'_{\hat{h}} + G_{\hat{h}}(\Gamma_{k+1})$, and 3) $G_i(\Gamma'_{k+1}) = G_i(\Gamma_{k+1})$, when $1 \le i \le k$ and $i \ne \hat{h}$. Note that
   – when $h \le \hat{h}$, $B_h^{(k+1)}(\Gamma'_{k+1}) = B_h^{(k+1)}(\Gamma_{k+1})$, $R_h^{(k+1)}(\Gamma'_{k+1}) = R_h^{(k+1)}(\Gamma_{k+1})$, and $l_h^{(max)}(\Gamma'_{k+1}) = l_h^{(max)}(\Gamma_{k+1})$;
   – when $h > \hat{h}$, $B_h^{(k+1)}(\Gamma'_{k+1}) = B_h^{(k+1)}(\Gamma_{k+1}) - \sum_{i\in G'_{\hat{h}}} b_i$, $R_h^{(k+1)}(\Gamma'_{k+1}) = R_h^{(k+1)}(\Gamma'_{k+1}) - \sum_{i\in G'_{\hat{h}}} r_i$, and $l_h^{(max)}(\Gamma'_{k+1}) = \max\left\{l_h^{(max)}(\Gamma_{k+1}),\ \max_{i\in G'_{\hat{h}}} l_i\right\}$.

Next we show that $R^*(\Gamma'_{k+1}) \le R^*(\Gamma_{k+1})$. To satisfy each flow's deadline, Γ'_{k+1} requires a bandwidth R such that[14]

$$\begin{cases} \sum_{i=1}^{n} r_i \le R, \\ \frac{B_{k+1}^{(k+1)}(\Gamma_{k+1}) - \sum_{i\in G'_{\hat{h}}} b_i + l_{k+1}^{(max)}(\Gamma'_{k+1})}{R} \le d_n, \\ \frac{B_{\hat{h}}^{(k+1)}(\Gamma_{k+1}) + l_{\hat{h}}^{(max)}(\Gamma_{k+1})}{R - R_{\hat{h}+1}^{(k+1)}(\Gamma_{k+1}) + \sum_{i\in G'_{\hat{h}}} r_i} \le d_{\hat{h}}^{(min)}(\Gamma_{k+1}) \\ \frac{B_h^{(k+1)}(\Gamma_{k+1}) - \sum_{i\in G'_{\hat{h}}} b_i + l_h^{(max)}(\Gamma'_{k+1})}{R - R_{h+1}^{(k+1)}(\Gamma_{k+1}) + \sum_{i\in G'_{\hat{h}}} r_i} \le d_h^{(min)}(\Gamma_{k+1}),\ \text{when } \hat{h} < h \le k \\ \frac{B_h^{(k+1)}(\Gamma_{k+1}) + l_h^{(max)}(\Gamma_{k+1})}{R - R_{h+1}^{(k+1)}(\Gamma_{k+1})} \le d_h^{(min)}(\Gamma_{k+1}),\ \text{when } h < \hat{h} \end{cases}$$

[14]When $G_{k+1}(\Gamma'_{k+1}) = \emptyset$, $\frac{B_{k+1}^{(k+1)}(\Gamma_{k+1}) - \sum_{i\in G'_{\hat{h}}} b_i + l_{k+1}^{(max)}(\Gamma'_{k+1})}{R} = \frac{\max_i l_i}{R}$, which is also no greater than d_n.

which gives a minimum required bandwidth $R^*(\Gamma')$ of

$$\max\begin{cases}\sum_{i=1}^{n} r_i,\\ \dfrac{B_{k+1}^{(k+1)}(\Gamma_{k+1})-\sum_{i\in G'_{\hat{h}}} b_i + l_{k+1}^{(max)}(\Gamma'_{k+1})}{d_n},\\ \dfrac{B_{\hat{h}}^{(k+1)}(\Gamma_{k+1})+l_{\hat{h}}^{(max)}(\Gamma_{k+1})}{d_{\hat{h}}^{(min)}(\Gamma_{k+1})}+R_{\hat{h}+1}^{(k+1)}(\Gamma_{k+1})-\sum_{i\in G'_{\hat{h}}} r_i\\ \dfrac{B_{h}^{(k+1)}(\Gamma_{k+1})-\sum_{i\in G'_{\hat{h}}} b_i + l_{h}^{(max)}(\Gamma'_{k+1})}{d_h^{(min)}(\Gamma_{k+1})}+R_{h+1}^{(k+1)}(\Gamma_{k+1})-\sum_{i\in G'_{\hat{h}}} r_i, \text{ when } \hat{h}<h\le k\\ \dfrac{B_{h}^{(k+1)}(\Gamma_{k+1})+l_{h}^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})}+R_{h+1}^{(k+1)}(\Gamma_{k+1}), \text{ when } h<\hat{h}\end{cases}$$

Note that if for all $\hat{h}<h\le k+1$, $l_h^{(max)}(\Gamma'_{k+1})-\sum_{i\in G'_{\hat{h}}} b_i \le l_h^{(max)}(\Gamma_{k+1})$, we will then have $R^*(\Gamma'_{k+1})\le R^*(\Gamma_{k+1})$. In fact, as $\max_{i\in G'_{\hat{h}}} l_i - \sum_{i\in G'_{\hat{h}}} b_i \le 0$ we have

$$\begin{aligned} l_h^{(max)}(\Gamma'_{k+1})-\sum_{i\in G'_{\hat{h}}} b_i &= \max\left\{ l_h^{(max)}(\Gamma_{k+1}),\ \max_{i\in G'_{\hat{h}}} l_i \right\} - \sum_{i\in G'_{\hat{h}}} b_i \\ &\le \max\left\{ l_h^{(max)}(\Gamma_{k+1}) - \sum_{i\in G'_{\hat{h}}} b_i, 0 \right\} < l_h^{(max)}(\Gamma_{k+1}). \end{aligned}$$

Thus, we show the existence of a mechanism Γ'_{k+1} satisfying Condition 1.

2) Next we show that for any $(k+1)$-priority mechanism Γ_{k+1} satisfying Condition 1, there exists a $(k+1)$-priority mechanism Γ'_{k+1} satisfying Condition 2.
For Γ_{k+1}, there to exist $1\le\hat{n}\le n$ such that $G_{k+1}(\Gamma_{k+1})=\{i \mid d_n\le d_i<d_{\hat{n}}\ \}$. If $G_{k+1}(\Gamma_{k+1})=\emptyset$, by induction of hypothesis $S(k)$ we have $S(k+1)$. Afterwards we consider the case where $G_{k+1}(\Gamma_{k+1})\neq\emptyset$.
Consider flows $\tilde{F}=\{(r_1,b_1,l_1,d_1),...,(r_{\hat{n}-1},b_{\hat{n}-1},l_{\hat{n}-1},d_{\hat{n}-1}),(r_{\hat{n}},\sum_{i\ge\hat{n}} b_i,l_{\hat{n}},d_{\hat{n}})\}$. According to $S(k)$, there exists a k-priority mechanism Γ'_k for $\tilde{F}$ such that $\forall j<i\le k$, $\tilde{d}_i^{(max)}(\Gamma'_k)<\tilde{d}_j^{(min)}(\Gamma'_k)$. Γ_k gives a minimum required bandwidth of

$$R^*(\Gamma'_k)=\max_{1\le h\le k}\left\{\sum_{i=1}^{\hat{n}} r_i,\ \frac{\tilde{B}_h^{(k)}(\Gamma'_k)+l_h^{(max)}(\Gamma'_k)}{d_h^{(min)}(\Gamma'_k)}+\tilde{R}_{h+1}^{(k)}(\Gamma'_k)\right\}$$

Consider the k-priority mechanism Γ_k, where $G_h(\Gamma_k)=G_h(\Gamma_{k+1})$ for all $1\le h\le k$. Applying Γ_k to $\tilde{F}$, we have

$$\begin{aligned} R^*(\Gamma_k) &= \max_{1\le h\le k}\left\{\sum_{i=1}^{\hat{n}} r_i, \frac{\tilde{B}_h^{(k)}(\Gamma_k)+l_h^{(max)}(\Gamma_k)}{d_h^{(min)}(\Gamma_k)}+\tilde{R}_{h+1}^{(k)}(\Gamma_k)\right\} \\ &= \max_{1\le h\le k}\left\{\sum_{i=1}^{\hat{n}} r_i, \frac{B_h^{(k+1)}(\Gamma_{k+1})+l_h^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})}+R_{h+1}^{(k+1)}(\Gamma_{k+1})-\sum_{i=\hat{n}+1}^{n} r_i\right\}. \end{aligned}$$

From $R^*(\Gamma_k)\ge R^*(\Gamma'_k)$, we know that

$$\begin{aligned} &\max_{1\le h\le k}\left\{\frac{\tilde{B}_h^{(k)}(\Gamma'_k)+l_h^{(max)}(\Gamma'_k)}{d_h^{(min)}(\Gamma'_k)}+\tilde{R}_{h+1}^{(k)}(\Gamma'_k)\right\} \\ &\quad\le \max\left\{\ \sum_{i=1}^{\hat{n}} r_i,\ \max_{1\le h\le k}\left\{\frac{B_h^{(k+1)}(\Gamma_{k+1})+l_h^{(max)}(\Gamma)}{d_h^{(min)}(\Gamma_{k+1})}+R_{h+1}^{(k+1)}(\Gamma_{k+1})-\sum_{i=\hat{n}+1}^{n} r_i\right\}\ \right\} \end{aligned}$$

which further gives

$$\begin{aligned} &\max_{1\le h\le k}\left\{\frac{\tilde{B}_h^{(k)}(\Gamma'_k)+l_h^{(max)}(\Gamma'_k)}{d_h^{(min)}(\Gamma'_k)}+\tilde{R}_{h+1}^{(k)}(\Gamma'_k)+\sum_{i=\hat{n}+1}^{n} r_i\right\} \\ &\quad\le \max\left\{\ \sum_{i=1}^{n} r_i, \max_{1\le h\le k}\left\{\frac{B_h^{(k+1)}(\Gamma_{k+1})+l_h^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})}+R_{h+1}^{(k+1)}(\Gamma_{k+1})\right\}\ \right\} \end{aligned} \tag{21}$$

Now consider the $(k+1)$-priority mechanism Γ'_{k+1}, where $G_h(\Gamma'_{k+1}) = G_h(\Gamma'_k)$ for all $1 \leq h \leq k$, and $G_{k+1}(\Gamma'_{k+1}) = G_{k+1}(\Gamma_{k+1})$. By the definition of $G_{k+1}(\Gamma_{k+1})$ and Γ'_k, we know that $d_i^{(max)}(\Gamma'_{k+1}) < d_j^{(min)}(\Gamma'_{k+1}), \forall j < i \leq k+1$. Next we show that $R^*(\Gamma'_{k+1}) \leq R^*(\Gamma_{k+1})$.
Applying Γ'_{k+1} to flow $F = \{(r_i, b_i, l_i, d_i) \mid 1 \leq i \leq n\}$, we have

$$\begin{aligned}
R^*(\Gamma'_{k+1}) &= \max_{1 \leq h \leq k+1} \left\{ \sum_{i=1}^{n} r_i,\ \frac{B_h^{(k+1)}(\Gamma'_{k+1}) + l_h^{(max)}(\Gamma'_{k+1})}{d_h^{(min)}(\Gamma'_{k+1})} + R_{h+1}^{(k+1)}(\Gamma'_{k+1}) \right\} \\
&= \max_{1 \leq h \leq k} \left\{ \sum_{i=1}^{n} r_i, \frac{B_{k+1}^{(k+1)}(\Gamma_{k+1}) + l_{k+1}^{(max)}(\Gamma_{k+1})}{d_{k+1}^{(min)}(\Gamma_{k+1})}, \frac{B_h^{(k+1)}(\Gamma'_{k+1}) + l_h^{(max)}(\Gamma'_{k+1})}{d_h^{(min)}(\Gamma'_{k+1})} + R_{h+1}^{(k+1)}(\Gamma'_{k+1}) \right\} \\
&= \max_{1 \leq h \leq k} \left\{ \sum_{i=1}^{n} r_i, \frac{B_{k+1}^{(k+1)}(\Gamma_{k+1}) + l_{k+1}^{(max)}(\Gamma_{k+1})}{d_{k+1}^{(min)}(\Gamma_{k+1})}, \frac{\tilde{B}_h^{(k)}(\Gamma'_k) + l_h^{(max)}(\Gamma'_k)}{d_h^{(min)}(\Gamma'_k)} + \tilde{R}_{h+1}^{(k)}(\Gamma'_k) + \sum_{i=\hat{n}+1}^{n} r_i \right\}
\end{aligned}$$

Combining it with Eq. (21), we have

$$\begin{aligned}
R^*(\Gamma'_{k+1}) &\leq \max \begin{cases} \sum_{i=1}^{n} r_i, \\ \frac{B_{k+1}^{(k+1)}(\Gamma_{k+1}) + l_{k+1}^{(max)}(\Gamma_{k+1})}{d_{k+1}^{(min)}(\Gamma_{k+1})}, \\ \max_{1 \leq h \leq k} \left\{ \frac{B_h^{(k+1)}(\Gamma_{k+1}) + l_h^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})} + R_{h+1}^{(k+1)}(\Gamma_{k+1}) \right\} \end{cases} \\
&= \max \left\{ \sum_{i=1}^{n} r_i, \max_{1 \leq h \leq k+1} \left\{ \frac{B_h^{(k+1)}(\Gamma_{k+1}) + l_h^{(max)}(\Gamma_{k+1})}{d_h^{(min)}(\Gamma_{k+1})} + R_{h+1}^{(k+1)}(\Gamma_{k+1}) \right\} \right\} \\
&= R^*(\Gamma_{k+1})
\end{aligned}$$

Hence we show the existence of a mechanism Γ'_{k+1} satisfying Condition 2.

□

B. Proof for Proposition 5

PROPOSITION 5. *Consider a link shared by n token bucket controlled flows, where flow $i, 1 \leq i \leq n$, has traffic profile (r_i, b_i). Assume a static priority scheduler that assigns flow i a priority of i, where priority n is the highest priority, and reprofiles flow i to (r_i, b'_i), where $0 \leq b'_i \leq b_i$. Given a link bandwidth of $R \geq \sum_{j=1}^{n} r_j$, the worst-case delay for flow i is*

$$D_i^* = \max \left\{ \frac{b_i + B'_{i+1}}{R - R_{i+1}},\ \frac{b_i - b'_i}{r_i} + \frac{B'_{i+1}}{R - R_{i+1}} \right\}. \tag{6}$$

Proof. Flow $1 \leq i \leq n$ receives a service curve of $SC_1^{(i)} = \begin{cases} b'_i + r_i t, & \text{when } t > 0 \\ 0 & \text{otherwise} \end{cases}$ inside the reprofiler, and a service curve of $SC_2^{(i)}(t) = \left[(R - R_{i+1})t - B'_{i+1}\right]^+$ at the shared link. Overall it receives a service curve of[15]

$$SC^{(i)}(t) = SC_1^{(i)} \otimes SC_2^{(i)} = \min \left\{ \left[b'_i + r_i \left(t - \frac{B'_{i+1}}{R - R_{i+1}} \right) \right]^+, [(R - R_{i+1})t - B'_{i+1}]^+ \right\}.$$

Hence, we have

$$D_i^* = \sup_{t \geq 0} \inf_{\tau \geq 0} \left\{ b_i + r_i t \leq SC^{(i)}(t + \tau) \right\} = \max \left\{ \frac{b_i + B'_{i+1}}{R - R_{i+1}},\ \frac{b_i - b'_i}{r_i} + \frac{B'_{i+1}}{R - R_{i+1}} \right\}.$$

□

[15] See THEOREM 1.4.6 in [17], page 28

C. *Proofs for Proposition 7 and 8*

This section provides the solution for **OPT_RSP**, from which Propositions 7 and 8 derive. For the reader's convenience, we restate **OPT_RSP**. Remember that we define $B'_i = \sum_{j=i}^{n} b'_j$ and $R_i = \sum_{j=i}^{n} r_j$, where $B'_i, R_i = 0$ when $i > n$.

$$
\begin{aligned}
\textbf{OPT\_RSP} \quad & \min_{\boldsymbol{b'}} R \\
\text{s.t} \quad & \max\left\{\frac{b_i + B'_{i+1}}{R - R_{i+1}},\ \frac{b_i - b'_i}{r_i} + \frac{B'_{i+1}}{R - R_{i+1}}\right\} \le d_i, \qquad \forall\ 1 \le i \le n, \\
& R_1 \le R, \qquad b'_1 \le b_1, \qquad 0 \le b'_i \le b_i, \qquad \forall\ 2 \le i \le n.
\end{aligned}
$$

Instead of solving **OPT_RSP**, for technical simplicity we consider **OPT_RSP'**, whose solution directly gives that for **OPT_RSP**. Next we first demonstrate the relationship between **OPT_RSP** and **OPT_RSP'** (Lemma 13), and then proceed to solve **OPT_RSP'** (Lemma 14). Combining Lemma 13 and 14, we then have Proposition 7 and 8.

Lemma 13. *For $1 \le i \le n$, define $H_i = b_i - r_i d_i$, and $\boldsymbol{B'} = (B'_1, ..., B'_n)$. Consider the following optimization:*

$$
\begin{aligned}
\textbf{OPT\_RSP'} \quad & \min_{\boldsymbol{B'}} R \\
\textit{s.t} \quad & R_1 \le R. \\
& B'_2 \le d_1(R - R_2) - b_1, \\
& B'_i \in \left[\max\left\{\frac{R - R_{i+1} + r_i}{R - R_{i+1}} B'_{i+1} + H_i,\ B'_{i+1}\right\},\ B'_{i+1} + \frac{b_i(R - R_i)}{R - R_{i+1}}\right], \quad \forall\ 2 \le i \le n.
\end{aligned}
\tag{22}
$$

*Suppose the optimal solution for **OPT_RSP'** is $(R^*, \boldsymbol{B'^*})$. Then $(R^*, \boldsymbol{b'^*})$, where*

$$
\boldsymbol{b'^*} = (b_1, B'^*_2 - B'^*_3, ..., B'^*_{n-1} - B'^*_n, B'^*_n),
$$

*is an optimal solution for **OPT_RSP**.*

Proof. We first show that $\boldsymbol{b'^*} = (b_1, B'^*_2 - B'^*_3, ..., B'^*_{n-1} - B'^*_n, B'^*_n)$ and R^* satisfy all the constraints for **OPT_RSP**, and then show that $(R^*, \boldsymbol{b'^*})$ is an optimal solution for **OPT_RSP**.

Substituting $b'_1 = b_1$ into $\max\left\{\frac{b_1+B'_2}{R-R_2},\ \frac{b_1-b'_1}{r_1} + \frac{B'_2}{R-R_2}\right\} \le d_1$ gives $\frac{b_1+B'_2}{R-R_2} \le d_1$, which is equivalent to $B'_2 \le d_1(R-R_2) - b_1$. Thus, to show the feasibility of $(R^*, \boldsymbol{b'^*})$, we only need to show that $(R^*, \boldsymbol{b'^*})$ satisfies $\max\left\{\frac{b_i+B'^*_{i+1}}{R^*-R_{i+1}},\ \frac{b_i-b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R^*-R_{i+1}}\right\} \le d_i$ and $b'^*_i \in [0, b_i]$ for all $2 \le i \le n$. Below we consider each constraint separately.

- Basic algebraic manipulation gives that

$$
\max\left\{\frac{b_i + B'^*_{i+1}}{R^* - R_{i+1}},\ \frac{b_i - b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R^* - R_{i+1}}\right\} = \begin{cases} \dfrac{b_i + B'^*_{i+1}}{R^* - R_{i+1}}, & \text{when } b'^*_i \ge \dfrac{b_i(R^* - R_i)}{R^* - R_{i+1}}, \\ \dfrac{b_i - b'^*_i}{r_i} + \dfrac{B'^*_{i+1}}{R^* - R_{i+1}}, & \text{otherwise.} \end{cases} \tag{23}
$$

  From **OPT_RSP'** we have $B'^*_i \le B'^*_{i+1} + \frac{b_i(R^*-R_i)}{R^*-R_{i+1}}$, *i.e.*, $b'^*_i = B'^*_i - B'^*_{i+1} \le \frac{b_i(R^*-R_i)}{R^*-R_{i+1}}$, and therefore $\max\{\frac{b_i+B'^*_{i+1}}{R^*-R_{i+1}},\ \frac{b_i-b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R^*-R}$ $\frac{b_i-b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R^*-R_{i+1}}$. From **OPT_RSP'** we also have $B'^*_i \ge \frac{R^*-R_{i+1}+r_i}{R^*-R_{i+1}} B'^*_{i+1} + H_i$, *i.e.*, $B'^*_i - B'^*_{i+1} - H_i = b'^*_i - b_i + r_i d_i \ge \frac{r_i B'^*_{i+1}}{R-R_{i+1}}$, which is equivalent to $\frac{b_i-b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R-R_{i+1}} \le d_i$. Combining them gives $\max\left\{\frac{b_i+B'^*_{i+1}}{R^*-R_{i+1}},\ \frac{b_i-b'^*_i}{r_i} + \frac{B'^*_{i+1}}{R^*-R_{i+1}}\right\} \le d_i$.
- From **OPT_RSP'** we have $B'^*_i \ge B'^*_{i+1}$, and therefore $b'^*_i = B'^*_i - B'^*_{i+1} \ge 0$. From **OPT_RSP'** we also have $B'^*_i \le B'^*_{i+1} + \frac{b_i(R^*-R_i)}{R^*-R_{i+1}}$, and therefore $b'^*_i \le \frac{b_i(R^*-R_i)}{R^*-R_{i+1}} < b_i$. Thus, we have $b'^*_i \in [0, b_i]$.

Next we show by contradiction that $(R^*, \boldsymbol{b'^*})$ is optimal for **OPT_RSP**. Suppose $(\tilde{R}, \tilde{\boldsymbol{b}}')$, where $\tilde{R} < R^*$ is an optimal solution for **OPT_RSP**. Denote $\tilde{B}'_i = \sum_{j=i}^{n} \tilde{b}'_j$, and $\tilde{\boldsymbol{B}}' = (\tilde{B}'_1, ..., \tilde{B}'_n)$.

- If $(\tilde{R}, \tilde{\boldsymbol{B}}')$ satisfies all constraints for **OPT_RSP'**, then by $(R^*, \boldsymbol{b'^*})$'s optimality we know that $\tilde{R} \ge R^*$, which contradicts to the assumption that $\tilde{R} < R^*$.
- Otherwise, there exists $2 \le i \le n$ such that $\max\left\{\frac{b_i+\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}},\ \frac{b_i-\tilde{b}'_i}{r_i} + \frac{\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}}\right\} \le d_i$ and $\tilde{b}'_i \in [0, b_i]$, whereas $\tilde{B}'_i \notin \left[\max\left\{\frac{\tilde{R}-R_{i+1}+r_i}{\tilde{R}-R_{i+1}} \tilde{B}'_{i+1} + H_i,\ \tilde{B}'_{i+1}\right\},\ \tilde{B}'_{i+1} + \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}\right]$.
  - When $\tilde{b}'_i \le \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}$, from Eq. (23) we have that $\max\left\{\frac{b_i+\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}},\ \frac{b_i-\tilde{b}'_i}{r_i} + \frac{\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}}\right\} = \frac{b_i-\tilde{b}'_i}{r_i} + \frac{\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}}$. Combining $\frac{b_i-\tilde{b}'_i}{r_i} + \frac{\tilde{B}'_{i+1}}{\tilde{R}-R_{i+1}} \le d_i$ and $\tilde{b}'_i \in [0, \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}]$ with $\tilde{b}'_i = \tilde{B}'_i - \tilde{B}'_{i-1}$, we have

$$
\tilde{B}'_i \in \left[\max\left\{\frac{\tilde{R} - R_{i+1} + r_i}{\tilde{R} - R_{i+1}} \tilde{B}'_{i+1} + H_i,\ \tilde{B}'_{i+1}\right\},\ \tilde{B}'_{i+1} + \frac{b_i(\tilde{R} - R_i)}{\tilde{R} - R_{i+1}}\right],
$$

and therefore a contradiction.

– When $\tilde{b}'_i > \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}$, we show that there exists an optimal solution $(\tilde{R}, \hat{\boldsymbol{b}}')$ for **OPT_RSP** such that $\hat{b}'_i \leq \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}$. Define $\hat{\boldsymbol{b}}'$ as 1) $\hat{b}'_i = \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}$, and 2) $\hat{b}'_h = \tilde{b}'_h$ when $h \neq i$. Denote $\hat{B}'_h = \sum_{j=h}^n \hat{b}'_j \leq \tilde{B}'_h$, and $\hat{\boldsymbol{B}}' = (\hat{B}'_1, ..., \hat{B}'_n)$. Next we show that $(\tilde{R}, \hat{\boldsymbol{b}}')$ satisfies all the constraints in **OPT_RSP**, and therefore is optimal. Observe first that from $\hat{B}'_2 < \tilde{B}'_2$, we have $\frac{b_1+\hat{B}'_2}{\tilde{R}-R_2} < \frac{b_1+\tilde{B}'_2}{\tilde{R}-R_2} \leq d_1$. Below we show that for all $2 \leq h \leq n$, $\max\left\{\frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\hat{b}'_h}{r_h} + \frac{\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} \leq d_h$.

  * When $h > i$, by definition $\hat{B}'_{h+1} = \tilde{B}'_{h+1}$. Thus we have

$$\max\left\{\frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\hat{b}'_h}{r_h} + \frac{\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} = \max\left\{\frac{b_h+\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\tilde{b}'_h}{r_h} + \frac{\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} \leq d_h.$$

  * When $h = i$, $\max\left\{\frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\hat{b}'_h}{r_h} + \frac{\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} = \frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}}$. Combining it with $\hat{B}'_{h+1} < \tilde{B}'_{h+1}$ and $\max\left\{\frac{b_h+\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\tilde{b}'_h}{r_h} + \frac{\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} \leq d_h$ gives the result.
  * When $h < i$, from $\hat{B}'_{h+1} < \tilde{B}'_{h+1}$ we have $\frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}} < \frac{b_h+\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}}$ and $\frac{b_h-\tilde{b}'_h}{r_h} + \frac{\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}} < \frac{b_h-\tilde{b}'_h}{r_h} + \frac{\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}}$, *i.e.*, $\max\left\{\frac{b_h+\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\hat{b}'_h}{r_h} + \frac{\hat{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} < \max\left\{\frac{b_h+\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}},\ \frac{b_h-\tilde{b}'_h}{r_h} + \frac{\tilde{B}'_{h+1}}{\tilde{R}-R_{h+1}}\right\} \leq d_h$.

  If $(\tilde{R}, \hat{\boldsymbol{b}}')$ satisfies all the constraints for **OPT_RSP**, it contradicts to the assumption that $\tilde{R} < R^*$. Otherwise, from the case for $\tilde{b}'_i \leq \frac{b_i(\tilde{R}-R_i)}{\tilde{R}-R_{i+1}}$, we know it again to produce a contradiction.

□

Next, we proceed to solve **OPT_RSP'**, which when combine with Lemma 13 then gives Propositions 7 and 8. For the reader's convenience, we restate the Propositions.

PROPOSITION 7. *For* $1 \leq i \leq n$, *denote* $H_i = b_i - d_i r_i$, $\Pi_i(R) = \frac{r_i+R-R_{i+1}}{R-R_{i+1}}$ *and* $V_i(R) = d_i(R - R_{i+1}) - b_i$. *Define* $S_1(R) = \{V_1(R)\}$, *and* $S_i(R) = S_{i-1}(R) \bigcup \{V_i(R)\} \bigcup \left\{\frac{s-H_i}{\Pi_i(R)} \mid s \in S_{i-1}(R)\right\}$ *for* $2 \leq i \leq n$. *Then we have* $\tilde{R}^*_R = \max\{R_1, \inf\{R \mid \forall s \in S_n(R), s \geq 0\}\}$.

PROPOSITION 8. *The optimal reprofiling solution* $\boldsymbol{b}'^*$ *satisfies*

$$b'^*_i = \begin{cases} \max\{0, b_n - r_n d_n\}, & i = n; \\ \max\left\{0,\ b_i - r_i d_i + \dfrac{r_i B'^*_{i+1}}{\tilde{R}^*_R - R_{i+1}}\right\}, & 2 \leq i \leq n-1. \end{cases} \tag{7}$$

Denote $\mathbb{B}_i := \left[\max\left\{\frac{R-R_{i+1}+r_i}{R-R_{i+1}} B'_{i+1} + H_i,\ B'_{i+1}\right\},\ B'_{i+1} + \frac{b_i(R-R_i)}{R-R_{i+1}}\right]$, where the interval overlaps with that in the third constraint of **OPT_RSP'**. We then show that the system meets each flow's deadline only if the shared link has a bandwidth no less than $\max\{R_1, \inf\{R \mid \forall s \in S_n(R), s \geq 0\}$, which in turn gives the minimum required bandwidth R^*. As mentioned before, we achieve this by first solving **OPT_RSP'**, from which we then get the solution for **OPT_RSP** based on Lemma 13.

Lemma 14. *Define* $s_1^{(i)} = \max\{\Pi_i(R) B'_{i+1} + H_i,\ B'_{i+1}\}$, $s_2^{(i)} = \min\left\{S_{i-1}(R), B'_{i+1} + \frac{b_i(R-R_i)}{R-R_{i+1}}\right\}$, *and* $\mathbb{S}_i = \left[s_1^{(i)}, s_2^{(i)}\right]$ *for* $2 \leq i \leq n$, *where* $\Pi_i(R) = \frac{r_i+R-R_{i+1}}{R-R_{i+1}}$, $H_i = b_i - d_i r_i$, $S_1(R) = \{V_1(R)\}$, $V_i(R) = d_i(R - R_{i+1}) - b_i$, *and* $S_i(R) = S_{i-1}(R) \bigcup \{V_i(R)\} \bigcup \left\{\frac{s-H_i}{\Pi_i(R)} \mid s \in S_{i-1}(R)\right\}$. *Then* R *and* $\boldsymbol{b}'$ *satisfies all the constraints in* ***OPT_RSP'*** *iff* $\mathbb{S}_n \neq \emptyset$ *and* $R \geq R_1$, i.e., $R \geq \max\{R_1, \inf\{R \mid \forall s \in S_n(R), s \geq 0\}\}$.

Proof. We rely on the following statements to show the Lemma:

Statement 1. $\left\{(R, \boldsymbol{b}') \mid B'_{i-1} \in \mathbb{S}_{i-1}\right\} \bigcap \{(R, \boldsymbol{b}') \mid B'_i \in \mathbb{B}_i\} = \{(R, \boldsymbol{b}') \mid B'_i \in \mathbb{S}_i\}$.

Given Statement 1, we then show that

Statement 2. $\{(R, \boldsymbol{b}') \mid B'_2 \leq V_1(R), B'_i \in \mathbb{B}_i, \forall\ 2 \leq i \leq n\} = \{(R, \boldsymbol{b}') \mid B'_n \in \mathbb{S}_n\}$.

Note that $B'_2 \leq V_1(R)$ corresponds to the second constraint in **OPT_RSP'**, while $B'_i \in \mathbb{B}_i, \forall\ 2 \leq i \leq n$ corresponds to its third constraint. Therefore, from Statement 2 we have that R and $\boldsymbol{b}'$ satisfies all the constraints in **OPT_RSP'** iff $\mathbb{S}_n \neq \emptyset$ and $R \geq R_1$. Note that $\mathbb{S}_n \neq \emptyset$ iff $s_1^{(n)} \leq s_2^{(n)}$, *i.e.*, $\max\{H_n, 0\} \leq \min\left\{S_{n-1}(R), \frac{b_n(R-R_n)}{R}\right\}$. As $(R - R_n) \geq 0$, $\max\{H_n, 0\} \leq \frac{b_n(R-R_n)}{R}$ iff $V_n(R) \geq 0$, whereas $\max\{H_n, 0\} \leq \min\{S_{n-1}(R)\}$ iff $s \geq \max\{0, H_n\}, \forall s \in S_{n-1}(R)$. Since $\Pi_n(R) > 0$, by the definition of S_n we have $\mathbb{S}_n \neq \emptyset$ and $R \geq R_1$ iff $R \geq \max\{R_1, \inf\{R \mid \forall s \in S_n(R), s \geq 0\}\}$.

a) Proof for Statement 1.: Note that $\left\{(R,\boldsymbol{b}') \mid B'_{i-1}\in\mathbb{S}_{i-1}\right\}=\{(R,\boldsymbol{b}') \mid \mathbb{S}_{i-1}\neq\emptyset\}$. Below we prove that $\mathbb{S}_{i-1}\neq\emptyset$ iff $B'_i\le\min\{S_{i-1}(R)\}$. As $s_2^{(i)}=\min\left\{S_{i-1}(R),B'_{i+1}+\frac{b_i(R-R_i)}{R-R_{i+1}}\right\}$, combining $B'_i\le\min\{S_{i-1}(R)\}$ with $B'_i\in\mathbb{B}_i=\left[s_i^{(1)},\ B'_{i+1}+\frac{b_i(R-R_i)}{R-R_{i+1}}\right]$ directly gives $B'_i\in\mathbb{S}_i=\left[s_1^{(i)},s_2^{(i)}\right]$.

From $\mathbb{S}_{i-1}\neq\emptyset\Leftrightarrow s_2^{(i-1)}\ge s_1^{(i-1)}$, we have

$$\begin{cases} B'_i\le s_2^{(i-1)}=\min\left\{S_{i-2}(R),B'_i+\dfrac{b_{i-1}(R-R_{i-1})}{R-R_i}\right\}\Leftrightarrow B'_i\le\min\left\{S_{i-2}(R)\right\} \\ \Pi_{i-1}(R)B'_i+H_{i-1}\le s_2^{(i-1)}\Leftrightarrow B'_i\le\min\{V_{i-1}(R)\}\bigcup\left\{\dfrac{s-H_{i-1}}{\Pi_{i-1}(R)}\mid s\in S_{i-2}(R)\right\} \end{cases}$$

As $S_{i-1}(R)=S_{i-2}(R)\bigcup\{V_{i-1}(R)\}\bigcup\left\{\frac{s-H_{i-1}}{\Pi_{i-1}(R)}\mid s\in S_{i-2}(R)\right\}$, we have $\mathbb{S}_{i-1}\neq\emptyset$ iff $B'_i\le\min\{S_{i-1}(R)\}$.

b) Proof for Statement 2.: We show by induction on the value of n.

- Base case: when $n=2$, basic algebraic manipulation gives that

$$\begin{aligned}\{(R,\boldsymbol{b}')\mid B'_2\le V_1(R),B'_2\in\mathbb{B}_i\}&=\left\{(R,\boldsymbol{b}')\mid B'_2\in\left[s_1^{(2)},\min\left\{V_1(R),B'_3+\frac{b_2(R-R_2)}{R-R_3}\right\}\right]\right\}\\&=\{(R,\boldsymbol{b}')\mid B'_2\in\mathbb{S}_2\}\end{aligned}$$

- Induction step: suppose $\{(R,\boldsymbol{b}')\mid B'_2\le V_1(R),B'_i\in\mathbb{B}_i,\forall\ 2\le i\le k\}=\{(R,\boldsymbol{b}')\mid B'_k\in\mathbb{S}_k\}$. Then we have

$$\begin{aligned}&\{(R,\boldsymbol{b}')\mid B'_2\le V_1(R),B'_i\in\mathbb{B}_i,\forall\ 2\le i\le k+1\}\\=&\{(R,\boldsymbol{b}')\mid B'_2\le V_1(R),B'_i\in\mathbb{B}_i,\forall\ 2\le i\le k\}\bigcap\left\{(R,\boldsymbol{b}')\mid B'_{k+1}\in\mathbb{B}_{k+1}\right\}\\=&\{(R,\boldsymbol{b}')\mid B'_k\in\mathbb{S}_k\}\bigcap\left\{(R,\boldsymbol{b}')\mid B'_{k+1}\in\mathbb{B}_{k+1}\right\}\\=&\left\{(R,\boldsymbol{b}')\mid B'_{k+1}\in\mathbb{S}_{k+1}\right\}\end{aligned}$$

  where the last equation comes from Statement 1.
- Conclusion: by the principle of induction, we have

$$\{(R,\boldsymbol{b}')\mid B'_2\le V_1(R),B'_i\in\mathbb{B}_i,\forall\ 2\le i\le n\}=\{(R,\boldsymbol{b}')\mid B'_n\in\mathbb{S}_n\}.$$

□

Observe from the proof of Lemma 14 that for all $2\le i\le n$, setting $B'_i=\max\left\{\Pi_i(R)B'_{i+1}+H_i,\ B'_{i+1}\right\}$ gives $\tilde{R}^*_R$. Combining it with Lemma 13, we know that R^* is the optimal solution for **OPT_RSP**, and therefore we have Proposition 7. Besides, since $\tilde{R}^*_R$ can be achieved by setting $b'_i=\max\left\{\frac{r_iB'_{i+1}}{R-R_{i+1}}+H_i,\ 0\right\}$, we then have Proposition 8.

Appendix D
Proofs for FIFO Scheduler

A. Proof for Proposition 9

Proposition 9. *Consider a system with n token bucket controlled flows with traffic profiles $(r_i,b_i),1\le i\le n$, sharing a fifo link with bandwidth $R\ge R_1=\sum_{j=1}^n r_j$. Assume that the system reprofiles flow i to (r_i,b'_i). The worst-case delay for flow i is then*

$$\widehat{D}^*_i=\max\left\{\frac{b_i-b'_i}{r_i}+\frac{\sum_{j\neq i}b'_j}{R},\frac{\sum_{j=1}^n b'_j}{R}+\frac{(b_i-b'_i)R_1}{r_iR}\right\}. \tag{8}$$

Proof. W.l.o.g we consider the worst-case delay for flow 1. First we show that there always exists a traffic pattern such that flow 1's worst-case delay can be achieved by the last bit inside a burst of size b_1, and then we characterize the worst-case delay for that b_1^{th} bit.

- Consider the traffic pattern $\boldsymbol{T}(t)=\{T_1(t),...,T_n(t)\}$ that realizes flow 1's worst-case delay, where $T_i(t)$ is right continuous and specifies the cumulative amount of data sent by flow i during time $[0,t]$. Suppose the worst-case delay is achieved at t_0, and at t_0 flow 1 sends a burst of $b\le b_1$. As under FIFO the last bit gets a strictly larger delay compared with all the other bits inside the burst, the b^{th} bit sent at t_0 achieves the worst-case delay.

  If $b=b_1$, flow 1's worst-case delay is achieved by the b_1^{th} bit inside a burst, *i.e.,* $\boldsymbol{T}(t)$ is the traffic pattern we want. Afterwards we consider the case $b<b_1$.

  First note that b is the maximum amount of data flow 1 can send at t_0 without violating its arrival-curve constraint, *i.e.,* there exists $t_s\in[0,t_0)$ such that $T(t_0)-T(t_s)=b_1+r_1(t_0-t_s)$. Otherwise, we can produce a worse delay by increasing b to the maximum value that remains conformant with the arrival-curve constraint. Next we show that if $b<b_1$, there

exists $T_1'(t)$ sending a burst of b_1 at t_0, such that under $\boldsymbol{T}'(t) = \{T_1'(t), ..., T_n(t)\}$ the last bit flow 1 sends at t_0 also achieves flow 1's worst-case delay.
Define $\hat{t} = \sup\{t \mid T_1(t_0) - T(t) \geq b_1\}$. As $b < b_1$, $\hat{t} \in (t_s, t_0)$. Note that $T_1(t_0) - T_1(\hat{t}) \leq b_1$ as $T_1(t)$ is right continuous.
Define

$$T_1'(t) = \begin{cases} T_1(t), & \text{when } t < \hat{t} \\ T_1(t_0) - b_1, & \text{when } \hat{t} \leq t < t_0 \\ T_1(t_0), & \text{otherwise} \end{cases}$$

which sends a burst of b_1 at t_0. Note that $T_1'(t)$ satisfies flow 1's arrival curve. Specifically, consider any $0 \leq t^{(1)} < t^{(2)}$.

- When $t^{(2)} < \hat{t}$, it has $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1\left(t^{(2)}\right) - T_1\left(t^{(1)}\right)$.
- When $\hat{t} \leq t^{(2)} < t_0$,
  * If $t^{(1)} < \hat{t}$, $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1(t_0) - b_1 - T_1\left(t^{(1)}\right) \leq T_1(\hat{t}) - T_1\left(t^{(1)}\right) \leq T_1\left(t^{(2)}\right) - T_1\left(t^{(1)}\right)$
  * If $\hat{t} \leq t^{(1)} < t_0$, $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1(t_0) - b_1 - [T_1(t_0) - b_1] = 0$
- When $t^{(2)} \geq t_0$,
  * If $t^{(1)} \leq \hat{t}$, $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1(t_0) - T_1\left(t^{(1)}\right) \leq T_1\left(t^{(2)}\right) - T_1\left(t^{(1)}\right)$
  * If $\hat{t} < t^{(1)} < t_0$, $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1(t_0) - [T_1(t_0) - b_1] = b_1 \leq b_1 + r_1\left(t^{(2)} - t^{(1)}\right)$
  * If $t^{(1)} \geq t_0$, $T_1'\left(t^{(2)}\right) - T_1'\left(t^{(1)}\right) = T_1(t_0) - T_1(t_0) = 0$

We then show that under $\boldsymbol{T}'(t)$ the last bit sent at t_0 also achieves flow 1's worst-case delay. First observe that under $\boldsymbol{T}'(t)$ the last bit sent at t_0 arrives at the shared link no earlier than that under $\boldsymbol{T}(t)$. This is because it arrives at the reprofiler later than under $T(t)$ and experiences a no smaller reshaping delay upon its arrival. Particularly, given the reprofiler's service curve of $\begin{cases} b_1' + r_1 t, & \text{when } t > 0 \\ 0 & \text{otherwise} \end{cases}$, the b_1^{th} bit of a burst gets a delay of $\frac{b_1 - b_1'}{r_1}$, which equals the worst-case delay for flow 1 inside the reprofiler. Next we show that under $\boldsymbol{T}'(t)$ the last bit leaves the shared link no earlier than that under $\boldsymbol{T}(t)$, *i.e.,* overall it gets a no smaller delay under $\boldsymbol{T}'(t)$. Since under $\boldsymbol{T}(t)$ the last bit achieves the worst-case delay, so does under $\boldsymbol{T}(t)$.
Suppose the last bit arrives at the shared link at $\hat{t}_0$ under $\boldsymbol{T}(t)$, and at $\hat{t}_0' \geq \hat{t}_0$ under $\boldsymbol{T}'(t)$. If under $\boldsymbol{T}(t)$ the last bit arrives to find the shared link with an empty queue, *i.e.,* it has no delay at the shared link, then combining it with $\hat{t}_0' \geq \hat{t}_0$ gives what we want. Afterwards, we consider the case where under $\boldsymbol{T}(t)$ the last bit arrives at the shared link with a non-empty queue, *i.e.,* there exists $\hat{t}_s < \hat{t}_0$ and $\delta > 0$ such that the shared link processes data at full speed R during $[\hat{t}_s, \hat{t}_0]$, and at a speed strictly less than R during $[\hat{t}_s - \delta, \hat{t}_s)$.
Under $\boldsymbol{T}(t)$, $T_1(\hat{t}_s, \hat{t}_0)$ data from flow 1 arrives at the shared link during $[\hat{t}_s, \hat{t}_0]$. Then the last bit leaves the shared link at time $\hat{t}_e = \hat{t}_s + \frac{\sum_{j \geq 2}[T_j(\hat{t}_0) - T_j(\hat{t}_s)] + T_1(\hat{t}_s, \hat{t}_0)}{R}$. Whereas under $\boldsymbol{T}'(t)$, suppose there is $M \geq 0$ amount of data in the buffer at $\hat{t}_s$, and $T_1'(\hat{t}_s, \hat{t}_0')$ amount of data from flow 1 arrives at the shared link during $[\hat{t}_s, \hat{t}_0']$. As $T_1'(t)$ delays some data to t_0, and by $\hat{t}_0'$ all of the delayed data arrives at the shared link, it has $T_1'(\hat{t}_s, \hat{t}_0') \geq T_1(\hat{t}_s, \hat{t}_0)$. Then the b_1^{th} bit leaves the shared link at a time no less than $\hat{t}_s + \frac{M + \sum_{j \geq 2}[T_j(\hat{t}_0') - T_j(\hat{t}_s)] + T_1'(\hat{t}_s, \hat{t}_0')}{R}$, which is no less than $\hat{t}_e$.

- Next we characterize the worst-case delay. Given that there always exists a traffic pattern that the b_1^{th} bit of a burst gives flow 1's worst-case delay, w.l.o.g we assume the worst-case delay to be achieved by the b_1^{th} bit sent at t_0.
Remember that the b_1^{th} bit of a burst gets a delay of $\frac{b_1 - b_1'}{r_1}$ inside the reprofiler. Next we consider the delay at the shared link. Suppose the b_1^{th} bit arrives at the shared link at $\hat{t}_0$. If at $\hat{t}_0$ the shared link processes at a speed strictly less than R, then the b_1^{th} bit gets no delay at the shared link, and therefore gets a overall worst-case delay of $\frac{b_1 - b_1'}{r_1}$. Otherwise, suppose the last busy period at the shared link starts at $0 \leq \hat{t}_s \leq \hat{t}_0$.

1) When $\hat{t}_0 - \hat{t}_s \geq \frac{b_1 - b_1'}{r_1}$, during $[\hat{t}_s, \hat{t}_0]$ at most $\sum_{j=1}^n b_j' + \sum_{j=1}^n r_j\left(\hat{t}_0 - \hat{t}_s\right)$ amount of data arrives at the shared link. Thus, the delay for the b_1^{th} bit at the shared link is $\frac{\sum_{j=1}^n b_j' + \sum_{j=1}^n r_j\left(\hat{t}_0 - \hat{t}_s\right)}{R} - \hat{t}_0$, which decreases with $\hat{t}_0$ since $R \geq \sum_{j=1}^n r_j$. Given $\hat{t}_0 \geq \frac{b_1 - b_1'}{r_1} + \hat{t}_s$, the worst-case delay at the shared link is achieved at $\hat{t}_0 = \frac{b_1 - b_1'}{r_1} + \hat{t}_s$, and has a value of $\frac{\sum_{j=1}^n b_j' + \frac{b_1 - b_1'}{r_1}\sum_{j=1}^n r_j}{R} - \frac{b_1 - b_1'}{r_1} - \hat{t}_s$. Thus the overall worst-case delay is achieved at $\hat{t}_s = 0$, with a value of
$$\frac{\sum_{j=1}^n b_j' + \frac{b_1 - b_1'}{r_1}\sum_{j=1}^n r_j}{R} - \frac{b_1 - b_1'}{r_1} + \frac{b_1 - b_1'}{r_1} = \frac{\sum_{j=1}^n b_j'}{R} + \frac{(b_1 - b_1')R_1}{r_1 R}.$$
2) When $\hat{t}_0 - \hat{t}_s < \frac{b_1 - b_1'}{r_1}$, as flow 1's burst of b_1' arrived at the shared link before $\hat{t}_0 - \frac{b_1 - b_1'}{r_1}$, it should has been cleared before $\hat{t}_s$. Hence, during $[\hat{t}_s, \hat{t}_0]$ at most $\sum_{j \neq 1}^n b_j' + \sum_{j=1}^n r_j\left(\hat{t}_0 - \hat{t}_s\right)$ data arrive at the shared link. Therefore, the b_1^{th} bit gets a delay of $\frac{\sum_{j \neq 1}^n b_j' + \sum_{j=1}^n r_j\left(\hat{t}_0 - \hat{t}_s\right)}{R} - \hat{t}_0$ at the shared link, which decreases with $\hat{t}_0$ under $R \geq \sum_{j=1}^n r_j$. Consequently, its worst-case delay is achieved at $\hat{t}_0 = \hat{t}_s$, with a value of $\frac{\sum_{j \neq 1}^n b_j'}{R} - \hat{t}_s$, which is maximized at $\hat{t}_s = 0$. Thus the overall worst-case delay is $\frac{b_1 - b_1'}{r_1} + \frac{\sum_{j \neq 1}^n b_j'}{R}$.

Combining case 1 and 2 gives flow 1's worst-case delay, *i.e.,*

$$\widehat{D}_1^* = \max\left\{\frac{b_1 - b_1'}{r_1} + \frac{\sum_{j\neq 1} b_j'}{R}, \frac{\sum_{j=1}^n b_j'}{R} + \frac{(b_1 - b_1')R_1}{r_1 R}\right\}.$$

□

B. Proofs for Proposition 10 and 11

This section provides the solution for **OPT_RF**, from which we then have Propositions 10 and 11. For the reader's convenience, we restate **OPT_RF**, where $R_1 = \sum_{i=1}^n r_i$.

$$\begin{aligned}
&\textbf{OPT_RF} \quad \min_{\boldsymbol{b'}} R \\
&\text{s.t} \quad \max\left\{\frac{b_i - b_i'}{r_i} + \frac{\sum_{j\neq i} b_j'}{R}, \frac{\sum_{j=1}^n b_j'}{R} + \frac{(b_i - b_i')R_1}{r_i R}\right\} \leq d_i, \quad \forall\ 1 \leq i \leq n, \\
&\qquad R_1 \leq R, \quad 0 \leq b_i' \leq b_i, \qquad \forall\ 1 \leq i \leq n.
\end{aligned}$$

Lemma 15. *For $1 \leq i \leq n$ define $T_i^{(1)} = \frac{R}{R+r_i}\left(H_i + \frac{r_i}{R}\widehat{B}_n'\right)$ and $T_i^{(2)} = b_i + \frac{r_i(\widehat{B}_n' - Rd_i)}{R_1}$. Denote $\widehat{B}_0' = 0$ and $\widehat{\boldsymbol{B}}' = (\widehat{B}_1', ...\widehat{B}_n')$. Consider the following optimization:*

$$\begin{aligned}
&\textbf{\textit{OPT_RF'}} \quad \min_{\widehat{\boldsymbol{B}}'} R \\
&\textit{s.t} \quad \max\left\{0, T_i^{(1)}, T_i^{(2)}\right\} \leq \widehat{B}_i' - \widehat{B}_{i-1}' \leq b_i, \quad \forall\ 1 \leq i \leq n, \\
&\qquad R_1 \leq R.
\end{aligned}$$

*Suppose the optimal solution for **OPT_RF'** is $(R^*, \widehat{\boldsymbol{B}}'^*)$. Then $(R^*, \boldsymbol{b}'^*)$, where*

$$\boldsymbol{b}'^* = (\widehat{B}_1'^*, \widehat{B}_2'^* - \widehat{B}_1'^*, ..., \widehat{B}_n'^* - \widehat{B}_{n-1}'^*)$$

*is an optimal solution for **OPT_RF**.*

Proof. Define $\widehat{B}_i = \sum_{j=1}^i b_j$. From basic algebraic manipulation **OPT_RF** is equivalent to **OPT_RF"**:

$$\begin{aligned}
&\textbf{OPT_RF"} \quad \min_{\boldsymbol{b'}} R \\
&\text{s.t} \quad \max\left\{0, T_i^{(1)}, T_i^{(2)}\right\} \leq \widehat{B}_i' - \widehat{B}_{i-1}' \leq b_i \quad \forall\ 1 \leq i \leq n, \\
&\qquad R_1 \leq R.
\end{aligned}$$

As the constraints of **OPT_RF'** and **OPT_RF"** are the same, the two optimizations share the same optimal value R^*. From $\widehat{B}_i = \sum_{j=1}^i b_j$, $\boldsymbol{b}'^* = (\widehat{B}_1'^*, \widehat{B}_2'^* - \widehat{B}_1'^*, ..., \widehat{B}_n'^* - \widehat{B}_{n-1}'^*)$ is then the optimal variable for **OPT_RF"**. Hence, $(R^*, \boldsymbol{b}'^*)$ is an optimal solution for **OPT_RF"**, and therefore an optimal solution for **OPT_RF**. □

Next we proceed to solve **OPT_RF'**, which when combined with Lemma 15 then gives Propositions 10 and 11. Define $\mathbb{Z}_i = \{1 \leq j \leq i \mid j \in \mathbb{Z}\}$ for $1 \leq i \leq n$,

$$X_F(R) = \max_{P_1, P_2 \subseteq \mathbb{Z}_n, P_2 \neq \mathbb{Z}_n, P_1 \bigcap P_2 = \emptyset} \frac{\sum_{i\in P_1} \frac{RH_i}{R+r_i} + \sum_{i\in P_2}\left(b_i - \frac{r_i d_i R}{R_1}\right)}{1 - \sum_{i\in P_1}\frac{r_i}{R+r_i} - \sum_{i\in P_2}\frac{r_i}{R_1}},$$

and

$$Y_F(R) = \min_{1\leq i\leq n-1}\left\{\widehat{B}_n, Rd_n, \min_{P_1, P_2 \subseteq \mathbb{Z}_i, P_1 \bigcap P_2 = \emptyset, P_1 \bigcup P_2 \neq \emptyset}\left\{\frac{\widehat{B}_i - \sum_{j\in P_1}\frac{RH_j}{R+r_j} - \sum_{j\in P_2}\left(b_j - \frac{r_j d_j R}{R_1}\right)}{\sum_{j\in P_1}\frac{r_j}{R+r_j} + \sum_{j\in P_2}\frac{r_j}{R_1}}\right\}\right\},$$

where $\widehat{B}_i = \sum_{j=1}^i b_j$ for $1 \leq i \leq n$. Then from Lemma 16 we know that R forms a feasible solution for **OPT_RF'** iff $R \geq \max\left\{R_1, \frac{\widehat{B}_n R_1}{\sum_{i=1}^n r_i d_i}\right\}$ and $X_F(R) \leq Y_F(R)$. Therefore, R^* is the minimum value satisfying these conditions.

Given R^*, from Lemma 16 we know that there exists an optimal solution with $\widehat{B}_n'^* = X_F(R^*)$. From Statement 2 in Lemma 17, we know that suppose there exists an optimal solution with $\widehat{B}_n' = \widehat{B}_n'^*$ and $\widehat{B}_i' = \widehat{B}_i'^*$ where $2 \leq i \leq n$, then there exists an optimal solution with $\widehat{B}_n' = \widehat{B}_n'^*$, $\widehat{B}_i' = \widehat{B}_i'^*$, and $\widehat{B}_{i-1}' = \max\left\{\sum_{j=1}^{i-1} T_j, \widehat{B}_i'^* - b_i\right\}$. Thus, based on $\widehat{B}_n'^*$ we can sequentially compute $\widehat{B}_i'^*$ from $i = n-1$ to $i = 1$.

Lemma 16. *Define* $T_i = \max\left\{0, T_i^{(1)}, T_i^{(2)}\right\}$*. Then* $\left\{\widehat{\boldsymbol{B}}' \mid T_h \le \widehat{B}'_h - \widehat{B}'_{h-1} \le b_h,\ \forall 1 \le h \le n\right\} \ne \emptyset$ *iff all of the following conditions hold:*

- $R \ge \frac{\sum_{i=1}^n b_i R_1}{\sum_{i=1}^n r_i d_i} = \frac{\widehat{B}_n R_1}{\sum_{i=1}^n r_i d_i}$
- $\widehat{B}'_n \ge \max_{P_1,P_2\subseteq\mathbb{Z}_n, P_2\ne\mathbb{Z}_n, P_1\bigcap P_2=\emptyset} \frac{\sum_{i\in P_1}\frac{RH_i}{R+r_i}+\sum_{p\in P_2}\left(b_i-\frac{r_i d_i R}{R_1}\right)}{1-\sum_{i\in P_1}\frac{r_i}{R+r_i}-\sum_{i\in P_2}\frac{r_i}{R_1}}$,
- $\widehat{B}'_n \le \min_{1\le i\le n-1}\left\{\widehat{B}_n, Rd_n, \min_{P_1,P_2\subseteq\mathbb{Z}_i, P_1\bigcap P_2=\emptyset, P_1\bigcup P_2\ne\emptyset}\left\{\frac{\widehat{B}_i-\sum_{j\in P_1}\frac{RH_j}{R+r_j}-\sum_{j\in P_2}\left(b_j-\frac{r_j d_j R}{R_1}\right)}{\sum_{j\in P_1}\frac{r_j}{R+r_j}+\sum_{j\in P_2}\frac{r_j}{R_1}}\right\}\right\}$

Proof. Define $\widehat{B}_i = \sum_{j=1}^i b_j$ for $1 \le i \le n$, and define $\widehat{\mathbb{B}}_i = \left[\max\left\{T_i, \widehat{B}'_{i+1} - b_{i+1}\right\},\ \min\left\{\widehat{B}_i, \widehat{B}'_{i+1} - T_{i+1}\right\}\right]$ for $1 \le i \le n-1$. As $\widehat{B}'_0 = 0$, from basic algebraic manipulation $T_i = \max\left\{0, T_i^{(1)}, T_i^{(2)}\right\} \le \widehat{B}'_i - \widehat{B}'_{i-1} \le b_i,\ \forall 1 \le i \le n$ is equivalent to

$$\begin{cases}\widehat{B}'_1 \in \left[\max\left\{T_1, \widehat{B}'_2 - b_2\right\},\ \min\left\{b_1, \widehat{B}'_2 - T_2\right\}\right] = \widehat{\mathbb{B}}_1, \\ T_i \le \widehat{B}'_i - \widehat{B}'_{i-1} \le b_i,\ \forall\ 2 < i \le n\end{cases}$$

Define $\widehat{\boldsymbol{B}}'_i = \left\{\widehat{B}_i, ..., \widehat{B}_n\right\}$, and $\widehat{\mathbb{B}}^{(i)} = \left\{\widehat{\boldsymbol{B}}'_i \mid T_h \le \widehat{B}'_h - \widehat{B}'_{h-1} \le b_h,\ \forall i < h \le n\right\}$. Then we have

$$\left\{\boldsymbol{B}' \mid T_i \le \widehat{B}'_i - \widehat{B}'_{i-1} \le b_i,\ \forall 1 \le i \le n\right\} \ne \emptyset \iff \widehat{\mathbb{B}}_1 \ne \emptyset \text{ and } \widehat{\mathbb{B}}^{(2)} \ne \emptyset,$$

which from Lemma 17 is equivalent to

$$\left\{\sum_{j=1}^n T_j \le \widehat{B}'_n \le \min\left\{\widehat{B}_n, Rd_n\right\} \mid \sum_{j=1}^i T_j \le \widehat{B}_i,\ \forall 1 \le i \le n-1\right\} \ne \emptyset. \tag{24}$$

Denote $\mathbb{Z}_i = \{1 \le j \le i \mid j \in \mathbb{Z}\}$ for $1 \le i \le n$, then

- $\sum_{j=1}^n T_j \le \widehat{B}'_n$ implies that for all $P_1, P_2 \subseteq \mathbb{Z}_n$ and $P_1 \bigcap P_2 = \emptyset$, $\sum_{i\in P_1} T_i^{(1)} + \sum_{i\in P_2} T_i^{(2)} \le B'_n$, *i.e.*, $\sum_{i\in P_1}\frac{RH_i}{R+r_i} + \sum_{p\in P_2}\left(b_i - \frac{r_i d_i R}{R_1}\right) \le \left(1 - \sum_{i\in P_1}\frac{r_i}{R+r_i} - \sum_{i\in P_2}\frac{r_i}{R_1}\right) B'_n$, which is equivalent to $R \ge \frac{\sum_{i=1}^n b_i R_1}{\sum_{i=1} r_i d_i}$ when $P_2 = \mathbb{Z}_n$, and $\widehat{B}'_n \ge \frac{\sum_{i\in P_1}\frac{RH_i}{R+r_i}+\sum_{p\in P_2}\left(b_i-\frac{r_i d_i R}{R_1}\right)}{1-\sum_{i\in P_1}\frac{r_i}{R+r_i}-\sum_{i\in P_2}\frac{r_i}{R_1}}$ otherwise.
- For $1 \le i \le n-1$, $\sum_{j=1}^i T_j \le \widehat{B}_i$ implies that for all $P_1, P_2 \subseteq \mathbb{Z}_i$ and $P_1 \bigcap P_2 = \emptyset$, $\sum_{i\in P_1} T_i^{(1)} + \sum_{i\in P_2} T_i^{(2)} \le \widehat{B}_i$, *i.e.*, $\sum_{i\in P_1}\frac{RH_i}{R+r_i} + \sum_{p\in P_2}\left(b_i - \frac{r_i d_i R}{R_1}\right) + \left(\sum_{i\in P_1}\frac{r_i}{R+r_i} + \sum_{i\in P_2}\frac{r_i}{R_1}\right) B'_n \le \widehat{B}_i$, which is equivalent to $\widehat{B}'_n \le \frac{\widehat{B}_i-\sum_{j\in P_1}\frac{RH_j}{R+r_j}-\sum_{j\in P_2}\left(b_j-\frac{r_j d_j R}{R_1}\right)}{\sum_{j\in P_1}\frac{r_j}{R+r_j}+\sum_{j\in P_2}\frac{r_j}{R_1}}$.

Therefore, Eq. (24) holds iff

$$\begin{cases} R \ge \dfrac{\sum_{i=1}^n b_i R_1}{\sum_{i=1} r_i d_i} \\ \widehat{B}'_n \ge \max\limits_{P_1,P_2\subseteq\mathbb{Z}_n, P_2\ne\mathbb{Z}_n, P_1\bigcap P_2=\emptyset} \dfrac{\sum_{i\in P_1}\frac{RH_i}{R+r_i} + \sum_{i\in P_2}\left(b_i - \frac{r_i d_i R}{R_1}\right)}{1 - \sum_{i\in P_1}\frac{r_i}{R+r_i} - \sum_{i\in P_2}\frac{r_i}{R_1}} \\ \widehat{B}'_n \le \min\limits_{1\le i\le n-1}\left\{\widehat{B}_n, Rd_n, \min\limits_{P_1,P_2\subseteq\mathbb{Z}_i, P_1\bigcap P_2=\emptyset, P_1\bigcup P_2\ne\emptyset}\left\{\dfrac{\widehat{B}_i - \sum_{j\in P_1}\frac{RH_j}{R+r_j} - \sum_{j\in P_2}\left(b_j - \frac{r_j d_j R}{R_1}\right)}{\sum_{j\in P_1}\frac{r_j}{R+r_j} + \sum_{j\in P_2}\frac{r_j}{R_1}}\right\}\right\} \end{cases}$$

□

Next we establish Lemma 17. For $\widehat{\mathbb{B}}_1 = \left[\max\left\{T_1, \widehat{B}'_2 - b_2\right\},\ \min\left\{b_1, \widehat{B}'_2 - T_2\right\}\right]$, $\widehat{\boldsymbol{B}}'_2 = \left\{\widehat{B}'_2, ..., \widehat{B}'_n\right\}$, and $\widehat{\mathbb{B}}^{(2)} = \left\{\widehat{\boldsymbol{B}}'_2 \mid T_h \le \widehat{B}'_h - \widehat{B}'_{h-1} \le b_h,\ \forall 2 < h \le n\right\}$, we have

Lemma 17. $\widehat{\mathbb{B}}_1 \ne \emptyset$ *and* $\widehat{\mathbb{B}}^{(2)} \ne \emptyset$ *iff*

$$\left\{\sum_{j=1}^n T_j \le \widehat{B}'_n \le \min\left\{\widehat{B}_n, Rd_n\right\} \mid \sum_{j=1}^i T_j \le \widehat{B}_i,\ \forall 1 \le i \le n-1\right\} \ne \emptyset.$$

Proof. For $1 \le i \le n$ define $\widehat{B}_i = \sum_{j=1}^{i} b_j$ and $\widehat{\boldsymbol{B}}'_i = \left\{\widehat{B}_i, ..., \widehat{B}_n\right\}$. For $1 \le i \le n-1$ further define $\widehat{\mathbb{B}}_i = \left[\max\left\{\sum_{j=1}^{i} T_j, \widehat{B}'_{i+1} - b_{i+1}\right\},\right.$ and
$\widehat{\mathbb{B}}^{(i)} = \left\{\widehat{\boldsymbol{B}}'_i \mid T_h \le \widehat{B}'_h - \widehat{B}'_{h-1} \le b_h,\ \forall i < h \le n\right\}$. Suppose we have

Statement 1. $\widehat{\mathbb{B}}_1 \neq \emptyset$ and $\widehat{\mathbb{B}}^{(2)} \neq \emptyset$ iff 1) $\widehat{\mathbb{B}}_{n-1} \neq \emptyset$, 2) $\sum_{j=1}^{h} T_h \le \widehat{B}_h$, for $1 \le h \le n-2$, and 3) $\widehat{B}'_n \le Rd_{n-1}$.

Basic algebraic manipulations give that $\widehat{\mathbb{B}}_{n-1} \neq \emptyset$, *i.e.,* $\max\left\{\sum_{j=1}^{n-1} T_j, \widehat{B}'_n - b_n\right\} \le \min\left\{\widehat{B}_{n-1}, \widehat{B}'_n - T_n\right\}$, iff 1) $\sum_{j=1}^{n} T_j \le \widehat{B}'_n \le \widehat{B}_n$, 2) $\sum_{j=1}^{n-1} T_j \le \widehat{B}_{n-1}$, and 3) $\widehat{B}'_n \le Rd_n$. Combining them with $\sum_{j=1}^{h} T_h \le \widehat{B}_h$, for $1 \le h \le n-2$ and $\widehat{B}'_n \le Rd_{n-1}$ gives $\sum_{j=1}^{n} T_j \le \widehat{B}'_n \le \min\left\{\widehat{B}_n, Rd_n\right\}$ and $\sum_{j=1}^{h} T_h \le \widehat{B}_h$, for $1 \le h \le n-1$. Therefore, we have Lemma 17.

Next, we show Statement 1 based on Statement 2. For convenience, define $\widehat{\mathbb{B}}^{(n)} = \{1\}$. Then we have

Statement 2. For $1 \le i \le n-2$, $\widehat{\mathbb{B}}_i \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+1)} \neq \emptyset$ iff 1) $\widehat{\mathbb{B}}_{i+1} \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+2)} \neq \emptyset$, 2) $\sum_{j=1}^{i} T_j \le \widehat{B}_i$, and 3) $\widehat{B}'_n \le Rd_{i+1}$.

a) Proof for Statement 1: we show Statement 1 by induction. For $1 \le i \le n-1$, define

$$S_i : \widehat{\mathbb{B}}_i \neq \emptyset,\ \widehat{\mathbb{B}}^{(i+1)} \neq \emptyset,\ \widehat{B}'_n \le Rd_i,\ \text{and}\ \sum_{j=1}^{h} T_h \le \widehat{B}_h, \forall 1 \le h \le i-1.$$

- When $i = 1$, Statement 2 directly gives that $\widehat{\mathbb{B}}_1 \neq \emptyset$ and $\widehat{\mathbb{B}}^{(2)} \neq \emptyset$ iff S_2 holds.
- When $i \ge 1$, suppose $\widehat{\mathbb{B}}_1 \neq \emptyset$ and $\widehat{\mathbb{B}}^{(2)} \neq \emptyset$ iff S_i holds, *i.e.,* 1) $\widehat{\mathbb{B}}_i \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+1)} \neq \emptyset$, 2) $\widehat{B}'_n \le Rd_i$, and 3)$\sum_{j=1}^{h} T_h \le \widehat{B}_h, \forall 1 \le h \le i-1$. Note that by Statement 2 we have $\widehat{\mathbb{B}}_i \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+1)} \neq \emptyset$ iff i) $\widehat{\mathbb{B}}_{i+1} \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+2)} \neq \emptyset$, ii) $\sum_{j=1}^{i} T_i \le \widehat{B}_i$, and iii) $\widehat{B}'_n \le Rd_{i+1}$. Thus, $\widehat{\mathbb{B}}_1 \neq \emptyset$ and $\widehat{\mathbb{B}}^{(2)} \neq \emptyset$ iff 1) $\widehat{\mathbb{B}}_{i+1} \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+2)} \neq \emptyset$, 2) $\widehat{B}'_n \le \min\{Rd_i, Rd_{i+1}\} = Rd_{i+1}$, and 3) $\sum_{j=1}^{h} T_h \le \widehat{B}_h, \forall 1 \le h \le i$, *i.e.,* S_{i+1} holds.

Thus, we have $\widehat{\mathbb{B}}_1 \neq \emptyset$ and $\widehat{\mathbb{B}}^{(2)} \neq \emptyset$ iff S_{n-1} holds: $\widehat{\mathbb{B}}_{n-1} \neq \emptyset,\ \widehat{\mathbb{B}}^{(n)} = \{1\} \neq \emptyset,\ \widehat{B}'_n \le Rd_{n-1}$, and $\sum_{j=1}^{h} T_h \le \widehat{B}_h, \forall 1 \le h \le n-2$, which then gives Statement 1.

b) Proof for Statement 2: Consider $\widehat{\mathbb{B}}_i \neq \emptyset$ and $\widehat{\mathbb{B}}^{(i+1)} \neq \emptyset$.

- $\widehat{\mathbb{B}}_i \neq \emptyset$ iff $\max\left\{\sum_{j=1}^{i} T_j, \widehat{B}'_{i+1} - b_{i+1}\right\} \le \min\left\{\widehat{B}_i, \widehat{B}'_{i+1} - T_{i+1}\right\}$, which from basic algebraic manipulation is equivalent to 1) $\sum_{j=1}^{i+1} T_j \le \widehat{B}'_{i+1} \le \widehat{B}_{i+1}$, 2) $\sum_{j=1}^{i} T_j \le \widehat{B}_i$, and 3) $b_{i+1} \ge T_{i+1} \iff \widehat{B}'_n \le Rd_{i+1}$.
- Consider $\widehat{\mathbb{B}}^{(i+1)} \neq \emptyset$. When $i < n-2$, from basic algebraic manipulation it is equivalent to 1) $T_{n+2} \le \widehat{B}'_{i+2} - \widehat{B}'_{i+1} \le b_{i+2}$ and 2) $\widehat{\mathbb{B}}^{(i+2)} \neq \emptyset$. When $i = n-2$, $\widehat{\mathbb{B}}^{(i+1)} = \left\{\widehat{\boldsymbol{B}}'_{n-1} \mid T_n \le \widehat{B}'_n - \widehat{B}'_{n-1} \le b_n\right\}$, which is non-empty iff $T_n \le \widehat{B}'_n - \widehat{B}'_{n-1} \le b_n$. Since $\widehat{\mathbb{B}}^n = \{1\}$, it also has $\widehat{\mathbb{B}}^{(i+2)} \neq \emptyset$.

Note that $\sum_{j=1}^{i+1} T_j \le \widehat{B}'_{i+1} \le \widehat{B}_{i+1}$ and $T_{n+2} \le \widehat{B}'_{i+2} - \widehat{B}'_{i+1} \le b_{i+2}$ iff $\widehat{\mathbb{B}}_{i+1} \neq \emptyset$. Thus we have Statement 2. ☐

APPENDIX E
ON THE BENEFIT OF GROUPING FLOWS WITH DIFFERENT DEADLINES

In this section, we explore scenarios that consist of two flows sharing a common link whose access is arbitrated by a static priority scheduler. The goal is to identify configurations that minimize the link bandwidth required to meet the flows' deadlines. Of particular interest is assessing when the two flows should be assigned different priorities or instead merged into the same priority class.

Recalling the discussion of Section VI-A, specializing Propositions 7 and 10 to two flows, we find that the minimum required bandwidth for the two-flow scenario under static priority+shaping is

$$\widetilde{R}^*_R = \begin{cases} max\left\{r_1 + r_2, \dfrac{b_2}{d_2}, \dfrac{b_1 + b_2 - r_2 d_2}{d_1} + r_2\right\}, & \text{when } \dfrac{b_2}{r_2} \ge \dfrac{b_1}{r_1} \\ max\left\{r_1 + r_2, \dfrac{b_2}{d_2}, \dfrac{b_1 + \max\{b_2 - r_2 d_2, 0\}}{d_1} + r_2\right\}, & \text{otherwise} \end{cases} \tag{13}$$

and that under fifo+shaping it is

$$\widehat{R}^*_R = \max\left\{r_1 + r_2, \frac{b_2}{d_2}, \frac{(b_1 + b_2)(r_2 + r_2)}{d_1 r_1 + d_2 r_2}, \frac{b_1 + b_2 - d_1 r_1 + \sqrt{(b_1 + b_2 - d_1 r_1)^2 + 4 r_1 d_2 b_2}}{2 d_2}\right\}. \tag{15}$$

Comparing them gives

Proposition 18. *For the two-flow scenario,* $\widetilde{R}^*_R > \widehat{R}^*_R$ *iff*

$$d_1 \in \left(\frac{b_2}{r_2}, \frac{b_1}{r_1}\right) \text{ and } d_2 \in \left(\frac{(b_1 + b_2)(r_1 + r_2)}{r_2(b_1/d_1 + r_2)} - \frac{d_1 r_1}{r_2}, d_1\right).$$

Proof. When $\widetilde{R}_R^* = max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$, from Eq. (14) $\widetilde{R}_R^* \leq \widehat{R}_R^*$. Below we consider 1) $\frac{b_2}{d_2} \geq \frac{b_1}{d_1}$ and 2) $\frac{b_2}{d_2} < \frac{b_1}{d_1}$ separately under $\widetilde{R}_R^* > max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$.

1) When $\frac{b_2}{d_2} \geq \frac{b_1}{d_1}$, we show that $\widetilde{R}_R^* \leq \widehat{R}_R^*$. Specifically, $\widetilde{R}_R^* > max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$ iff $\frac{b_1+b_2-r_2d_2}{d_1}+r_2 > \max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$, which from basic algebraic manipulation equivalents to $b_1+b_2 > r_1d_1+r_2d_2$ and $d_2 < \frac{b_1+b_2-\sqrt{(b_1+b_2)^2-4r_2b_2d_1}}{2r_2}$. Next we show that under $b_1+b_2 > r_1d_1+r_2d_2$, it has $\frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2} \geq \frac{b_1+b_2-r_2d_2}{d_1}+r_2$, and therefore $\widetilde{R}_R^* \leq \widehat{R}_R^*$. Consider $f(d_2) = \frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2} - \frac{b_1+b_2-r_2d_2}{d_1} - r_2$, which equals 0 when $d_2 = d_1$. Basic algebraic gives that
$$\begin{aligned}\frac{df(d_2)}{dd_2} &= \frac{r_2}{d_1} - \frac{(b_1+b_2)(r_2+r_2)r_2}{(d_1r_1+d_2r_2)^2} \\ &= \frac{r_2}{(d_1r_1+d_2r_2)^2}\left(\frac{(d_1r_1+d_2r_2)^2}{d_1} - (b_1+b_2)(r_2+r_2)\right) \\ &\leq \frac{r_2}{(d_1r_1+d_2r_2)^2}\left(\frac{(d_1r_1+d_2r_2)^2}{d_1} - (d_1r_1+d_2r_2)(r_2+r_2)\right) \\ &= \frac{r_2^2(d_2-d_1)}{d_1(d_1r_1+d_2r_2)} \leq 0\end{aligned}$$
Thus, for all $d_2 \leq d_1$, it has $f(d_2) \geq f(d_1) = 0$, *i.e.,* $\frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2} \geq \frac{b_1+b_2-r_2d_2}{d_1}+r_2$.

2) When $\frac{b_2}{d_2} < \frac{b_1}{d_1}$, we show that $\widetilde{R}_R^* > \widehat{R}_R^*$ iff $d_1 \in \left(\frac{b_2}{r_2}, \frac{b_1}{r_1}\right)$ and $d_2 \in \left(\frac{(b_1+b_2)(r_1+r_2)}{r_2(b_1/d_1+r_2)} - \frac{d_1r_1}{r_2}, d_1\right)$. Specifically, $\widetilde{R}_R^* > max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$ iff $\frac{b_1+\max\{b_2-r_2d_2,0\}}{d_1}+r_2 > \max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$, which from basic algebraic manipulation equivalents to
$$\begin{cases}\frac{b_1+b_2-r_2d_2}{d_1}+r_2 > \max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}, & \text{when } d_2 \leq \frac{b_2}{r_2} \\ \frac{b_1}{d_1}+r_2, & \text{otherwise}\end{cases}.$$
When $d_2 \leq \frac{b_2}{r_2}$, similar as before we have $\widetilde{R}_R^* \leq \widehat{R}_R^*$. When $d_2 > \frac{b_2}{r_2}$, basic algebraic manipulation gives that $\frac{b_1}{d_1}+r_2 > max\left\{r_1+r_2, \frac{b_2}{d_2}\right\}$ iff $d_1 < \frac{b_1}{r_1}$. Combining them gives that $\widetilde{R}_R^* \leq \widehat{R}_R^*$ when $(d_1,d_2) \notin \left\{(d_1,d_2) \mid \frac{b_2}{r_2} < d_2 < d_1 < \frac{b_1}{r_1}\right\}$. When $(d_1,d_2) \in \left\{(d_1,d_2) \mid \frac{b_2}{r_2} < d_2 < d_1 < \frac{b_1}{r_1}\right\}$, basic algebraic manipulation gives that $\frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2} > \frac{b_1+b_2-d_1r_1+\sqrt{(b_1+b_2-d_1r_1)^2}}{2d_2}$ and $r_1+r_2 > \frac{b_2}{d_2}$, *i.e.,*
$$\widehat{R}_R^* = \max\left\{r_1+r_2, \frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2}\right\} = \begin{cases}r_1+r_2, & \text{if } b_1+b_2 \leq r_1d_1+r_2d_2 \\ \frac{(b_1+b_2)(r_2+r_2)}{d_1r_1+d_2r_2}, & \text{otherwise}\end{cases}$$
When $b_1+b_2 \leq r_1d_1+r_2d_2$, it has $\widetilde{R}_R^* > \widehat{R}_R^*$. Otherwise, basic algebraic manipulation gives that $\widetilde{R}_R^* > \widehat{R}_R^*$ iff $d_2 > \frac{(b_1+b_2)(r_1+r_2)}{r_2(b_1/d_1+r_2)} - \frac{d_1r_1}{r_2} := g(d_1)$. Note that when $\frac{b_2}{r_2} < d_1 < \frac{b_1}{r_1}$, $g(d_1) > \frac{b_2}{r_2}$; and when $d_1 = \frac{b_2}{r_2}$ or $d_1 = \frac{b_1}{r_1}$, $g(d_1) = d_1$. Therefore, $\widetilde{R}_R^* > \widehat{R}_R^*$ iff $d_2 \in \left(\frac{(b_1+b_2)(r_1+r_2)}{r_2(b_1/d_1+r_2)} - \frac{d_1r_1}{r_2}, d_1\right)$.

□

APPENDIX F
EXTENSIONS TO PACKET-BASED MODELS IN THE TWO-FLOW STATIC PRIORITY CASE

In this section, we consider a more general packet-based model, where flow i has maximum packet size of l_i and the scheduler relies on static priorities. For ease of exposition, we only consider scenarios that consist of $n = 2$ flows, and consequently two priority classes (low and high).

We first characterize in Proposition 19 the worst-case delay of high-priority packets, and use the result to identify a condition for when adding a reprofiler can help lower the required bandwidth (Corollary 20). We also confirm (Proposition 21) the intuitive property that reshaping the low-priority flow does not contribute to lowering the required bandwidth. We then proceed to characterize in Proposition 22 the worst-case delay of low-priority packets. The results of Propositions 19 and 22 are used to formulate an optimization, **OPT_2**, that seeks to identify the optimum reshaping parameters for the high-priority flow that minimizes the link bandwidth required to meet the flows' deadlines. The bulk the section is devoted to solving this optimization, while also establishing the intermediate result that the optimal reshaping can be realized simply by reducing the flow's burst size, *i.e.,* keeping its rate constant.

Returning to our two-flow, packet-based scenario, consider two token-bucket controlled flows (r_1,b_1) and (r_2,b_2) sharing a link with a bandwidth of R whose access is controlled by a static priority scheduler. Assume that flow (r_i,b_i) has a deadline of d_i ($d_1 > d_2 > 0$), and (r_2, b_2) has non-preemptive priority over (r_1, b_1) at the shared link. Denote the maximum packet length of (r_i, b_i) as $l_i < b_i$. Note that by setting $l_i = 0$, the packet-based model defaults to the fluid model. To guarantee that none of the packets in (r_1, b_1) or (r_2, b_2) misses the deadline, R needs to satisfy [17]:

$$r_1 + r_2 \leq R, \tag{25a}$$

$$\frac{b_2 + l_1}{R} \leq d_2, \tag{25b}$$

$$\frac{b_2 + b_1}{R - r_2} \leq d_1. \tag{25c}$$

Therefore, the minimum bandwidth for the link satisfies:

$$\tilde{R}^{(2)*} = \max\left\{r_1 + r_2, \frac{l_1 + b_2}{d_2}, \frac{b_1 + b_2}{d_1} + r_2\right\}. \tag{26}$$

Now consider adding a lossless packet-based greedy leaky-bucket (re)profiler for each flow before the shared link, as shown in the above figure. Denote (r_i, b_i)'s reprofiler as (r'_i, b'_i), where $b'_i \geq l_i$. To guarantee a finite delay inside the reprofiler, we also need $r'_i \geq r_i$. Under this assumption, if $b'_i \geq b_i$, the reprofiler has no effect. Hence, we further require that $b'_i < b_i$.

Next, we proceed to characterize the optimal minimum required bandwidth under static priority and (re)shaping. Denote it as $\tilde{R}_R^{(2)*}$.

Under a non-preemptive static priority discipline, the worst-case delay of the high-priority flow is unaffected by the low-priority flow's arrival curve (r_1, b_1), and only depends on its maximum packet size l_1. This is because high-priority packets arriving to an empty high-priority queue wait for at most the transmission time of one low-priority packet. This property holds whether reprofilers are present or not. Specifically,

Proposition 19. *For a high priority token bucket-controlled flow (r_2, b_2) traversing a lossless packet-based greedy token bucket reprofiler (r'_2, b'_2), where $r_2 \leq r'_2$ and $b_2 > b'_2$, before going through a shared link with bandwidth $R > r_2$, the worst-case delay is*

$$D_2^* = \max\left\{\frac{b_2 + l_1}{R}, \frac{b_2 - b'_2}{r'_2} + \frac{l_1 + l_2}{R}\right\},$$

where l_1 is the maximum packet length of the low-priority flow with which it shares the link, and l_2 is its own maximum packet length.

Proof. Denote the virtual delay at t inside the reprofiler as $D_1(t)$, and that at the shared link as $D_2(t)$. Then for a packet arriving the system at t, its virtual delay inside the system is $D_1(t) + D_2(t + D_1(t))$. For $D_1(t)$, we have

$$D_1(t) = \inf_{0\leq\tau}\{b_2 + r_2 t \leq b'_2 + t'_2(t+\tau)\} = \left[\frac{b_2 - b'_2 + r_2 t}{r'_2} - t\right]^+.$$

For $D_2(t + D_1(t))$, we have

$$D_2(t + D_1(t)) = \frac{l_1 + l_2}{R} + \inf_{0\leq\tau}\{b_2 + r_2 t - l_2 \leq R(t + \tau + D_1(t))\} = \frac{l_1 + l_2}{R} + \left[\frac{b_2 + r_2 t - l_2}{R} - t - D_1(t)\right]^+.$$

Then we have

$$\begin{aligned} D_2^* &= \sup_{0\leq t}\{D_1(t) + D_2(t + D_1(t))\} \\ &\leq \sup_{0\leq t}\left\{\max\left\{\frac{l_1 + l_2}{R} + \frac{b_2 - b'_2 + r_2 t}{r'_2} - t, \frac{l_1 + l_2}{R} + \frac{b_2 + r_2 t - l_2}{R} - t\right\}\right\} \\ &\leq \max\left\{\frac{l_1 + l_2}{R} + \frac{b_2 - b'_2}{r'_2}, \frac{l_1 + b_2}{R}\right\} \end{aligned} \tag{27}$$

□

Note that after adding the reprofiler, we have $D_2^* \geq \frac{b_2 + l_1}{R}$. Comparing this expression to Eq. (25b), we know that adding reprofilers will never decrease the high-priority flow's worst-case delay. Furthermore, since ensuring stability of the shared queue mandates $R \geq r_1 + r_2$ irrespective of whether reprofilers are used, Eq. (26) then gives

Corollary 20. *Adding reprofilers decreases the minimum required bandwidth only when*

$$\tilde{R}^{(2)*} = \frac{b_1 + b_2}{d_1} + r_2 > \max\left\{r_1 + r_2, \frac{b_2 + l_1}{d_2}\right\}. \tag{28}$$

Conversely, we note that for the low-priority flow (r_1, b_1), its service curve is determined by both the high-priority flow's arrival curve at the shared link and the shared link's bandwidth, and does not depend on the presence or absence of its own reprofiler. As a result, adding a reprofiler to the low-priority flow cannot decrease its worst-case delay (though it can increase it). Consequently, reprofiling the low-priority flow cannot contribute to reducing the bandwidth of the shared link while meeting the delay bounds of both the high and low-priority flows. This is formally stated in the next proposition that simplifies the investigation of the worst-case delay experienced by the low-priority flow by allowing us to omit the use of a reprofiler for it.

Proposition 21. *Given the service curve assigned to a low-priority flow, adding a packet-based greedy reprofiler cannot decrease its worst-case delay, and consequently cannot reduce the minimum link bandwidth required to meet the worst-case delay guarantees of both the high and low-priority flows.*

Proof. Denote the service curve of the low-priority flow as $\beta(t)$ and its arrival curve as $\alpha(t)$. Without reprofiler, the system's virtual delay at t is

$$D'(t) = \inf_{\tau \geq 0}\{\tau : \alpha(t) \leq \beta(t+\tau)\}.$$

Denote the reprofiler's maximum service curve as $\sigma(t)$, which is also the arrival curve for the shared link. Due to packetization, the system provides the flow a service curve no greater than

$$\sigma \otimes \beta(t) = \inf_{0 \leq s \leq t}\{\sigma(s) + \beta(t-s)\} \leq \sigma(0) + \beta(t) = \beta(t),$$

Hence, the virtual delay is

$$D(t) \geq \inf_{\tau \geq 0}\left\{\tau : \alpha(t) \leq \inf_{0 \leq s \leq t}\{\sigma(s) + \beta(t+\tau-s)\}\right\} \geq \inf_{\tau \geq 0}\{\tau : \alpha(t) \leq \beta(t+\tau)\} = D'(t).$$

As $D(t) \geq D'(t)$, $\forall t \geq 0$, we have $\sup_{t \geq 0}\{D(t)\} \geq \sup_{t \geq 0}\{D'(t)\}$, *i.e.,* adding a reprofiler cannot decrease the system's worst-case delay.

□

Next, we characterize the worst-case delay D_1^* of the low-priority flow.

Proposition 22. *Given a token bucket-controlled high priority flow (r_2, b_2) with a packet-based greedy token bucket reprofiler (r_2', b_2') going through a shared link with bandwidth R, the low-priority flow (r_1, b_1)'s worst-case delay d_1^* is*

1) *when $r_2 = r_2'$, $D_1^* = \frac{b_1+b_2'}{R-r_2}$;*
2) *when $r_2 < r_2'$ and $\frac{(R-r_2')(b_2-b_2')}{r_2'-r_2} - (b_1+b_2') < 0$, $D_1^* = \frac{b_1+b_2}{R-r_2}$;*
3) *otherwise,* i.e., $r_2 < r_2'$ *and* $\frac{(R-r_2')(b_2-b_2')}{r_2'-r_2} - (b_1+b_2') \geq 0$ *(recall that $r_2 \leq r_2'$)* $D_1^* = \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\}$.

Proof. The high-priority flow's arrival curve at the shared link is

$$\alpha_2(t) = \min\{\gamma_{r,b}(t), \gamma_{r',b'}(t)\}, \tag{29}$$

and the low-priority flow's service curve at the shared link is

$$\beta_1(t) = [Rt - \alpha_2(t)]^+ = \left[Rt - \min\left\{\gamma_{r_2,b_2}(t), \gamma_{r_2',b_2'}(t)\right\}\right]^+. \tag{30}$$

Hence, the virtual delay at $t > 0$ is

$$D(t) = \inf_{\tau \geq 0}\left\{\tau : r_1 t + b_1 \leq [R(t+\tau) - \min\{r_2(t+\tau) + b_2, r_2'(t+\tau) + b_2'\}]^+\right\} \tag{31}$$

When $r_2 = r_2'$, we have

$$\begin{aligned} D(t) &= \inf_{\tau \geq 0}\left\{\tau : r_1 t + b_1 \leq [R(t+\tau) - r_2(t+\tau) - b_2']^+\right\} \\ &= \left[\frac{r_1 t + b_1 + b_2'}{R - r_2} - t\right]^+ \leq \frac{b_1 + b_2'}{R - r_2}. \end{aligned} \tag{32}$$

When $r_2 < r_2'$, we can rewrite Eq. (30) as

$$\beta_1(t) = [Rt - \min\{\gamma_{r,b}(t), \gamma_{r',b'}(t)\}]^+ = \begin{cases} 0 & \text{when } t = 0 \\ [(R-r_2')t - b_2']^+ & \text{when } t < \frac{b_2-b_2'}{r_2'-r_2} \\ [(R-r_2)t - b_2]^+ & \text{otherwise.} \end{cases} \tag{33}$$

- When $\beta_1(\frac{b_2-b_2'}{r_2'-r_2}) \le b_1 + r_1 t$, *i.e.,* $t \ge \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1}$ we have

$$\begin{aligned} D(t) &= \inf_{\tau\ge 0}\left\{\tau : r_1 t + b_1 \le [(R-r_2)(t+\tau) - b_2]^+\right\} \\ &= \left[\frac{b_1+b_2}{R-r_2} - \frac{t(R-r_1-r_2)}{R-r_2}\right]^+ \\ &\le \left[\frac{b_1+b_2}{R-r_2} - \frac{R-r_1-r_2}{R-r_2}\left[\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1}\right]^+\right]^+ \\ &= \begin{cases} \frac{b_1+b_2}{R-r_2} & \text{when } \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} < 0 \\ \left[\frac{b_1+b_2}{r_1} - \frac{(b_2-b_2')(R-r_1-r_2)}{r_1(r_2'-r_2)}\right]^+ & \text{otherwise.} \end{cases} \end{aligned} \tag{34}$$

Note that when $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} < 0$, we have $t \ge \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1}, \forall t$. Therefore, $d_1^* = \frac{b_1+b_2}{R-r_2}$. Next we consider the case when $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} \ge 0$.

- When $\beta_1(\frac{b_2-b_2'}{r_2'-r_2}) > b_1 + r_1 t$, *i.e.,* $t < \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1}$. As when $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} > 0$ implies $R - r_2' > 0$, we have

$$\begin{aligned} D(t) &= \inf_{\tau\ge 0}\left\{\tau : r_1 t + b_1 \le [(R-r_2')(t+\tau) - b_2']^+\right\} \\ &= \left[\frac{b_1+b_2'}{R-r_2'} + \frac{t(r_1+r_2'-R)}{R-r_2'}\right]^+, \end{aligned} \tag{35}$$

As $\left[\frac{b_1+b_2'}{R-r_2'} + \frac{t(r_1+r_2'-R)}{R-r_2'}\right]^+$ is a linear function with t, we know $D(t)$ achieves its maximum at either $t = 0$ or $t = \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1}$, which gives $d_1^* = \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\}$. Combine it with Eq. (34), when $r_2' > r_2$ we have:

$$D_1^* = \begin{cases} \frac{b_1+b_2}{R-r_2}, & \text{when } \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} < 0 \\ \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\}, & \text{otherwise} \end{cases} \tag{36}$$

□

Note that when $r_2 < r_2'$ and $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} < 0$ (case 2 of Proposition 22), $\tilde{R}_R^{(2)*}$ ensures $d_1 \ge d_1^* = \frac{b_1+b_2}{R^*-r_2}$, *i.e.,* $R^* \ge \frac{b_1+b_2}{d_1} + r_2$. Combined with Corollary 20, we then know that in this case $\tilde{R}_R^{(2)*}$ is no smaller than $\tilde{R}^{(2)*}$, *i.e.,* the optimal system needs no reprofiling. Therefore, we only need to focus on cases 1 and 3 when seeking to characterize $\tilde{R}_R^{(2)*}$ in the presence of reprofilers.

Next, we establish that these two cases can be combined. Specifically, note that $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} \ge 0$ implies $R - r_2' > 0$. This means that as $r_2' \to r_2^+$, $\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} \to +\infty > 0$, so that we are always in case 3 as $r_2' \to r_2^+$. Furthermore, $\lim_{r_2' \to r_2^+} \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\} = \max\left\{\frac{b_1+b_2'}{R-r_2'}, -\infty\right\} = \frac{b_1+b_2'}{R-r_2}$, or in other words the value of d_1^* of case 3 is the same as that of case 1 as $r_2' \to r_2^+$. This therefore allows us to write that when $\frac{(R-r_2')(b_2-b_2')}{r_2'-r_2} - (b_1 + b_2') \ge 0$, $d_1^* = \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\}$.

Together with Proposition 19, this yields the following optimization for $\tilde{R}_R^{(2)*}$:

$$\begin{aligned} \textbf{OPT\_2} \quad & \min_{r_2', b_2'} R \\ \text{subject to} \quad & \max\left\{\frac{b_1+b_2'}{R-r_2'}, \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)}\right\} \le d_1, \\ & \max\left\{\frac{b_2+l_1}{R}, \frac{b_2-b_2'}{r_2'} + \frac{l_1+l_2}{R}\right\} \le d_2, \\ & \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} - \frac{b_1+b_2'}{r_1} \ge 0, \\ & r_1 + r_2 \le R, \qquad r_2 \le r_2', \qquad l_2 \le b_2' \le b_2 \end{aligned} \tag{37}$$

Solving **OPT_2** gives the following combination of five cases, four of which yield values $\tilde{R}_R^{(2)*} < \tilde{R}^{(2)*}$, *i.e.,* the introduction of reprofilers helps reduce the link bandwidth required to meet the flows delay targets, where $r_2'^*$ and $b_2'^*$ defines the optimal reprofiler:

(i) $\tilde{R}_R^{(2)*} = r_1 + r_2 < \tilde{R}^{(2)*}$, $r_2'^* = r_2$, and $b_2'^*$ can be any values inside $[b_2 + r_2\left(\frac{l_1+l_2}{r_1+r_2} - d_2\right), d_1 r_1 - b_1] \cap [l_2, b_2)$, when $d_1 \in [\frac{l_2+b_1}{r_1}, \frac{b_1+b_2}{r_1})$ and $d_2 \geq \max\left\{\frac{b_2+l_1}{r_1+r_2}, \frac{b_1+b_2-d_1r_1}{r_2} + \frac{l_1+l_2}{r_1+r_2}\right\}$;

(ii) $\tilde{R}_R^{(2)*} = \frac{b_2+l_1}{d_2} < \tilde{R}^{(2)*}$, $r_2'^*$ can be any values inside $[r_2, \min\left\{\tilde{R}_R^{(2)*} - r_1, \tilde{R}_R^{(2)*} - \frac{b_1+l_2}{d_1-(b_2-l_2)/\tilde{R}_R^{(2)*}}\right\}]$, and $b_2'^*$ can be any values inside $[b_2 - \frac{r_2'^*(b_2-l_2)}{\tilde{R}_R^{(2)*}}, d_1(\tilde{R}_R^{(2)*} - r_2'^*) - b_1] \cap [l_2, b_2)$, when $d_2 < \frac{b_2+l_1}{r_1+r_2}$ and $d_1 \in [\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$.

(iii) $\tilde{R}_R^{(2)*} = \frac{l_2+b_1}{d_1} + r_2 < \tilde{R}^{(2)*}$, $r_2'^* = r_2$, and $b_2' = l_2$, when $d_1 < \min\left\{\frac{b_1+l_2}{r_1}, \frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}\right\}$ and $d_2 < \frac{l_1+b_2}{r_2}$; and when $d_2 \geq \frac{l_1+b_2}{r_2}$ and $d_1 < \frac{b_1+l_2}{r_1}$.

(iv) $\tilde{R}_R^{(2)*} = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1} < R_0^*$, $r_2'^* = r_2$, and $b_2' = \frac{b_2-b_1-(d_1+d_2)r_2+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2}$

- ∘ when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1})$; and
- ∘ when $\frac{b_2+l_1}{r_1+r_2} \leq d_2 \leq \frac{b_2+l_1}{r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2r_2}{r_1})$.

(v) otherwise, $\tilde{R}_R^{(2)*} = \tilde{R}^{(2)*}$.

From the solution of **OPT_2**, we directly get

Corollary 23. *We can achieve the optimality of **OPT_2** through setting $r_2' = r_2$.*

1) Solving **OPT_2***:* We can divide the optimization into two sub-optimizations:

Sub-optimization 1:

$$
\begin{aligned}
\text{minimize}_{r_2', b_2'} \quad & R \\
\text{subject to} \quad & r_1 + r_2' - R \leq 0, \quad \frac{b_1 + b_2'}{R - r_2'} - d_1 \leq 0, \\
& \frac{b_2 + l_1}{R} - d_2 \leq 0, \quad \frac{b_2 - b_2' + l_2}{r_2'} + \frac{l_1}{R} - d_2 \leq 0, \\
& \frac{b_1 + b_2'}{r_1} - \frac{(R - r_2')(b_2 - b_2')}{r_1(r_2' - r_2)} \leq 0, \\
& r_1 + r_2 - R \leq 0, \quad r_2 - r_2' \leq 0, \\
& l_2 - b_2' \leq 0, \quad b_2' - b_2 \leq 0
\end{aligned} \tag{38}
$$

Sub-optimization 2:

$$
\begin{aligned}
\text{minimize}_{r_2', b_2'} \quad & R \\
\text{subject to} \quad & R - r_1 - r_2' < 0, \quad \frac{b_1 + b_2}{r_1} - \frac{(R - r_1 - r_2)(b_2 - b_2')}{r_1(r_2' - r_2)} - d_1 \leq 0, \\
& \frac{b_2 + l_1}{R} - d_2 \leq 0, \quad \frac{b_2 - b_2' + l_2}{r_2'} + \frac{l_1}{R} - d_2 \leq 0, \\
& \frac{b_1 + b_2'}{r_1} - \frac{(R - r_2')(b_2 - b_2')}{r_1(r_2' - r_2)} \leq 0, \\
& r_1 + r_2 - R \leq 0, \quad r_2 - r_2' \leq 0, \\
& l_2 - b_2' \leq 0, \quad b_2' - b_2 \leq 0
\end{aligned} \tag{39}
$$

Denote the solution of sub-optimizations 1 and 2 as R_1^* and R_2^*, respectively. Then we have $R^* = \min\{R_1^*, R_2^*\}$. Next, we solve sub-optimizations 1 and 2. Note than when $R = r_1 + r_2'$, the two sub-optimizations are the same. Therefore, when solving sub-optimization 2, we only consider the case where $R - r_1 - r_2' < 0$. Then we have:

Lemma 24. *The solution for Sub-optimiaztion 1 is*

- $R_1^* = r_1 + r_2$, *when* $d_1 \in [\frac{l_2+b_1}{r_1}, \frac{b_1+b_2}{r_1})$ *and* $d_2 \geq \max\left\{\frac{b_2+l_1}{r_1+r_2}, \frac{b_1+b_2-d_1r_1}{r_2} + \frac{l_1+l_2}{r_1+r_2}\right\}$;
- $R_1^* = \frac{b_2+l_1}{d_2}$, *when* $d_2 < \frac{b_2+l_1}{r_1+r_2}$ *and* $d_1 \in [\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$;
- $R_1^* = \frac{l_2+b_1}{d_1} + r_2$, *when* $d_2 < \frac{l_1+b_2}{r_2}$ *and* $d_1 < \min\left\{\frac{b_1+l_2}{r_1}, \frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}\right\}$; *and when* $d_2 \geq \frac{l_1+b_2}{r_2}$ *and* $d_1 < \frac{b_1+l_2}{r_1}$;
- $R_1^* = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1}$,
  - ∘ *when* $d_2 < \frac{b_2+l_1}{r_1+r_2}$, *and* $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1})$; *and*

- $\circ$ *when* $\frac{b_2+l_1}{r_1+r_2} \le d_2 \le \frac{b_2+l_1}{r_2}$, *and* $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2r_2}{r_1})$.

Lemma 25. *The solution for Sub-optimization* 2 *is*

- *when* $d_2 < \frac{b_2+l_1}{r_1+r_2}$, *and* $d_1 \in [\frac{d_2(b_2-l_2)}{b_2+l_1} + \frac{b_1+l_2}{r_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$, $R_2^* = \frac{b_2+l_1}{d_2} < R_0^*$;
- *otherwise,* $R_2^* = R_0^*$.

Basic algebraic manipulation gives that when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, $\frac{d_2(b_2-l_2)}{b_2+l_1} + \frac{b_1+l_2}{r_1} \ge \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}$. Therefore, we have $R_2^* \ge R_1^*$.

Solution for Sub-optimization **1***:* The Lagrangian function for sub-optimization 1 is

$$\begin{aligned} L_1(R, r_2', b_2', \boldsymbol{\lambda}) &= R + \lambda_1(r_1 + r_2' - R) + \lambda_2\left(\frac{b_1 + b_2'}{R - r_2'} - d_1\right) + \lambda_3\left(\frac{b_2 + l_1}{R} - d_2\right) \\ &+ \lambda_4\left(\frac{b_2 - b_2'}{r_2'} + \frac{l_1 + l_2}{R} - d_2\right) + \lambda_5\left(\frac{b_1 + b_2'}{r_1} - \frac{(R - r_2')(b_2 - b_2')}{r_1(r_2' - r_2)}\right) \\ &+ \lambda_6(r_1 + r_2 - R) + \lambda_7(r_2 - r_2') + \lambda_8(l_2 - b_2') + \lambda_9(b_2' - b_2) \end{aligned} \tag{40}$$

$$\nabla_{R,r_2',b_2'} L_1 = \begin{bmatrix} 1 - \lambda_1 - \frac{\lambda_2(b_1+b_2')}{(R-r_2')^2} - \frac{\lambda_3(b_2+l_1)}{R^2} - \frac{\lambda_4(l_1+l_2)}{R^2} - \frac{\lambda_5(b_2-b_2')}{r_1(r_2'-r_2)} - \lambda_6 \\ \lambda_1 + \frac{\lambda_2(b_1+b_2')}{(R-r_2')^2} - \frac{\lambda_4(b_2-b_2')}{r_2'^2} + \frac{\lambda_5(b_2-b_2')(R-r_2)}{r_1(r_2'-r_2)^2} - \lambda_7 \\ \frac{\lambda_2}{R-r_2'} - \frac{\lambda_4}{r_2'} + \lambda_5\left(\frac{1}{r_1} + \frac{R-r_2'}{r_1(r_2'-r_2)}\right) - \lambda_8 + \lambda_9 \end{bmatrix} \tag{41}$$

$$\text{diag}(\triangle_{R,r_2',b_2'} L_1) = \begin{bmatrix} \frac{2\lambda_2(b_1+b_2')}{(R-r_2')^3} + \frac{2\lambda_3(b_2+l_1)}{R^3} + \frac{2\lambda_4(l_1+l_2)}{R^3} \\ \frac{2\lambda_2(b_1+b_2')}{(R-r_2')^3} + \frac{2\lambda_4(b_2-b_2')}{r_2'^3} - \frac{2\lambda_5(b_2-b_2')(R-r_2)}{r_1(r_2'-r_2)^3} \\ 0 \end{bmatrix} \tag{42}$$

From Eq. (34), we know that when $\frac{b_1+b_2'}{r_1} - \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} = 0$, $d_1^* = \frac{b_1+b_2}{R-r_2}$. Combine it with Corollary 20, we have $R_1^* = R_0^*$.

Next we consider the case where $\frac{b_1+b_2'}{r_1} - \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} > 0$. Then from KKT conditions' complementary slackness requirement, we have $\lambda_5 = 0$. Substitute $\lambda_5 = 0$ into Eq. (42), we have

$$\text{diag}(\triangle_{R,r_2',b_2'} L_1) = \begin{bmatrix} \frac{2\lambda_2(b_1+b_2')}{(R-r_2')^3} + \frac{2\lambda_3(b_2+l_1)}{R^3} + \frac{2\lambda_4(l_1+l_2)}{R^3} \\ \frac{2\lambda_2(b_1+b_2')}{(R-r_2')^3} + \frac{2\lambda_4(b_2-b_2')}{r_2'^3} \\ 0 \end{bmatrix} \ge 0 \tag{43}$$

Therefore, for any $(r_2', b_2', \boldsymbol{\lambda})$ satisfying KKT's necessary conditions, it is a local optimum. The necessary conditions for the optimization under $\lambda_5 = 0$ is:

$$
\begin{aligned}
&1-\lambda_1-\frac{\lambda_2(b_1+b_2')}{(R-r_2')^2}-\frac{\lambda_3(b_2+l_1)}{R^2}-\frac{\lambda_4(l_1+l_2)}{R^2}-\lambda_6=0, && \frac{\lambda_2}{R-r_2'}-\frac{\lambda_4}{r_2'}-\lambda_8+\lambda_9=0,\\
&\lambda_1+\frac{\lambda_2(b_1+b_2')}{(R-r_2')^2}-\frac{\lambda_4(b_2-b_2')}{r_2'^2}-\lambda_7=0, && \lambda_i\ge 0,\ \text{for } i=1,...9,\\
&r_1+r_2'-R\le 0, && \lambda_1(r_1+r_2'-R)=0,\\
&\frac{b_1+b_2'}{R-r_2'}-d_1\le 0, && \lambda_2\left(\frac{b_1+b_2'}{R-r_2'}-d_1\right)=0,\\
&\frac{b_2+l_1}{R}-d_2\le 0, && \lambda_3\left(\frac{b_2+l_1}{R}-d_2\right)=0,\\
&\frac{b_2-b_2'}{r_2'}+\frac{l_1+l_2}{R}-d_2\le 0, && \lambda_4\left(\frac{b_2-b_2'}{r_2'}+\frac{l_1+l_2}{R}-d_2\right)=0,\\
&\frac{b_1+b_2'}{r_1}-\frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)}<0, && \lambda_5=0,\\
&r_1+r_2-R\le 0, && \lambda_6(r_1+r_2-R)=0,\\
&r_2-r_2'\le 0, && \lambda_7(r_2-r_2')=0,\\
&l_2-b_2'\le 0, && \lambda_8(l_2-b_2')=0,\\
&b_2'-b_2\le 0 && \lambda_9(b_2'-b_2)=0
\end{aligned}
\tag{44}
$$

For the conditions in Eq. (44), we have:

- $R_1^*=r_1+r_2,$ when $d_1\in[\frac{l_2+b_1}{r_1},\frac{b_1+b_2}{r_1})$ and $d_2\ge\max\left\{\frac{b_2+l_1}{r_1+r_2},\frac{b_1+b_2-d_1r_1}{r_2}+\frac{l_1+l_2}{r_1+r_2}\right\}$;
- $R_1^*=\frac{b_2+l_1}{d_2}\ge r_1+r_2,$ when $d_2<\frac{b_2+l_1}{r_1+r_2}$ and $d_1\in[\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2}+\frac{d_2(b_2-l_2)}{b_2+l_1},\frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$;
- $R_1^*=\frac{l_2+b_1}{d_1}+r_2$, when $d_2<\frac{l_1+b_2}{r_2}$ and $d_1<\min\left\{\frac{b_1+l_2}{r_1},\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}\right\}$; and when $d_2\ge\frac{l_1+b_2}{r_2}$ and $d_1<\frac{b_1+l_2}{r_1}$;
- $R_1^*=\frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1}$,
  - ◦ when $d_2<\frac{b_2+l_1}{r_1+r_2}$, and $d_1\in[\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2},\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2}+\frac{d_2(b_2-l_2)}{b_2+l_1})$; and
  - ◦ when $\frac{b_2+l_1}{r_1+r_2}\le d_2\le\frac{b_2+l_1}{r_2}$, and $d_1\in[\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2},\frac{r_2(l_1+l_2)}{r_1(r_1+r_2)}+\frac{b_1+b_2-d_2r_2}{r_1})$.

Proof. Note that when $b_2'=b_2$, the reprofiler has no effect, *i.e.,* $R^*=R_0^*$. Therefore, we consider only $\lambda_9=0$ and $b_2'<b_2$. Then from $\frac{\lambda_2}{R-r_2'}-\frac{\lambda_4}{r_2'}-\lambda_8+\lambda_9=0$ we have 1) if $\lambda_2=0$, then $\lambda_4=\lambda_8=0$; and 2) if $\lambda_2>0$, then $\lambda_4+\lambda_8>0$.

- When $\lambda_2=0$, $\lambda_4=\lambda_8=0$, from $\lambda_1+\frac{\lambda_2(b_1+b_2')}{(R-r_2')^2}-\frac{\lambda_4(b_2-b_2')}{r_2'^2}-\lambda_7=0$, we have $\lambda_1=\lambda_7$.
  - When $\lambda_1=\lambda_7>0$, it has $R=r_1+r_2'$ and $r_2=r_2'$. Therefore, $R=r_1+r_2$. Then the constraints become:

$$
\begin{cases}
\frac{b_1+b_2'}{r_1}-d_1\le 0 \implies b_2'\le d_1r_1-b_1,\\
\frac{b_2+l_1}{r_1+r_2}-d_2\le 0 \implies d_2\ge\frac{b_2+l_1}{r_1+r_2},\\
\frac{b_2-b_2'}{r_2'}+\frac{l_1+l_2}{r_1+r_2}-d_2\le 0, \implies b_2'\ge b_2+r_2\left(\frac{l_1+l_2}{r_1+r_2}-d_2\right)\\
l_2\le b_2'<b_2
\end{cases}
\tag{45}
$$

Hence we have $b_2'\in[b_2+r_2\left(\frac{l_1+l_2}{r_1+r_2}-d_2\right),d_1r_1-b_1]\cap[l_2,b_2)$. As $d_2\ge\frac{b_2+l_1}{r_1+r_2}>\frac{l_1+l_2}{r_1+r_2}$, we have $b_2-r_2\left(\frac{l_1+l_2}{r_1+r_2}-d_2\right)<b_2$. Therefore, to guarantee $[b_2+r_2\left(\frac{l_1+l_2}{r_1+r_2}-d_2\right),d_1r_1-b_1]\cap[l_2,b_2)\ne\emptyset$:

$$
\begin{cases}
d_1r_1-b_1\ge l_2 \implies d_1\ge\frac{l_2+b_1}{r_1}\\
b_2+r_2\left(\frac{l_1+l_2}{r_1+r_2}-d_2\right)\le d_1r_1-b_1 \implies d_2\ge\frac{b_1+b_2-d_1r_1}{r_2}+\frac{l_1+l_2}{r_1+r_2}
\end{cases}
\tag{46}
$$

Remember that adding a reprofiler is beneficial only when $R_0^*=\frac{b_1+b_2}{d_1}+r_2>\max\left\{r_1+r_2,\frac{b_2+l_1}{d_2}\right\}$, *i.e.,* $d_1<\frac{b_1+b_2}{r_1}$ and $d_2>\frac{b_2+l_1}{\frac{b_1+b_2}{d_1}+r_2}$, which gives $\frac{b_2+l_1}{r_1+r_2}>\frac{b_2+l_1}{\frac{b_1+b_2}{d_1}}$. Therefore, we have

$$
\circ\ d_1\in[\frac{l_2+b_1}{r_1},\frac{b_1+b_2}{r_1})\text{ and } d_2\ge\max\left\{\frac{b_2+l_1}{r_1+r_2},\ \frac{b_1+b_2-d_1r_1}{r_2}+\frac{l_1+l_2}{r_1+r_2}\right\}.
\tag{47}
$$

• When $\lambda_1 = \lambda_7 = 0$, from $1 - \lambda_1 - \frac{\lambda_2(b_1+b_2')}{(R-r_2')^2} - \frac{\lambda_3(b_2+l_1)}{R^2} - \frac{\lambda_4(l_1+l_2)}{R^2} - \lambda_6 = 0$ we have $\lambda_3 + \lambda_6 > 0$. If $\lambda_6 > 0$, it has $r_1 + r_2 - R = 0$. As $r_1 + r_2' - R \le 0$, it has $r_2' = r_2$. Therefore, it produces the same optimization as Eq. (45). Therefore, we only consider $\lambda_6 = 0$ in this case. When $\lambda_6 = 0$, $\lambda_3 > 0$, then we have $R = \frac{b_2+l_1}{d_2}$. Note that $R = \frac{b_2+l_1}{d_2}$ implies $\frac{b_2+l_1}{R} \ge \frac{b_2-b_2'}{r_2'} + \frac{l_1+l_2}{R}$, *i.e.*, $b_2' \ge b_2 - \frac{r_2'(b_2-l_2)}{R}$. Then the constraints become

$$\begin{cases} r_2' \in [r_2, R-r_1], \qquad b_2' \in [l_2, b_2), \qquad r_1 + r_2 \le R, \\ \frac{b_1+b_2'}{R-r_2'} \le d_1 \implies b_2' \le d_1(R-r_2') - b_1, \\ b_2' \ge b_2 - \frac{r_2'(b_2-l_2)}{R}, \\ b_1 + b_2' - \frac{(R-r_2')(b_2-b_2')}{r_2'-r_2} < 0 \implies b_2' < b_2 - \frac{(b_1+b_2)(r_2'-r_2)}{R-r_2}. \end{cases} \tag{48}$$

Note that $b_2 - \frac{(b_1+b_2)(r_2'-r_2)}{R-r_2} - d_1(R-r_2') + b_1 = (R-r_2')\left(\frac{b_1+b_2}{R-r_2} - d_1\right) = \frac{d_1(R-r_2')(R_0^*-R)}{R-r_2}$. As $r_2' < R$, under $R < R_0^*$, it has $b_2 - \frac{(b_1+b_2)(r_2'-r_2)}{R-r_2} > d_1(R-r_2') - b_1$. Hence, we have $b_2' \in [b_2 - \frac{r_2'(b_2-l_2)}{R}, d_1(R-r_2') - b_1]$. Next we configure the conditions where $\exists r_2' \in [r_2, R-r_1]$, such that $[b_2 - \frac{r_2'(b_2-l_2)}{R}, d_1(R-r_2') - b_1] \cap [l_2, b_2) \ne \emptyset$.
For $[b_2 - \frac{r_2'(b_2-l_2)}{R}, d_1(R-r_2') - b_1] \cap [l_2, b_2) \ne \emptyset$, it requires

$$\begin{cases} b_2 - \frac{r_2'(b_2-l_2)}{R} < b_2 \implies b_2 > l_2, \\ d_1(R-r_2') - b_1 \ge l_2 \implies r_2' \le R - \frac{b_1+l_2}{d_1}, \\ b_2 - \frac{r_2'(b_2-l_2)}{R} - d_1(R-r_2') + b_1 \le 0 \implies r_2' \le R - \frac{b_1+l_2}{d_1-(b_2-l_2)/R} \end{cases} \tag{49}$$

Basic algebraic manipulation gives $R - \frac{b_1+l_2}{d_1} > R - \frac{b_1+l_2}{d_1-(b_2-l_2)/R}$. Hence, Eq. (49) gives $r_2' \le R - \frac{b_1+l_2}{d_1-(b_2-l_2)/R}$. Combine it with $r_2' \in [r_2, R-r_1]$ and $R = \frac{b_2+l_1}{d_2}$, we have

$$r_2 \le R - \frac{b_1+l_2}{d_1-(b_2-l_2)/R} \implies d_1 \ge \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{(b_2-l_2)d_2}{b_2+l_1}, \tag{50}$$

Also, from $r_1 + r_2 < R < \frac{b_1+b_2}{d_1} + r_2$, we have $d_2 < \frac{b_2+l_1}{r_1+r_2}$ and $d_1 < \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2}$. Hence, we have

$$\circ\ d_2 < \frac{b_2+l_1}{r_1+r_2}, \text{ and } d_1 \in [\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{(b_2-l_2)d_2}{b_2+l_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2}). \tag{51}$$

Basic algebraic manipulation shows that the interval is always valid.

- When $\lambda_2 > 0$, as $\lambda_9 = 0$, from $\frac{\lambda_2}{R-r_2'} - \frac{\lambda_4}{r_2'} - \lambda_8 + \lambda_9 = 0$ we have $\frac{\lambda_2}{R-r_2'} = \frac{\lambda_4}{r_2'} + \lambda_8$ and $\lambda_4 + \lambda_8 > 0$. Combining it with $\lambda_1 + \frac{\lambda_2(b_1+b_2')}{(R-r_2')^2} - \frac{\lambda_4(b_2-b_2')}{r_2'^2} - \lambda_7$ and $\frac{b_1+b_2'}{R-r_2'} = d_1$. We have $\lambda_8 d_1 + \frac{\lambda_4}{r_2'}\left(d_1 - \frac{b_2-b_2'}{r_2'}\right) + \lambda_1 - \lambda_7 = 0$. As $\frac{b_2-b_2'}{r_2'} + \frac{l_1+l_2}{R} \le d_2$, $\lambda_8 d_1 + \frac{\lambda_4}{r_2'}\left(d_1 - \frac{b_2-b_2'}{r_2'}\right) + \lambda_1 - \lambda_7 \ge d_1\lambda_8 + \lambda_1 - \lambda_7 + \frac{\lambda_4}{r_2'}\left(d_1 - d_2 + \frac{l_1+l_2}{R}\right) = 0$. Therefore, given $\lambda_4 + \lambda_8 > 0$, we have $\lambda_7 > 0$, *i.e.*, $r_2' = r_2$.

• When $\lambda_8 = 0$, $\lambda_4 > 0$, *i.e.*, $\frac{b_2-b_2'}{r_2} + \frac{l_1+l_2}{R} = d_2$. Then the constraints become

$$\begin{cases} r_1 + r_2 \le R, \qquad b_2' \in [l_2, b_2), \\ \frac{b_1+b_2'}{R-r_2} = d_1 \implies R = \frac{b_1+b_2'}{d_1} + r_2, \\ \frac{b_2-b_2'}{r_2} + \frac{l_1+l_2}{R} = d_2, \\ R \ge \frac{b_2+l_1}{d_2}. \end{cases} \tag{52}$$

Substituting $R = \frac{b_1+b_2'}{d_1} + r_2$ into $\frac{b_2-b_2'}{r_2} + \frac{l_1+l_2}{R} = d_2$ gives

$$\frac{d_1}{r_2}R^2 - \left[d_1 - d_2 + \frac{b_1+b_2}{r_2}\right]R - (l_1+l_2) = 0,$$

which gives

$$R = \frac{(d_1-d_2)r_2 + (b_1+b_2) + \sqrt{((d_1-d_2)r_2 + b_1 + b_2)^2 + 4d_1r_2(l_1+l_2)}}{2d_1},$$

and

$$b_2' = \frac{(b_2 - b_1) - (d_1 + d_2)r_2 + \sqrt{((d_1 - d_2)r_2 + b_1 + b_2)^2 + 4d_1 r_2 (l_1 + l_2)}}{2}.$$

$b_2' \in [l_2, b_2)$ gives

$$d_2 \in \left(\frac{d_1(l_1 + l_2)}{b_1 + b_2 + d_1 r_2},\ \frac{d_1(l_1 + l_2)}{b_1 + l_2 + d_1 r_2} + \frac{b_2 - l_2}{r_2} \right].$$

Note that when $R_0^* = \frac{b_1+b_2}{d_1} + r_2$, we have $d_2 > \frac{(b_2+l_1)d_1}{b_1+b_2+r_2 d_1} \in (\frac{d_1(l_1+l_2)}{b_1+b_2+d_1 r_2}, \frac{d_1(l_1+l_2)}{b_1+l_2+d_1 r_2} + \frac{b_2-l_2}{r_2})$. Hence we have $d_2 \in (\frac{(b_2+l_1)d_1}{b_1+b_2+r_2 d_1}, \frac{d_1(l_1+l_2)}{b_1+l_2+d_1 r_2} + \frac{b_2-l_2}{r_2}]$, *i.e.*,

$$d_2 < \frac{b_2 + l_1}{r_2}, \text{ and } d_1 \in [\frac{(b_1 + l_2)\left(d_2 - \frac{b_2 - l_2}{r_2}\right)}{b_2 + l_1 - r_2 d_2}, \frac{d_2(b_1 + b_2)}{b_2 + l_1 - r_2 d_2}). \tag{53}$$

From $R \geq r_1 + r_2$ and $R \geq \frac{b_2+l_1}{d_2}$, we have: when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, $d_1 \leq \frac{r_2(l_1+l_2)}{\frac{b_2+l_1}{d_2}\left(\frac{b_2+l_1}{d_2} - r_2\right)} + \frac{b_1+b_2-d_2 r_2}{\frac{b_2+l_1}{d_2} - r_2} = \frac{d_2(b_1+l_2)}{b_2+l_1-d_2 r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}$, which is greater than $\frac{d_2(b_1+b_2)}{b_2+l_1-r_2 d_2}$; and when $d_2 \geq \frac{b_2+l_1}{r_1+r_2}$, $d_1 \leq \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2 r_2}{r_1} < \frac{d_2(b_1+b_2)}{b_2+l_1-r_2 d_2}$. Combining it with Eq. (53) gives:

- ∘ when $d_2 < \frac{l_1+b_2}{r_2+r_1}$, $d_1 \in [\frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2 d_2}, \frac{d_2(b_1+b_2)}{b_2+l_1-r_2 d_2})$;
- ∘ when $\frac{l_1+b_2}{r_1+r_2} \leq d_2 \leq \frac{b_2+l_1}{r_2}$, $d_1 \in [\frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2 d_2}, \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2 r_2}{r_1})$.

- When $\lambda_8 > 0$, *i.e.*, $b_2' = l_2$, we have $\frac{b_2-b_2'}{r_2} + \frac{l_1+l_2}{R} = \frac{l_1+b_2}{R} + \left(\frac{1}{r_2} - \frac{1}{R}\right)(b_2 - l_2) > \frac{b_2+l_1}{R}$. Then the constraints become

$$\begin{cases} \frac{b_1+l_2}{R-r_2} = d_1 \implies R = \frac{b_1+l_2}{d_1} + r_2 < \frac{b_2+b_1}{d_1} + r_2 = R_0^*, \\ R > r_1 + r_2 \implies d_1 < \frac{b_1+l_2}{r_1} \\ \frac{b_2-l_2}{r_2} + \frac{l_1+l_2}{R} \leq d_2 \implies d_2 \geq \frac{b_2-l_2}{r_2} + \frac{(l_1+l_2)d_1}{b_1+l_2+r_2 d_1} > \frac{(b_2+l_1)d_1}{b_1+b_2+r_2 d_1 s} \end{cases} \tag{54}$$

Hence, we have $d_1 < \frac{b_1+l_2}{r_1}$, and $d_2 \geq \frac{b_2-l_2}{r_2} + \frac{(l_1+l_2)d_1}{b_1+l_2+r_2 d_1}$. Basic algebraic manipulation gives that it is equivalent to:

- ∘ when $d_2 \geq \frac{l_1+b_2}{r_2}$, $d_1 < \frac{b_1+l_2}{r_1}$;
- ∘ when $d_2 < \frac{l_1+b_2}{r_2}$, $d_1 \leq \min\left\{\frac{b_1+l_2}{r_1}, \frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2 d_2}\right\}$.

In summary, the local optimums are:

- $R = r_1 + r_2$, when $d_1 \in [\frac{l_2+b_1}{r_1}, \frac{b_1+b_2}{r_1})$ and $d_2 \in [\max\left\{\frac{b_2+l_1}{r_1+r_2}, \frac{b_1+b_2-d_1 r_1}{r_2} + \frac{l_1+l_2}{r_1+r_2}\right\}, d_1)$;
- $R = \frac{b_2+l_1}{d_2} \geq r_1 + r_2$, when $d_2 < \frac{b_2+l_1}{r_1+r_2}$ and $d_1 \in [\frac{d_2(b_1+l_2)}{b_2+l_1-d_2 r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2 r_2})$;
- $R = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1 r_2(l_1+l_2)}}{2d_1} \geq \max\left\{r_1 + r_2, \frac{b_2+l_1}{d_2}, \frac{b_1+l_2}{d_1} + r_2\right\}$,
  - ∘ when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2 d_2}, \frac{d_2(b_1+b_2)}{b_2+l_1-r_2 d_2})$;
  - ∘ when $\frac{b_2+l_1}{r_1+r_2} \leq d_2 \leq \frac{b_2+l_1}{r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2 d_2}, \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2 r_2}{r_1})$;
- $R = \frac{b_1+l_2}{d_1} + r_2 > \max\left\{r_1 + r_2, \frac{b_2+l_1}{d_2}\right\}$,
  - ∘ when $d_2 < \frac{l_1+b_2}{r_2}$ and $d_1 < \min\left\{\frac{b_1+l_2}{r_1}, \frac{(b_1+l_2)\left(d_2 - \frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2 d_2}\right\}$;
  - ∘ when $d_2 \geq \frac{l_1+b_2}{r_2}$ and $d_1 < \frac{b_1+l_2}{r_1}$.
- $R = R_0^*$, otherwise.

Next we characterize the global optimum R_1^*. We considering three cases: $d_2 \geq \frac{b_2+l_1}{r_2}$, $\frac{b_2+l_1}{r_1+r_2} < d_2 < \frac{b_2+l_1}{r_2}$, and $d_2 \leq \frac{b_2+l_1}{r_1+r_2}$.

- When $d_2 \geq \frac{b_2+l_1}{r_2}$, we consider whether $\frac{b_1+b_2}{r_1+r_2} + \frac{(l_1+l_2)r_2}{(r_1+r_2)^2} - \frac{l_2+b_1}{r_1} \geq 0$ or not.
  - ∘ When $\frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2} - \frac{l_2+b_1}{r_1} \geq 0$, *i.e.*, $r_1 r_2 (r_1 + r_2)\left(\frac{b_2-l_2}{r_2} - \frac{b_1-l_1}{r_1}\right) \geq r_2^2(l_1 + l_2)$, basic algebraic manipulation gives that for all $d_2 \geq \frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2}$, $R = r_1 + r_2$. As $\frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2} - \frac{l_1+b_2}{r_2} = \frac{r_1 r_2}{r_2(r_2+r_2)}\left(\frac{b_1-l_1}{r_1} - \frac{b_2-l_2}{r_2}\right) + \frac{(r_2^2-r_1^2-r_1 r_2)(l_1+l_2)}{r_2(r_1+r_2)^2} \leq -r_1(r_1 + r_2)(l_1 + l_2) < 0$, we have $\frac{l_2+b_1}{r_2} > \frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2}$. Therefore, we have $R_1^* = r_1 + r_2$.
  - ∘ When $\frac{b_1+b_2}{r_1+r_2} + \frac{(l_1+l_2)r_2}{(r_1+r_2)^2} - \frac{l_2+b_1}{r_1} < 0$, from the local optimums we have:
    - ∘ when $\frac{l_2+b_1}{r_1} \leq \frac{l_1+b_2}{r_2}$, $R_1^* = r_1 + r_2$;
    - ∘ when $\frac{l_2+b_1}{r_1} > \frac{l_1+b_2}{r_2}$, $R_1^* = r_1 + r_2$ when $d_1 \geq \frac{l_2+b_1}{r_1}$, while $R_1^* = \frac{b_1+l_2}{d_1} + r_2$ when $d_1 < \frac{l_2+b_1}{r_1}$.

- When $\frac{b_2+l_1}{r_1+r_2} < d_2 < \frac{b_2+l_1}{r_2}$, we separate it into three conditions: $d_1 \geq \frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2}$, $d_1 \in (\frac{b_1+b_2-d_2r_2}{r_1} + \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)}, \frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2})$, and $d_1 < \frac{b_1+b_2-d_2r_2}{r_1} + \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)}$.
  - ○ When $d_1 \geq \frac{b_1+b_2}{r_1+r_2} + \frac{r_2(l_1+l_2)}{(r_1+r_2)^2}$, we have $\frac{b_2+l_1}{r_1+r_2} > \frac{b_1+b_2-d_1r_1}{r_2} + \frac{l_1+l_2}{r_1+r_2} < \frac{b_2+l_1}{r_1+r_2} < d_2$. Therefore, we have $R_1^* = r_1 + r_2$.
  - ○ When $d_1 \in (\frac{b_1+b_2-d_2r_2}{r_1}+\frac{r_2(l_1+l_2)}{r_1(r_1+r_2)}, \frac{b_1+b_2}{r_1+r_2}+\frac{r_2(l_1+l_2)}{(r_1+r_2)^2})$, basic algebraic manipulation gives $d_1 > \frac{b_1+b_2-d_2r_2}{r_1}+\frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} > \frac{b_2+l_1}{r_1+r_2}$. Therefore, we have $R_1^* = r_1 + r_2$.
  - ○ When $d_1 < \frac{b_1+b_2-d_2r_2}{r_1} + \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)}$, from the local optimums we directly have:
    - ○ when $d_2 \geq \frac{l_1+b_2}{r_1+r_2} + \frac{r_1(b_2-l_2)}{r_2(r_1+r_2)}$, $R_1^* = \frac{b_1+l_2}{d_1} + r_2$.
    - ○ when $d_2 < \frac{l_1+b_2}{r_1+r_2}+\frac{r_1(b_2-l_2)}{r_2(r_1+r_2)}$, $R_1^* = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1}$ when $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{b_1+b_2}{r}$ $\frac{r_2(l_1+l_2)}{r_1(r_1+r_2)})$, while $R_1^* = \frac{b_1+l_2}{d_1} + r_2$ when $d_1 < \frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}$.
- When $d_2 \leq \frac{b_2+l_1}{r_1+r_2}$, we have $\frac{b_1+l_2}{r_1} > \frac{(b_2+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}$. Therefore, from local optimums we directly have:
  - ○ when $d_1 \in (d_2, \frac{(b_2+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}]$, $R_1^* = \frac{b_1+l_2}{d_1} + r_2$;
  - ○ when $d_1 \in [\frac{(b_2+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}, \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1})$,
    $R_1^* = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1}$;
  - ○ when $d_1 \in (\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}), \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$, $R_1^* = \frac{b_2+l_1}{d_2}$.

Remark: The above analysis also shows: $R_1^* > R_0^*$ as long as $R_0^* = \frac{b_1+b_2}{d_1} + r_2 > \max\left\{\frac{b_2+l_1}{d_2}, r_1 + r_2\right\}$.

In summary, the global optimum is:

- $R_1^* = r_1 + r_2$, when $d_1 \in [\frac{l_2+b_1}{r_1}, \frac{b_1+b_2}{r_1})$ and $d_2 \geq \max\left\{\frac{b_2+l_1}{r_1+r_2}, \frac{b_1+b_2-d_1r_1}{r_2} + \frac{l_1+l_2}{r_1+r_2}\right\}$;
- $R_1^* = \frac{b_2+l_1}{d_2} \geq r_1 + r_2$, when $d_2 < \frac{b_2+l_1}{r_1+r_2}$ and $d_1 \in [\frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$;
- $R_1^* = \frac{l_2+b_1}{d_1} + r_2$, when $d_2 < \frac{l_1+b_2}{r_2}$ and $d_1 < \min\left\{\frac{b_1+l_2}{r_1}, \frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{l_1+b_2-r_2d_2}\right\}$; and when $d_2 \geq \frac{l_1+b_2}{r_2}$ and $d_1 < \frac{b_1+l_2}{r_1}$;
- $R_1^* = \frac{(d_1-d_2)r_2+(b_1+b_2)+\sqrt{((d_1-d_2)r_2+b_1+b_2)^2+4d_1r_2(l_1+l_2)}}{2d_1}$,
  - ○ when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{d_2(b_1+l_2)}{b_2+l_1-d_2r_2} + \frac{d_2(b_2-l_2)}{b_2+l_1})$; and
  - ○ when $\frac{b_2+l_1}{r_1+r_2} \leq d_2 \leq \frac{b_2+l_1}{r_2}$, and $d_1 \in [\frac{(b_1+l_2)\left(d_2-\frac{b_2-l_2}{r_2}\right)}{b_2+l_1-r_2d_2}, \frac{r_2(l_1+l_2)}{r_1(r_1+r_2)} + \frac{b_1+b_2-d_2r_2}{r_1})$.

□

Solution for Sub-optimization **2***:* The Lagrangian function for sub-optimization 2 is

$$\begin{aligned} L_2(R, r_2', b_2', \boldsymbol{\lambda}) = {} & R + \lambda_1(R - r_1 - r_2') + \lambda_2\left(\frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)} - d_1\right) \\ & + \lambda_3\left(\frac{b_2+l_1}{R} - d_2\right) + \lambda_4\left(\frac{b_2-b_2'}{r_2'} + \frac{l_1+l_2}{R} - d_2\right) \\ & + \lambda_5\left(\frac{b_1+b_2'}{r_1} - \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)}\right) + \lambda_6(r_1+r_2-R) \\ & + \lambda_7(r_2-r_2') + \lambda_8(l_2-b_2') + \lambda_9(b_2'-b_2) \end{aligned} \tag{55}$$

$$\nabla_{R,r_2',b_2'} L_2 = \begin{bmatrix} 1 + \lambda_1 - \frac{\lambda_2(b_2-b_2')}{r_1(r_2'-r_2)^2} - \frac{\lambda_3(b_2+l_1)}{R^2} - \frac{\lambda_4(l_1+l_2)}{R^2} - \frac{\lambda_5(b_2-b_2')}{r_1(r_2'-r_2)} - \lambda_6 \\ -\lambda_1 + \frac{\lambda_2(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)^2} - \frac{\lambda_4(b_2-b_2')}{r_2'^2} + \frac{\lambda_5(b_2-b_2')(R-r_2)}{r_1(r_2'-r_2)^2} - \lambda_7 \\ \frac{\lambda_2(R-r_1-r_2)}{r_1(r_2'-r_2)} - \frac{\lambda_4}{r_2'} + \lambda_5\left(\frac{1}{r_1} + \frac{R-r_2'}{r_1(r_2'-r_2)}\right) - \lambda_8 + \lambda_9 \end{bmatrix} \tag{56}$$

$$\mathrm{diag}(\triangle_{R,r_2',b_2'} L_1) = \begin{bmatrix} \frac{2\lambda_3(b_2+l_1)}{R^3} + \frac{2\lambda_4(l_1+l_2)}{R^3} \\ -\frac{2\lambda_2(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)^3} + \frac{2\lambda_4(b_2-b_2')}{r_2'^3} - \frac{2\lambda_5(b_2-b_2')(R-r_2)}{r_1(r_2'-r_2)^3} \\ 0 \end{bmatrix} \tag{57}$$

Similar as that for suboptimization 1, we have $\lambda_5^* = 0$. Substitute it into Eq. (57), we have

$$\mathrm{diag}(\triangle_{R,r_2',b_2'} L_1) = \begin{bmatrix} \frac{2\lambda_3(b_2+l_1)}{R^3} + \frac{2\lambda_4(l_1+l_2)}{R^3} \\ \\ -\frac{2\lambda_2(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)^3} + \frac{2\lambda_4(b_2-b_2')}{r_2'^3} \\ \\ 0 \end{bmatrix} \tag{58}$$

Next we consider KKT's necessary conditions for the optimization under $\lambda_5^* = 0$. Remember that when solving sub-optimization 2, we only consider $R < r_1 + r_2'$. As before, we consider only $b_2' < b_2$. Therefore, we have:

$$\begin{aligned}
&1 + \lambda_1 - \frac{\lambda_2(b_2 - b_2')}{r_1(r_2' - r_2)^2} - \frac{\lambda_3(b_2 + l_1)}{R^2} - \frac{\lambda_4(l_1 + l_2)}{R^2} - \lambda_6 = 0, && \frac{\lambda_2(R - r_1 - r_2)}{r_1(r_2' - r_2)} - \frac{\lambda_4}{r_2'} - \lambda_8 + \lambda_9 = 0, \\
&\frac{\lambda_2(R - r_1 - r_2)(b_2 - b_2')}{r_1(r_2' - r_2)^2} - \frac{\lambda_4(b_2 - b_2')}{r_2'^2} = \lambda_1 + \lambda_7, && \lambda_i \geq 0, \text{ for } i = 1, ...9, \\
&R - r_1 - r_2' < 0, && \lambda_1 = 0, \\
&\frac{b_1 + b_2}{r_1} - \frac{(R - r_1 - r_2)(b_2 - b_2')}{r_1(r_2' - r_2)} - d_1 \leq 0, && \lambda_2 \left(\frac{b_1 + b_2}{r_1} - \frac{(R - r_1 - r_2)(b_2 - b_2')}{r_1(r_2' - r_2)} - d_1 \right) = 0, \\
&\frac{b_2 + l_1}{R} - d_2 \leq 0, && \lambda_3 \left(\frac{b_2 + l_1}{R} - d_2 \right) = 0, \\
&\frac{b_2 - b_2'}{r_2'} + \frac{l_1 + l_2}{R} - d_2 \leq 0, && \lambda_4 \left(\frac{b_2 - b_2'}{r_2'} + \frac{l_1 + l_2}{R} - d_2 \right) = 0, \\
&\frac{b_1 + b_2'}{r_1} - \frac{(R - r_2')(b_2 - b_2')}{r_1(r_2' - r_2)} < 0, && \lambda_5 = 0, \\
&r_1 + r_2 - R \leq 0, && \lambda_6(r_1 + r_2 - R) = 0, \\
&r_2 - r_2' \leq 0, && \lambda_7(r_2 - r_2') = 0, \\
&l_2 - b_2' \leq 0, && \lambda_8(l_2 - b_2') = 0, \\
&b_2' - b_2 < 0 && \lambda_9 = 0
\end{aligned} \tag{59}$$

From the conditions in Eq. (59), we have:

- when $d_2 < \frac{b_2+l_1}{r_1+r_2}$, and $d_1 \in [\frac{d_2(b_2-l_2)}{b_2+l_1} + \frac{b_1+l_2}{r_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2})$, $R_2^* = \frac{b_2+l_1}{d_2} < R_0^*$;
- otherwise, $R_2^* = R_0^*$.

Proof. When $R = r_1 + r_2$, from $\frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)} - d_1 \leq 0$ we have $\frac{b_1+b_2}{d_1} \leq r_1$, which gives $\frac{b_1+b_2}{d_1} + r_2 \leq r_1 + r_2$. From Corollary 20, we know that adding reprofilers cannot decrease the minimum required bandwidth. Therefore, we consider only $R > r_1 + r_2$ afterwards. Combine $R > r_1 + r_2$ with $R < r_1 + r_2'$, we have $r_2' > r_2$. Hence, $\lambda_6 = \lambda_7 = 0$.

As $\lambda_1 = 0$, we have $\frac{\lambda_2(R-r_1-r_2)(b_2-b'-2)}{r_1(r_2'-r_2)^2} - \frac{\lambda_4(b_2-b_2')}{\lambda_2'^2} = 0$. Substitute it to $\frac{\lambda_2(R-r_1-r_2)}{r_1(r_2'-r_2)} - \frac{\lambda_4}{r_2'} - \lambda_8 + \lambda_9 = 0$, we have $-\frac{\lambda_4 r_2}{r_2'^2} - \lambda_8 + \lambda_9 = 0$. As $\lambda_9 = 0$, *i.e.,* $b_2' \neq b_2$, we have $\lambda_2 = \lambda_4 = \lambda_8 = 0$. As $\lambda_1 = \lambda_6 = 0$, from $1 + \lambda_1 - \frac{\lambda_2(b_2-b_2')}{r_1(r_2'-r_2)^2} - \frac{\lambda_3(b_2+l_1)}{R^2} - \frac{\lambda_4(l_1+l_2)}{R^2} - \lambda_6 = 0$, we have $\lambda_3 > 0$, *i.e.,* $R = \frac{b_2+l_1}{d_2}$. Note that when $\lambda_2 = \lambda_4 = 0$ and $\lambda_3 > 0$, from Eq. (57), we have $\mathrm{diag}(\triangle_{R,r_2',b_2'} L_1) \geq 0$. Hence, $R = \frac{b_2+l_1}{d_2}$ is a local optimum.

Hence, the constraints of the optimization reduce to

$$\begin{cases} r_2' \in (R - r_1, R), \qquad b_2' \in [l_2, b_2) \\ \\ \frac{b_1+b_2}{r_1} - \frac{(R-r_1-r_2)(b_2-b_2')}{r_1(r_2'-r_2)} \leq d_1 \implies \frac{b_2-b_2'}{r_2'-r_2} \geq \frac{b_1+b_2-d_1r_1}{R-r_1-r_2}, \\ \\ \frac{b_2-b_2'}{r_2'} + \frac{l_1+l_2}{R} \leq d_2 = \frac{b_2+l_1}{R} \implies b_2 - b_2' \leq \frac{r_2'(b_2-l_2)}{R}, \\ \\ \frac{b_1+b_2'}{r_1} - \frac{(R-r_2')(b_2-b_2')}{r_1(r_2'-r_2)} < 0 \implies \frac{b_2-b_2'}{r_2'-r_2} > \frac{b_1+b_2}{R-r_2}, \end{cases} \tag{60}$$

Basic algebraic manipulation gives $\frac{b_1+b_2}{R-r_2} > \frac{b_1+b_2-d_1r_1}{R-r_1-r_2}$ iff $R > \frac{b_1+b_2}{d_1} + r_2 = R_0^*$. Hence, we only consider the case where $\frac{b_1+b_2}{R-r_2} < \frac{b_1+b_2-d_1r_1}{R-r_1-r_2}$. Under such a condition, we have $b_2 - b_2' \in [\frac{(b_1+b_2-d_1r_1)(r_2'-r_2)}{R-r_1-r_2}, \frac{r_2'(b_2-l_2)}{R}]$.

Next we configure the conditions where:

$$\exists\ r_2' \in (R-r_1, R),\ \text{ s.t. } S := [\frac{(b_1+b_2-d_1r_1)(r_2'-r_2)}{R-r_1-r_2}, \frac{r_2'(b_2-l_2)}{R}] \cap (0, b_2-l_2] \neq \emptyset.$$

When $R^* < R_0^*$, it has $\frac{b_1+b_2}{d_1} + r_2 > r_1 + r_2$, *i.e.*, $b_1 + b_2 - d_1r_1 > 0$. Hence we have $\frac{(b_1+b_2-d_1r_1)(r_2'-r_2)}{R-r_1-r_2} > 0$. From $r_2' < R$, we have $\frac{r_2'(b_2-l_2)}{R} < b_2 - l_2$. Therefore, $S \neq \emptyset$ is equivalent to

$$\exists\ r_2' \in (R-r_1, R) \text{ s.t. } \frac{(b_1+b_2-d_1r_1)(r_2'-r_2)}{R-r_1-r_2} \leq \frac{r_2'(b_2-l_2)}{R}.$$

Define

$$g(x) = \frac{x(b_2-l_2)}{R} - \frac{(b_1+b_2-d_1r_1)(x-r_2)}{R-r_1-r_2}.$$

As $g(x)$ is linear w.r.t x, $S \neq \emptyset$ is equivalent to at least one of $g(R)$ and $g(R-r_1)$ is non-negative, which gives $d_1 > \frac{b_1+l_2}{r_1}$ and $\frac{b_2+l_1}{d_2} > \frac{r_1(b_2-l_2)}{d_1r_1-b_1-l_2}$ through basic algebraic manipulations. Combine it with $\frac{b_1+b_2}{d_1} + r_2 > \frac{b_2+l_1}{d_2} \geq r_1 + r_2$, we have:

$$\circ\ d_2 < \frac{b_2+l_1}{r_1+r_2},\ \text{ and } d_1 \in [\frac{d_2(b_2-l_2)}{b_2+l_1} + \frac{b_1+l_2}{r_1}, \frac{d_2(b_1+b_2)}{b_2+l_1-d_2r_2}).$$

□