Multi-stage Aggregation and Lightpath Provisioning of Geo-distributed Data over EON Assisted by MEC

Zhen Liu, Jiawei Zhang, Zizheng Guo, Yuefeng Ji

State Key Lab of Information Photonics and Optical Communications, Beijing Univ. of Posts and Telecommunications (BUPT), Beijing, China; Email: {liuzhen207, zjw, zzg, jyf}@bupt.edu.cn

Abstract: A multi-stage aggregation and lightpath provisioning algorithm is proposed for geo-distributed data in EON assisted by MEC. Simulation results show the algorithm can reduce the job completion time and bandwidth consumption.

OCIS codes: (060.4250) networks; (060.4251) Networks, assignment and routing algorithms.

1. Introduction

With the unprecedented scale of growing Internet of Things (IoT) devices, mobile edge computing (MEC) is introduced to bring computing and storage closer to IoT devices for enhanced service/application performance [1]. One of the typical application scenarios is multiple edge datacenters (EDC) collaboration for IoT job, since an IoT job may involve distributed data from sensors at different places. The geo-distributed data is firstly processed at the local edge datacenters (EDCs) for latency and bandwidth reduction, then transmitted by the network to a selected EDC for aggregation. The data processing at EDC can reduce the bandwidth through filtering irrelevant data for the job. The aggregation is to collect the intermediate data to an EDC for further processing or convergence. This is also described as a coflow problem in the literatures [2], in which the job completion time (JCT) depends on the distributed task (data) processing and transmission time.

The current approaches in the literature exploit "single-stage aggregation", which means their works gather all the required geo-distributed data of a job to a selected EDC through once network transmission and data aggregation [3, 4]. However, as data volumes continue to grow in an unprecedented manner, such "single-stage aggregation" approach may not be practical feasible: first, when a large number of geo-distributed data are transmitted to a selected EDC at the same time, the competition of network resources near the selected EDC is severe, which results in increased transmission delay. Second, owing to the limited processing capacities of single EDC, processing delay is increased when all the required intermediate data of a job are processed in single EDC.

To alleviate these issues, we proposed a multi-stage aggregation and lightpath provisioning (MSALP) algorithm of geo-distributed data in elastic optical network (EON) assisted by MEC to minimize the JCT. The proposed algorithm is to gather geo-distributed data into one EDC through multiple stages and divide the geo-distributed data into multiple clusters for parallel processing in each stage.

2. System Architecture and Problem Description

Fig. 1 shows a geo-distributed data storage scenario in EON assisted by MEC, which includes multiple EDCs. Each EDC stores multiple types of raw data generated by nearby IoT devices. Multiple EDCs are connected through the EON. The EON can achieve more flexibility in bandwidth allocation. Fig. 2(a) shows an example of single-stage aggregation. The all requested data of a job is aggregated into one EDC through once data transmission in the network. Fig. 2(b) illustrates an example of multi-stage aggregation. First, the geo-distributed data is divided into four clusters at stage 1. In each cluster, raw data are first processed locally within a local EDC (Map). Then, the intermediate data are transmitted to cluster center for data aggregation (Reduce 1). Finally, the process moves to the stage 2 after all clusters of the first stage is finished. The intermediate data generated after the stage 1 is further aggregated at stage 2 (Reduce 2). When all intermediate data is transmitted to one EDC, the job is completed.

In the above process, to reduce the data volume, instead of relaying received data directly, the cluster center of each cluster perform data aggregation. In Fig. 3, we describe how the data is transmitted and aggregated. At stage *m*,

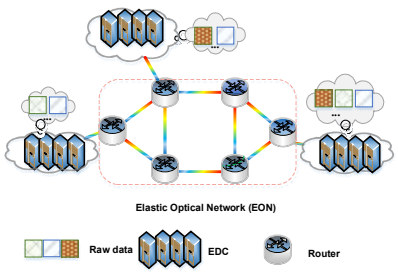


Fig. 1 Geo-distributed data storage scenario in EON

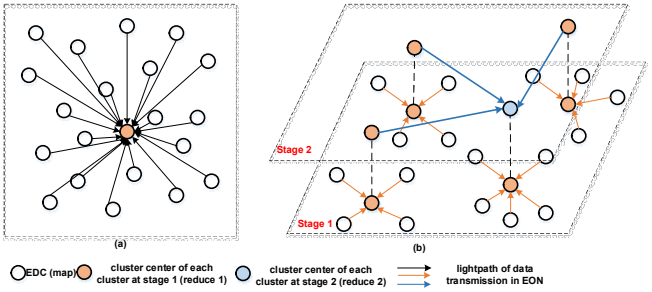


Fig. 2 Geo-distributed data aggregation in EON. (a) single-stage aggregation; (b) multi-stage aggregation

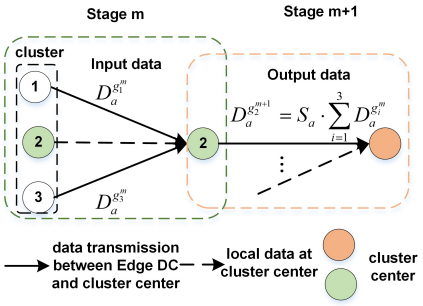
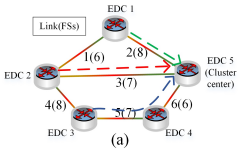


Fig. 3 An example of data transmission and aggregation cross stage



Candidate path	Unavailable link
Link 6	Link 6
Link (5-4-3)	Link 5 and 3
Link (5-4-1-2)	Link 5 and 2

(c)

EDC	Data Volume	Path	$n_{r,r'}^{pre}$	$t_{r,r'}$
1	4 Gbit	Link 2	8	0.08 s
2	3 Gbit	Link 3	7	0.068 s
3	2 Gbit	Link (5-6)	6	0.053 s

(b)

EDC	Data Volume	Path	$n_{r,r'}^{pre}$	$t_{r,r'}$
1	4 Gbit	Link 2	8	0.08 s
2	3 Gbit	Link 3	7	0.068 s
3	2 Gbit	Link (5-6)	4	0.08 s
4	1 Gbit	Link 6	2	0.08 s

(d)

Fig. 4 An example of routing and FSs allocation in a cluster.

a cluster includes EDC 1, 2 and 3. And, the EDC 2 is selected as cluster center. In cluster, data from EDC 1 and EDC 3 are transmitted to the cluster center (EDC 2). The all data volume of cluster center is $D_a^{g_1^m} + D_a^{g_2^m} + D_a^{g_3^m}$. After processing or convergence in EDC 2, the output data volume of cluster center is $D_a^{g_2^{m+1}} = S_a \cdot (D_a^{g_1^m} + D_a^{g_2^m} + D_a^{g_3^m})$, where S_a is the data selectivity as the result of data aggregation. The output data of stage m is used as the input data of stage m+1, which is further processed in stage m+1.

Fig. 4 shows an example of routing and frequency slots (FSs) allocation when data is transmitted over EON. In a cluster, there are five EDCs that store required data, of which EDC 5 is cluster center, as shown in Fig. 4(a). To reduce the transmission time, data stored in EDC 1, 2 and 3 select the path with the maximum available FSs $n_{r,r'}^{pre}$, as shown in Fig. 4(b). For data stored in EDC 4, we select three candidate paths between EDC 4 and cluster center. None of the three candidate paths has available FSs, and we search unavailable links on each candidate path in Fig. 4(c). Due to the **completion time of each cluster** (CCT) depends on the transmission time of the slowest data, we reallocate FSs of the data with the minimum transmission time. Combining Fig. 4(b) and 4(c), we find that the data 3 has minimum transmission time and link 6 is occupied by data 3. Thus, we reallocate FSs on link 6 according to the ratio of data volume. In Fig. 4(d), the transmission time of data 3 and data 4 is 0.08s according to the reallocated FSs.

3. Delay-Optimal Multi-stage Aggregation and Lightpath Provisioning Algorithm

In this section, we propose a MSALP algorithm of geo-distributed data to minimize the JCT. The JCT is equal to the sum of the completion time of multiple stages, and the **completion time of each stage** (SCT) depends on the completion time of the bottleneck cluster that lags behind the most. The CCT is composed of transmission time and processing time. The transmission time depends on data volume and FSs. The processing time depends on data volume and processing capability of cluster center. Thus, we first determine clusters and cluster centers (Procedure 1). Then, we calculate CCT by allocating routing and FSs (Procedure 2). Finally, we calculate the SCT and JCT.

Step 1: Determine cluster centers. First, we calculate the metric w_r^m by considering the node degree (*Deg*), the available bandwidth (*ASPB*), data volume (*D*) and the processing capability of each EDC (*AC*). We select the EDC that has the largest w_r^m as the first cluster center. The EDC is farthest from the first initial cluster center as the second cluster center to reduce the competition of network resources. The *kth* cluster center is select according to the distance L_k of shortest path between the *k*th EDC and other cluster centers. Then, the EDC and its nearest cluster center are divided into a cluster. Finally, we re-select the EDC that has the largest metric w_r^m as cluster center in each cluster. ***Step 2***: Calculate CCT. To minimize the transmission time, we first select the path with the largest available FSs. If there is no available FSs on any of the candidate paths, we re-allocate the FSs according to the ratio of data volume, which has minimum transmission time and occupied unavailable link. Then, we calculate the transmission time of each data in the cluster according to Eq. (1) and the processing time according to Eq. (2). The

Algorithm 1: MSALP algorithm for geo-distributed data

1: Initialize the number of EDCs in each cluster n according to node degree
2: **for** stage $m \in M$ **do**
3: calculate the number of clusters according to $k=\lceil |R^m|/n \rceil$
4: **if** k >=2 **then**
5: determine k cluster centers in **Procedure 1**
6: calculate the CCT in **Procedure2**
7: calculate $T^m = \max_{r \in \Phi m} T^{g_r^m}$ 8: **end if**
9: **if** k=1 **then**
10: the job is completed, calculate T^m by invoking **Procedure 2**, $T^m = T^{g_r^m}$
11: **break**
12: **end if** 13: **end for**

Procedure 1: Determine cluster centers

Initial cluster centers:

1: calculate the metric w_r^m for EDC $r^m \in R^m$ with $w_{r^m} = \begin{cases} Deg_{r^m} \cdot ASPB_{r^m} \cdot AC_{r^m} \cdot D_{r^m}, if\ m=1 \\ Deg_{r^m} \cdot ASPB_{r^m} \cdot AC_{r^m} \cdot D_a^{g_r^m}, if\ m>1 \end{cases}$
2: select the first initial cluster center p_1^m
3: select EDC farthest from p_1^m as p_2^m
4: calculate the distance L_k
5: $\Omega^m \leftarrow \arg_{k \in R^m/\Omega^m} \max L_k$
6: divide k clusters

Re-determine cluster centers:

7: **for** each cluster $g_r^m \in G^m$ **do**
8: **for** data of EDC $r \in g_r^m$ **do**
9: calculate the metric w_r^m 10: **end for**
11: $\Phi^m \leftarrow \arg_{r \in g_r^m} \max w_r^m$ 12: **end for**

Procedure 2: Calculate the CCT

1: **for** each cluster $g_r^m \in G^m$ **do**
2: **for** data of EDC $r' \in g_r^m / \{r\}$ **do**
3: **if** has available FSs on candidate paths
4: select the path k with maximum FSs $n_{r,r'}$
5: **if** no available FSs on candidate paths
6: re-allocate FSs of data, which has pre-allocated routing and FSs
7: calculate the transmission time $t_{r,r'}$ $t_{r,r'} = D_a^{g_r^m} / (n_{r,r'} \cdot C_{slot})$ (1) 8: **end for**
9: $t^{g_r^m} \leftarrow max_{r' \in g_r^m / \{r\}} t_{r,r'}$
10: calculate the processing time of cluster with $t_a^{g_r^m} = \left(D_a^{g_r^m} + \sum_{r' \in g_r^m / \{r\}} D_a^{g_r^m} \right) / C_r$ (2)
11: calculate $T^{g_r^m} = t^{g_r^m} + t_a^{g_r^m}$ (3)12: **end for**

maximum $t_{r,r'}$ in a cluster is set as the transmission time $t^{g_r^m}$ of cluster. Thus, the CCT $T^{g_r^m}$ is calculated according to Eq. (3). ***Step 3***: Calculate JCT. The maximum $T^{g_r^m}$ in a cluster is selected as the SCT T^m. Due to the JCT is equal to the sum of the completion time of all stages, the JCT T equals $T = \sum_{m=1}^{M} T^m$.

4. Performance Evaluation

Fig. 5 shows a metro network topology, which deploys EDC at each node [5]. The processing capacity of EDC is uniformly distributed within [1, 5] *10^2 Gbit/s. The available FSs on the links is uniformly distributed within [5, 20] FSs. The capacity C_{slot} of one FS is 6.25Gb/s. The required data of a job are modeled by Zipf distribution among the EDCs. The data volume D_r of the raw data is uniformly distributed within [10, 50] Gbit.

In Fig. 6 shows the completion time of each stage and JCT under the different number of EDCs in each cluster. First, we observe that as the average number of EDCs in a cluster increases, the number of stages required to complete job decreases. Second, we observe that the JCT increases firstly and then reduces gradually with the increase of the average number of EDCs in a cluster. This is because when there are fewer EDCs in each cluster, more stages are needed to complete geo-distributed data aggregation. When there are more EDCs in each cluster, the competition of network resources around the cluster center will be severe and the transmission time will be increased. This results in increased completion time for each stage. Results show that the average number of EDCs in each cluster can weigh the number of stages and the completion time of each stage.

In Fig. 7 and Fig. 8 show the JCT and the bandwidth consumption of data aggregation policies under various job width, respectively. The job width refers to the number of geo-distributed data in a job. We fix the average number of EDCs in each cluster as 4. As shown in Fig. 7, we first observe that the JCT of single-stage data aggregation is higher than that of MSALP algorithm (S_a=0.2, S_a=0.4) and lower than that of MSALP algorithm (S_a=0.8). This is because when the data selectivity factor S_a is large, the amount of data transmitted to the next stage is still large, which lead to larger transmission time and processing time. When the data selectivity factor S_a is small, the amount of data transmitted and processed in the next stage will be significantly reduced, thus reducing the transmission time and processing time of next stage. Secondly, we observe that the larger job width, the better the performance of our proposed MSALP algorithm. This is because the data are processed in parallel by grouping at each stage in the MSALP algorithm, so the competition of network resources around a single cluster center can be alleviated. From the above simulation results, it is concluded that the MSALP algorithm is suitable for scenarios with better data aggregation effect and larger amount of data.

Fig. 8 shows that the MSALP algorithm has a higher total bandwidth consumption than single-stage aggregation algorithm. This is because the MSALP algorithm gathers geo-distributed data stored in multiple EDCs into one EDC through multiple stages. The data in each stage need to be transmitted through the network. Thus, the more stages and the more bandwidth consumption. However, the bandwidth consumption of each stage in MSALP algorithm is less than that of the bandwidth consumption of single-stage aggregation algorithm. Therefore, the MSALP algorithm reduces the occupation of network resources in the same time period.

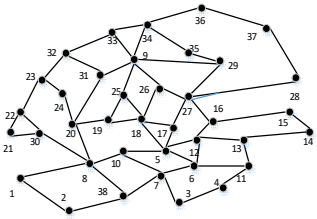


Fig .5 Metro network topology

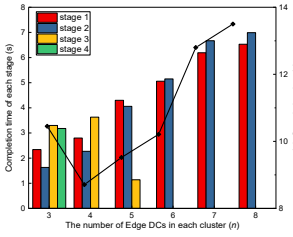


Fig. 6 The completion time of each stage and JCT

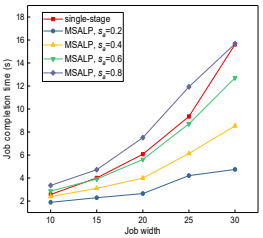


Fig. 7 The JCT

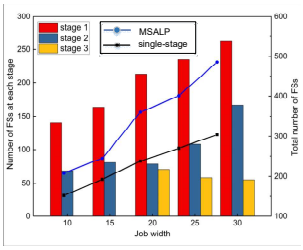


Fig. 8 The bandwidth consumption

5. Conclusion

We proposed a MSALP algorithm, which includes determining the number of clusters, the location of each cluster center, routing and FSs allocation. Simulation results showed the MSALP algorithm could reduce JCT and bandwidth consumption at the scenario of larger data amount and better data aggregation effect.

Acknowledgment This work was supported by the National KeyR&D Program of China (2018YFB1800802), the National Nature Science Foundation of China Projects (61871051, 61971055), the Beijing Natural Science Foundation (4192039), the fund of State Key Laboratory of Information Photonics and Optical Communications, China, IPOC2019ZT05, and BUPT Excellent Ph.D. Students Foundation (CX2019310).

Reference

[1] Y. Hu, et al., "Mobile edge computing a key technology towards 5G," ETSI White Paper, 2015.
[2] S. Wang S, et al., "A survey of coflow scheduling schemes for data center networks," IEEE Communications Magazine, 2018.
[3] Z. Hu, et al., "Flutter: Scheduling Tasks Closer to Data Across Geo-Distributed Datacenters," *Proc .INFOCOM*, San Francisco, USA, 2016.
[4] L. Chen, et al., "Scheduling Jobs across Geo-Distributed Datacenters with Max-Min Fairness", *IEEE T on Net Sci and Eng*., 2018.
[5] Z. Liu, et al., "Joint jobs scheduling and lightpath provisioning in fog computing micro datacenter networks," *J Opt Commun Netw*., 2018.