Optimal Balancing of Data Transfer Time and Cost in High-Performance Networks

1[st] Liudong Zuo
Computer Science Department
California State University, Dominguez Hills
Carson, CA 90747, USA
Email: lzuo@csudh.edu

2[nd] Daqing Yun
Computer and Information Sciences Program
Harrisburg University of Science and Technology
Harrisburg, PA 17101, USA
Email: dyun@harrisburgu.edu

***Abstract*—The proliferation of large-scale applications has led to the generation of vast datasets across diverse scientific domains. The subsequent need to transfer such expansive data across geographical distances is essential for collaborative data storage and analysis. While reserving bandwidth on dedicated links within high-performance networks (HPNs) has proved as an efficient means for such extensive data transfers, certain crucial challenges remain to be investigated. In this paper, we delve into the intricate trade-off between cost and completion time of data transfers using bandwidth reservation on fixed paths with fixed bandwidth of the HPNs, the most common type of bandwidth reservation or data transfer paths. Our focus centers on the scheduling of two types of bandwidth reservation requests (BRRs) that encapsulate this trade-off: (i) minimizing data transfer cost within prescribed deadlines, and (ii) achieving the earliest data transfer completion time while adhering to predefined cost constraints. We propose two algorithms to optimize the scheduling of individual BRRs of these two types. We then compare the proposed algorithms with existing ones from the perspective of different performance metrics, and efficacy of the proposed algorithms is verified through extensive simulations.**

***Index Terms*—Bandwidth reservation, bandwidth scheduling, resource allocation, data routing, dynamic provisioning, QoS**

I. Introduction

A deluge of data is emanating from diverse scientific domains spanning particle physics, earth science, and social simulations, driven by the pervasive adoption of large-scale applications. After generation, these monumental volumes of data are often needed to be transferred from the generation center to geographically distributed counterparts, each equipped with robust processing and storage capabilities to facilitate collaborative analysis [1], [2]. The swift, reliable, and cost-effective transfer of such substantial datasets is pivotal to expediting the data processing and knowledge extraction inherent to scientific inquiry. Fortuitously, the reservation of bandwidth along dedicated links within high-performance networks (HPNs) has emerged as a potent solution to accommodate these demanding data transfers [3], [4]. This confluence of technological advances holds promise in facilitating seamless data transfers, fostering scientific exploration and discovery.

Over recent years, significant strides have been made in addressing issues pertaining to data transfers utilizing bandwidth reservation services. Nevertheless, certain critical challenges persist, warranting further investigation. A common scenario emerges where diverse users harbor distinct data transfer requisites. For instance, some users prioritize expeditious completion of their transfers, aiming to attain the data transfer's earliest completion time (ECT). Conversely, others seek to minimize costs charged by bandwidth reservation service providers. Resolving bandwidth reservations to meet single data transfer objectives, such as ECT or minimal cost, is generally a straightforward task. However, complexities arise when users stipulate bandwidth reservations that fulfill one data transfer performance criterion while adhering to another. This interplay between various data transfer performance parameters can pose a formidable challenge, demanding a delicate equilibrium to be struck.

This paper delves into the aforementioned challenge, with a primary focus on navigating the intricate balance between cost and a quintessential performance metric, ECT, of data transfers. Specifically, we investigate this dynamic interplay within the context of bandwidth reservation along fixed paths with fixed bandwidth (FPFB) in dedicated HPNs, the most common type of bandwidth reservation or data transfer paths. Our examination centers around two distinct types of bandwidth reservation requests (BRRs), each tailored to address this trade-off: (i) BRR-MinC, aimed at achieving minimal data transfer cost within stipulated deadline, and (ii) BRR-MinT, focused on securing the data transfer ECT while adhering to designated cost constraint. We present two bandwidth reservation algorithms, Opt-MinC and Opt-MinT, to optimize the scheduling of individual BRRs of these two types. A comprehensive assessment ensues, wherein our proposed algorithms are juxtaposed against existing counterparts tailored for analogous challenges, from diverse performance metric perspectives. The efficacy of Opt-MinC and Opt-MinT is duly substantiated through a battery of extensive simulations.

The rest of this paper is organized as follows. We briefly describe the work related to bandwidth reservation from recent years in Section II. The bandwidth reservation concepts and data transfer cost models are presented in Section III. The detailed algorithm designs and analysis of Opt-MinC and Opt-MinT are given in Section IV. We conduct the extensive performance simulation and data analysis in Section V, and then conclude our work in Section VI.

II. Related Work

In recent years, numerous inquiries into bandwidth reservation challenges have yielded substantial research. A succinct overview of the pertinent literature is presented below. Note that different sources may offer varying definitions of BRRs.

Balman [3] delved into the scheduling of individual BRR and considered two data transfer performance metrics: ECT and the shortest data transfer duration. Optimal algorithms were proposed. Users may request diverse path and bandwidth combinations for their data transfers. A comprehensive exploration of these combinations yields four advance bandwidth scheduling problems, all oriented towards achieving the ECT of the data transfer [1]: (i) fixed path with fixed bandwidth, (ii) fixed path with variable bandwidth, (iii) variable path with fixed bandwidth, and (iv) variable path with variable bandwidth. Lin and Wu assessed the complexity of each problem and proposed corresponding optimal or heuristic algorithms.

Zuo *et al.* [5] studied the scheduling of multiple BRRs within batches, and tackled two maximization problems: (i) the maximization of data transfer volume and (ii) the maximization of scheduled BRR count. Both problems were proven to be NP-complete, and corresponding heuristic algorithms were proposed. Yang *et al.* [6] introduced TeaVisor for software-defined networking-based network virtualization. It employs a combination of path virtualization, bandwidth reservation, and path establishment via multi-path routing, and effectively ensures both minimum and maximum bandwidth guarantees while maintaining integrity of tenant routing configurations.

Al–khatib *et al.* [7] investigated the utility of bandwidth reservation in safety-critical vehicular scenarios. Their investigation was specifically tailored to minimize the cumulative reservation cost across diverse problem scenarios, encompassing exact-booking, under-booking, and over-booking scenarios. The presented algorithm dynamically optimizes the allocation of bandwidth resources within these dynamic vehicular environments. On a different note, Zhang *et al.* [8] introduced a novel approach for elastic bandwidth reservation, which effectively mitigates congestion by harnessing the advantages of both fixed and dynamic reservation strategies. Furthermore, they introduced a traffic monitoring framework characterized by minimal monitoring delays and costs, and the dynamic traffic control algorithm effectively balances resource utilization and congestion prevention.

Our prior work [9] explored analogous problems to those studied in this paper, but with two assumptions: (i) the earliest feasible data transfer start time and the data transfer deadline are both time dots of the HPN (the concept of time dots will be elucidated in the forthcoming section), and (ii) the duration of bandwidth reservation aligns with the duration of data transfer. These assumptions facilitated problem-solving but restricted the applicability of the proposed algorithms. In reality, bandwidth reservation entails allocating resources for entire time slots, even if they are only required for fractions of those slots [10]. This paper undertakes a fresh investigation of the problems, unburdened by the aforementioned assumptions.

III. Bandwidth Reservation Concepts and Data Transfer Cost Model

In this section, we present the bandwidth reservation concepts followed by the data transfer cost model.

A. Bandwidth Reservation Concepts

We define the HPN offering bandwidth reservation service within time interval $[T_S, T_E]$ as a weighted graph denoted by $G(V, E)$, where V and E stand for the sets of nodes and edges, respectively [1]. The two types of BRRs, BRR-MinC and BRR-MinT, are elucidated as follows:

- $(v_s, v_d, D, [t_S, t_E])$: make bandwidth reservations on the edges of a FPFB path within G to achieve the minimum data transfer cost while ensuring completion of the data transfer before the designated deadline t_E.
- $(v_s, v_d, D, t_S, C_{max})$: make bandwidth reservations on the edges of a FPFB path within G to attain the ECT for the data transfer while ensuring cost of the transfer not exceeding the specified maximum cost C_{max}.

In the notations above, v_s, v_d, and D correspond to the source node, destination node, and the total data size to be transferred from the earliest feasible data transfer start time t_S to the data transfer deadline t_E, respectively [11].

Each edge $e \in E$ is associated with a set of residual or available bandwidth functions defined with respect to time. Each function is represented as $B(e, [t_i^e, t_{i+1}^e])$, signifying the available bandwidth of edge e during time slot $[t_i^e, t_{i+1}^e]$, where $i = 0, 1, \ldots, T_e - 1$. Here, T_e indicates the total number of time slots that edge e comprises within time interval $[T_S, T_E]$. The term "time dot" refers to each of the starting and ending time points of these time slots. Collectively, all the time dots associated with the edges of G are organized into a sorted set, which encompasses unique and sorted elements arranged in ascending order. If we denote this sorted set as $\{t_0, t_1, \ldots, t_n\}$, an interval in the form of $[t_i, t_{i+1}]$, $0 \le i < n$, is referred to as an intersected time slot. Within this slot, the available bandwidth of edge e is denoted as $B(e, [t_i, t_{i+1}])$.

We refer to the time interval $[t_i, t_j]$, $0 \le i < j \le n$, as a time window. Time window $[t_i, t_j]$ contains consecutive intersected time slots $[t_i, t_{i+1}]$, $[t_{i+1}, t_{i+2}]$, $\ldots$, $[t_{j-1}, t_j]$. The available bandwidth of edge e during $[t_i, t_j]$, denoted as $B(e, [t_i, t_i])$, is determined by taking the minimum among the available bandwidths in each of the intersected time slots, i.e., $min\,(B(e, [t_i, t_{i+1}]), B(e, [t_{i+1}, t_{i+2}]), \ldots, B(e, [t_{j-1}, t_j]))$. For a given path p expressed as $e_0 - e_1 - \cdots - e_{m-1}$, its available bandwidth within the time window $[t_i, t_j]$, represented as $B(p, [t_i, t_j])$, is determined by considering the minimum available bandwidth along its constituent edges, namely $min\,(b(e_0, [t_i, t_j]), B(e_1, [t_i, t_j]), \ldots, b(e_{m-1}, [t_i, t_j]))$.

A successful scheduling of a given BRR is achieved when we can pinpoint a bandwidth reservation (BR) that fulfills all the BRR's requirements. This BR is denoted as $((p, b, [t_i, t_j]), [t_s, t_e], c)$, signifying the allocation of b bandwidth units on FPFB p during the time window $[t_i, t_j]$, while data transfer takes place in the time interval $[t_s, t_e]$ with an associated transfer cost of c.

For a given BRR, it is possible to identify multiple BRs, even potentially an infinite number of them. Among these options, we are particularly interested in two: the BR with the minimum data transfer cost (BR-MinC) and the one with the earliest completion time (BR-MinT). This paper's focal point is the algorithmic design aimed at pinpointing BR-MinC for a BRR-MinC, as well as BR-MinT for a BRR-MinT.

B. Data Transfer Cost Model

We define the bandwidth resource of an edge e within intersected time slot $[t_i, t_{i+1}]$ as the maximum amount of data that e can transfer within $[t_i, t_{i+1}]$, specifically $B(e, [t_i, t_{i+1}]) \cdot (t_{i+1} - t_i)$. The weight of an edge $e \in E$, denoted as $w(e)$, signifies the cost coefficient associated with data transfers over that edge. This coefficient is primarily influenced by factors such as the actual link distance, maintenance cost of the link, and other relevant considerations. We assume that the total cost of reserving b bandwidth units on edge e within $[t_i, t_{i+1}]$, as charged by the bandwidth reservation service provider, is equal to $w(e) \cdot B(e, [t_i, t_{i+1}]) \cdot (t_{i+1} - t_i)$. Considering a BRR with a corresponding BR of the form $((p, b, [t_i, t_j]), [t_s, t_e], c)$, where path p follows the sequence $e_0 - e_1 - \cdots - e_{m-1}$, the transfer cost c can be calculated using the following equation:

$$
\begin{aligned}
c = \sum_{x=i}^{j-1} \sum_{y=0}^{m-1} w(e_y) \cdot b \cdot (t_{x+1} - t_x) \\
= b \cdot \sum_{x=i}^{j-1} (t_{x+1} - t_x) \cdot \sum_{y=0}^{m-1} w(e_y) \\
= b \cdot (t_j - t_i) \cdot \sum_{y=0}^{m-1} w(e_y). \quad (1)
\end{aligned}
$$

The summation $\sum_{y=0}^{m-1} w(e_y)$ signifies the total weight of all edges of path p, commonly referred to as the weight of p, denoted as $w(p)$ for simplicity. Consequently, the data transfer cost c can be succinctly articulated as $b \cdot (t_j - t_i) \cdot w(p)$.

IV. Algorithm Design and Analysis

In this section, we focus on the optimal algorithm designs and analysis for scheduling a BRR-MinC and a BRR-MinT.

A. Optimal Algorithm Design for Scheduling a BRR-MinC

Please refer to Algorithm 1 for the detailed algorithm design and pseudocode of Opt-MinC. In the worst case, its complexity is $O\left(n^2 \cdot (|E| + |V| \cdot \log|V|)\right)$, where n is the number of time dots of G within time interval $[T_S, T_E]$. Note that δ in Line 8 of Algorithm 1 denotes the bandwidth unit.

Theorem 1. *Opt-MinC returns the BR-MinC, if it exists, for the input BRR-MinC.*

Optimality proof is omitted due to space limit

Algorithm 1 Optimal Algorithm for Scheduling a BRR-MinC and Returning the BR-MinC (Opt-MinC)

GIVEN: $G(V, E)$ within $[T_S, T_E]$
INPUT: BRR-MinC $(v_s, v_d, D, [t_S, t_E])$
OUTPUT: BR-MinC of the input BRR-MinC or $NULL$ if no BR exists.

1: Create a sorted set STD containing all the time dots of G within $[T_S, T_E]$, and then identify index u of the largest element in STD that is no larger than t_S and index v of the smallest element that is no less than t_E;
2: Declare and initialize data transfer cost $c = +\infty$ and BR-MinC $br = NULL$;
3: **for** $u \le i \le (v-1)$ **do**
4:   Declare and initialize variable $j = i$;
5:   **for** each $e \in E$ **do**
6:     $B(e, [t_i, t_j]) = +\infty$;
7:   **for** $i + 1 \le j \le v$ **do**
8:     Declare and initialize variable $b = \lceil \frac{D}{(min(t_E, t_j) - max(t_S, t_i)) \cdot \delta} \rceil \cdot \delta$;
9:     **for** each $e \in E$ **do**
10:       $B(e, [t_i, t_j]) = min(B(e, [t_i, t_{j-1}]), B(e, [t_{j-1}, t_j]))$;
11:       Declare a flag for e and initialize value to $False$;
12:       **if** $B(e, [t_i, t_j]) \ge b$ **then**
13:         Set the flag value of e to $True$;
14:     Execute Dijkstra's algorithm to determine and retrieve the path with the minimum weight p from v_s to v_d through edges with $True$ flag values;
15:     **if** $p \ne NULL$ and $c < b \cdot w(p) \cdot (t_j - t_i)$ **then**
16:       $c = b \cdot w(p) \cdot (t_j - t_i)$;
17:       $br = \left((p, b, [t_i, t_j]), [max(t_S, t_i), min(t_E, t_j)], c\right)$;
18: **return** br.

B. Optimal Algorithm Design for Scheduling a BRR-MinT

Please refer to Algorithm 2 for the detailed algorithm design and pseudocode of Opt-MinT. In the worst case, its complexity is $O\left(n^2 \cdot |E| \cdot (|E| + |V| \cdot \log|V|)\right)$, where n is the number of time dots of G within time interval $[T_S, T_E]$.

Theorem 2. *Opt-MinT returns the BR-MinT, if it exists, for the input BRR-MinT.*

Optimality proof is omitted due to space limit

V. Performance Evaluation

The OSCARS (On-Demand Secure Circuits and Advance Reservation System) offered by ESnet enjoys widespread adoption within the scientific community [4], [12]–[14]. To ensure the authenticity and fidelity of our performance evaluation, we replicate the ESnet infrastructure by constructing a network topology based on data gathered from ESnet [11], [15]. Subsequently, we embark on extensive simulations leveraging this topology.

Our simulation endeavor encompasses a total of 10 distinct sets, each denoted as simulation i, where $1 \le i \le 10$. Each of these simulations comprises 10 batches of BRRs, with the

Algorithm 2 Optimal Algorithm for Scheduling a BRR-MinT and Returning the BR-MinT (Opt-MinT)

GIVEN: $G(V, E)$ within $[T_S, T_E]$
INPUT: BRR-MinT $(v_s, v_d, D, t_S, C_{max})$
OUTPUT: BR-MinT of the input BRR-MinT or $NULL$ if no BR exists.

1: Create a sorted set STD containing all the time dots of G within $[T_S, T_E]$ and then identify index u of the largest element in STD that is no larger than t_S;
2: Declare and initialize data transfer completion time $ect = +\infty$ and BR-MinT $br = NULL$;
3: **for** $u \leq i \leq (|STD| - 2)$ **do**
4: The same as Lines 4 – 6 of Algorithm 1;
5: **for** $i + 1 \leq j \leq (|STD| - 1)$ **do**
6: The same as Lines 9 – 10 of Algorithm 1;
7: Declare and initialize a list $sortedEdges$ containing edges sorted by their available bandwidths in the descending order by default. Make a deep copy of all the edges E of G and add them to $sortedEdges$;
8: **for** each $e' \in sortedEdges$ **do**
9: **if** $i + 1 < j$ and $B(e', [t_i, t_j]) \geq \frac{D}{t_{j-1} - max(t_S, t_i)}$ **then**
10: $B(e', [t_i, t_j]) = \lceil \frac{D}{(t_{j-1} - max(t_S, t_i)) \cdot \delta} \rceil \cdot \delta - \delta$;
11: **if** $B(e', [t_i, t_j]) < \lceil \frac{D}{(t_j - max(t_S, t_i)) \cdot \delta} \rceil \cdot \delta$ **then**
12: Delete e' from $sortedEdges$;
13: Make a deep copy of all the nodes V of G, suppose the network is G' (G' does not contain any edges);
14: Declare and initialize variable $k = 0$;
15: **while** $k \leq |sortedEdges| - 1$ **do**
16: $b = B(sortedEdges[k], [t_i, t_j])$;
17: Add edge $sortedEdges[k]$ to G';
18: **while** $(k < |sortedEdges| - 1)$ and $(b == B(sortedEdges[k + 1], [t_i, t_j]))$ **do**
19: Add edge $sortedEdges[k + 1]$ to G';
20: $k + +$;
21: **if** $k < |sortedEdges| - 1$ **then**
22: $b' = B(sortedEdges[k + 1], [t_i, t_j]) + \delta$;
23: **else**
24: $b' = \lceil \frac{D}{(t_j - max(t_S, t_i)) \cdot \delta} \rceil \cdot \delta$;
25: Execute Dijkstra's algorithm to determine and retrieve the path with the minimum weight p from v'_s to v'_d in G';
26: **if** $p \neq NULL$ **then**
27: **for** $0 \leq x \leq \frac{b - b'}{\delta}$ **do**
28: **if** $(b - x \cdot \delta) \cdot w(p) \cdot (t_j - t_i) \leq C_{max}$ and $max(t_S, t_i) + \frac{D}{b - x \cdot \delta} < ect$ **then**
29: $ect = max(t_S, t_i) + \frac{D}{b - x \cdot \delta}$;
30: $br = \big((p, b - x \cdot \delta, [t_i, t_j]), [max(t_S, t_i), ect], (b - x \cdot \delta) \cdot w(p) \cdot (t_j - t_i) \big)$;
31: Continue the loop in Line 5;
32: $k + +$;
33: **return** br.

number of BRRs in each batch being $i \times 100$. In every BRR, we randomly designate two nodes, v_s and v_d, from the node set. The bandwidth unit is standardized at 100, while the data size D is generated as a random integer within the range of [10, 2000] multiplied by the bandwidth unit. Both t_S and t_E of each BRR are random integers within the interval [0, 20]. The cost threshold C_{max} is assigned a random number ranging from 20 to 30, and the weight of each edge is set to a random integer from 1 to 10, subsequently divided by 10000. The scheduling network is confined within the time interval [0,30].

As previously detailed in Section II, we explored analogous problems in [9], and therein introduced two algorithms, namely Opt-MinC-TC and Opt-MinT-CC. We proceed to implement these aforementioned algorithms, along with the algorithms presented in this paper, subsequently subjecting them to performance comparison. Both Opt-MinC and Opt-MinC-TC, as well as Opt-MinT and Opt-MinT-CC, are executed to process identical batches of BRRs. Following the BRR processing phase, we amass multiple performance metrics and subsequently craft corresponding figures (Figs. 1 – 4). For the sake of accuracy in our experimental results, each figure represents the average measurements of performance metrics, accompanied by corresponding variances at a 95% confidence level, drawn across the various simulation sets.

Fig. 1 visually captures the average transfer cost per data unit of scheduled BRRs computed by Opt-MinC, revealing it to be slightly higher than that computed by Opt-MinC-TC. This discrepancy arises from our assumption that bandwidth reservation necessitates resource allocation for entire time slots, even if those slots are required only partially for the data transfer. Both algorithms manifest similar time requirements for processing a BRR (Fig. 2). On the other hand, Fig. 3 and 4 divulge that the average completion time of data transfers for the scheduled BRRs computed by Opt-MinT is notably lower than that computed by Opt-MinT-CC. Furthermore, Opt-MinT demonstrates swifter processing times for individual BRRs. These results collectively underscore the superior performance of Opt-MinT in contrast to Opt-MinT-CC.

VI. Conclusion

This study delved into the intricate balance between cost and the earliest completion time (ECT) of data transfers in the context of bandwidth reservation on fixed paths with fixed bandwidths within HPNs, the most common type of bandwidth reservation or data transfer paths. We focused on the scheduling of two types of bandwidth reservation requests (BRRs): (i) minimizing the data transfer cost while ensuring deadline adherence and (ii) achieving the data transfer ECT while constraining cost. In response to these challenges, we presented two optimal algorithms to efficiently optimize the scheduling of both BRR types. Through extensive simulations and comparisons with existing algorithms, we demonstrated the effectiveness and efficiency of our proposed algorithms.

Moving ahead, we aim to investigate similar trade-off problems on different data transfer paths within HPNs.

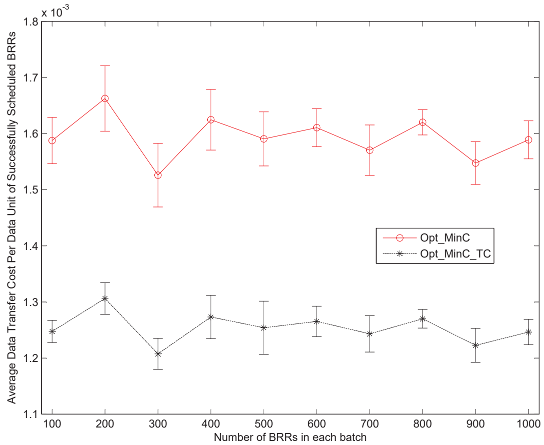

Figure 1: Average transfer cost per data unit of scheduled BRRs computed by Opt-MinC and Opt-MinC-TC.

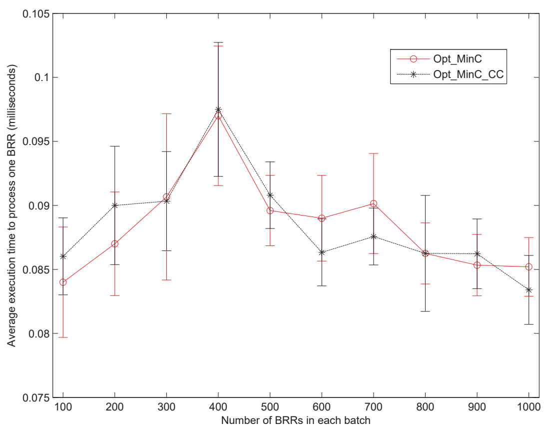

Figure 2: Average time needed to process one BRR by Opt-MinC and Opt-MinC-TC.

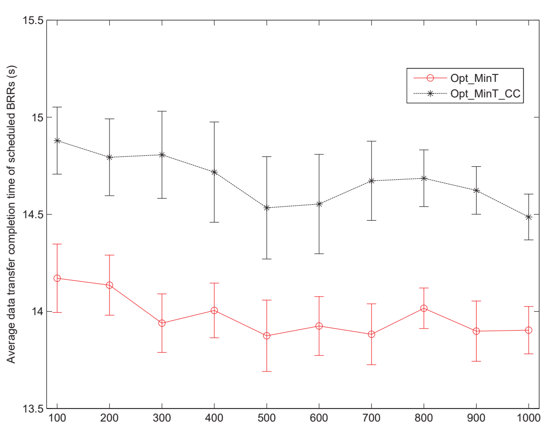

Figure 3: Average data transfer ECT of scheduled BRRs computed by Opt-MinT and Opt-MinT-CC.

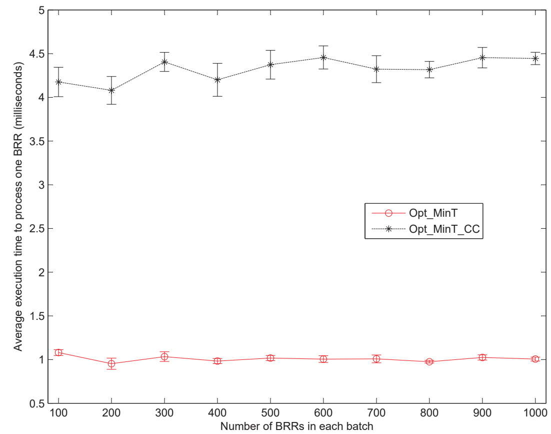

Figure 4: Average time needed to process one BRR by Opt-MinT and Opt-MinT-CC.

REFERENCES

[1] Y. Lin and Q. Wu, "Complexity analysis and algorithm design for advance bandwidth scheduling in dedicated networks," *IEEE/ACM Trans. on Netw.*, vol. 21, no. 1, pp. 14–27, Feb. 2013.

[2] L. Zuo, M. M. Zhu, and C. Q. Wu, "Concurrent bandwidth reservation strategies for big data transfers in high-performance networks," *IEEE Trans. Netw. Serv. Manage.*, vol. 12, no. 2, pp. 232–247, June 2015.

[3] M. Balman, E. Chaniotakisy, A. Shoshani, and A. Sim, "A flexible reservation algorithm for advance network provisioning," in *Proc. of the 2010 ACM/IEEE Int. Conf. for High Perform. Comput., Netw., Storage and Anal.*, Washington, DC, USA, 2010, pp. 1–11.

[4] C. Guok, D. Robertson, M. Thompson, J. Lee, B. Tierney, and W. Johnston, "Intra and interdomain circuit provisioning using the oscars reservation system," in *3rd Int. Conf. on Broadband Commun., Netw. and Syst.*, Oct 2006, pp. 1–8.

[5] L. Zuo, M. M. Zhu, and C. Q. Wu, "Bandwidth reservation strategies for scheduling maximization in dedicated networks," *IEEE Trans. Netw. Serv. Manage.*, vol. 15, no. 2, pp. 544–554, 2018.

[6] G. Yang, Y. Yoo, M. Kang, H. Jin, and C. Yoo, "Bandwidth isolation guarantee for sdn virtual networks," in *IEEE Conference on Computer Communications (INFOCOM)*, 2021, pp. 1–10.

[7] A. A. Al–khatib, M. U. Hassan, and K. Moessner, "Heuristic optimization of bandwidth reservation cost for vehicular applications," in *IEEE Global Communications (GLOBECOM)*, 2022, pp. 4909–4915.

[8] X. Zhang and T. Wang, "Elastic and reliable bandwidth reservation based on distributed traffic monitoring and control," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 4563–4580, 2022.

[9] L. Zuo, M. Zhu, and C. Chang, "Optimizing trade-off between cost and performance of data transfers using bandwidth reservation in dedicated networks," *J. of Network and Syst. Manage.*, vol. 27, no. 1, pp. 166–187, Jan. 2019.

[10] R. A. Guérin and A. Orda, "Networks with advance reservations: the routing perspective," in *INFOCOM 2000*, vol. 1, 2000, pp. 118–127.

[11] L. Zuo, M. Zhu, and C. Wu, "Fast and efficient bandwidth reservation algorithms for dynamic network provisioning," *J. of Network and Syst. Manage.*, vol. 23, no. 3, pp. 420–444, Jul 2015.

[12] N. Charbonneau, V. M. Vokkarane, C. Guok, and I. Monga, "Advance reservation frameworks in hybrid ip-wdm networks," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 132–139, May 2011.

[13] T. Lehman, X. Yang, N. Ghani, F. Gu, C. Guok, I. Monga, and B. Tierney, "Multilayer networks: an architecture framework," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 122–130, May 2011.

[14] I. Monga, C. Guok, W. E. Johnston, and B. Tierney, "Hybrid networks: lessons learned and future challenges based on esnet4 experience," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 114–121, May 2011.

[15] "ESnet Network Weathermap," https://my.es.net/, accessed July 5, 2023.