Received August 31, 2017, accepted October 16, 2017, date of publication October 27, 2017, date of current version December 5, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2767058

Priority Control in Communication Networks for Accuracy-Freshness Tradeoff in Real-Time Road-Traffic Information Delivery

SHINGO KATO AND RYOICHI SHINKUMA

Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan

Corresponding author: Shingo Kato (skato@icn.cce.i.kyoto-u.ac.jp)

This work was supported in part by JSPS KAKENHI under Grant 17H01732.

ABSTRACT Delivering real-time road-traffic information to the driver is a straightforward solution to the problem of road-traffic congestion. The information is more effective as it is fresh and more accurate. However, real-time road-traffic information delivery has a fundamental problem: an accuracy-freshness tradeoff. Unfortunately, real-time road-traffic information delivery has difficulty satisfying both requirements. To guarantee the freshness, the information needs to be delivered on the basis of the data received by a cloud or edge server before a predetermined deadline. However, only a limited amount of data is received due to bandwidth limitation and processing overhead in communication networks, which results in the poor accuracy of the delivered information. The only way to improve the accuracy is to make the deadline less strict, which results in deteriorating the freshness of information. The proposed system solves this tradeoff. The key idea is that data more "important" for the accuracy of information are more prioritized when the data are transferred in communication networks. In the proposed system, "importance" is determined by how helpful the data are when the system needs to estimate missing spatial information from a limited amount of received data by using the machine learning technique. In this paper, simulation results verify that the proposed system ensures the accuracy of road-traffic information while satisfying the freshness requirement.

INDEX TERMS Internet of Things, road-traffic information, real-time information delivery, priority control, active learning, edge computing.

I. INTRODUCTION

Road traffic congestion is still a serious problem in many countries, creating huge adverse economic and environmental impacts. According to a report by the Centre for Economics and Business Research [1], the total economy-wide costs across four advanced countries (UK, France, Germany, and USA) are forecast to rise from US$200.7 billion in 2013 to US$293.1 billion by 2030. The report also suggests that the four economies will incur a cumulative social cost related to CO2 emissions of around US$13 billion between 2013 and 2030.

Delivering fine-grained information on road traffic conditions to vehicles is a straightforward solution to the congestion problem [2]. Such information would help drivers, which could be robots [3] as well as humans in the near future, optimize their route in terms of traveling time and fuel consumption. Especially, realtime information is more useful than past information because it supports more suitable route choices.

Realtime road-traffic information delivering can be considered as a promising application in the context of Internet of Things (IoT) [4]. Fig. 1 illustrates a typical system and the components in an application that delivers information created from IoT data. First, data collection is done by IoT devices, which could be smartphones, wearable devices, smart meters, and so on. In the road-traffic scenario, probe vehicles or road-side cameras work for collecting data as IoT devices. Then, to upload IoT data to the cloud or edge server, IoT devices transfer it through communication networks. Finally, to produce information useful for people, the edge or cloud server aggregate IoT data and perform computational processing, which could include conversion of data format and estimation of missing data.

However, realtime road-traffic information delivery has a fundamental problem: an accuracy-freshness tradeoff. The accuracy of road-traffic information is an index that indicates the accuracy of the spatial information about road traffic delivered to drivers. The freshness of road-traffic information

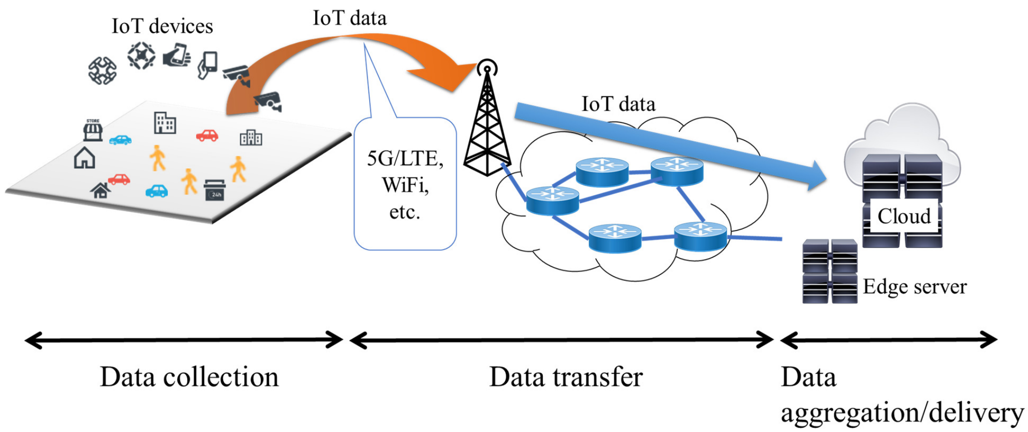


FIGURE 1. Typical system for IoT information delivery.

means how soon drivers can obtain road-traffic information: the more recently the information is updated, the more useful it should be for drivers. Unfortunately, realtime road-traffic information delivery has difficulty satisfying both requirements. To guarantee the freshness, the information needs to be delivered on the basis of the data received by a cloud or edge server before a predetermined deadline. However, only a limited amount of data is received due to bandwidth limitation and processing overhead in communication networks, which results in the poor accuracy of the delivered information. The only way to improve the accuracy is to make the deadline less strict, which results in deteriorating the freshness of information.

However, no previous work has focused on this problem. The previous works that work on road-traffic information delivery can be categorized into the following two. The one focused on how to reduce latency for realtime delivery without discussing how to improve the accuracy of information. These conventional efforts have mainly work on data screening, routing, and resource assignment [5]–[7]. The other one is about prediction and estimation of road-traffic information using machine learning [8]–[10]. Their objective is how to improve the accuracy; latency in communication networks have not been considered in those works.

Therefore, this paper proposes a data transfer control method that solves the accuracy-freshness tradeoff in realtime road-traffic information delivery. The key idea in the proposed system is that data more 'important' for the accuracy of information is prioritized when the data is transferred in communication networks. 'Importance' is determined by how helpful the data is when the system needs to estimate missing spatial information from a limited amount of received data by using the machine learning technique [11]. The recent developments of software-defined-networking (SDN) technology [12] and active-learning technology [11] have enabled us to realize our key idea. By using the proposed system, since data more important for accurate information is transferred with higher priority, the freshness of information is guaranteed while the accuracy of information is ensured even with the bandwidth limitation and processing overhead in communication networks. In this paper, we present the problem formulation of our system and show the results of a performance evaluation using a real dataset to verify that the proposed method the ensures accuracy of information while satisfying freshness requirement.

The rest of this paper is organized as follows. Section II introduces the technical background of our study. Section III presents the problem formulation of this study and the proposed priority control method. Section IV provides the performance evaluation of the proposed method through a simulation. Finally, we conclude this paper in Section V.

II. BACKGROUND TECHNOLOGIES

In this section, we briefly introduce basic ideas of the SDN and active learning. Several studies related to delivering the road-traffic information are also mentioned.

A. SDN

SDN has several good features are useful to collect, transfer, and aggregate big data: the separation of the control and data planes, logically centralized control, global view of the network, and ability to program the network [13]. For example, Tortonesi et al. proposed SPF (Sieve, Process, and Forward) [14] to address the explosion of IoT data by dynamic data filtering that is based on Value of Information. SDN is used to filter information.

There have been some recent research efforts in using SDN to deliver road-traffic information. For example, Jiao et al. proposed a predictive data collection algorithm called Preco [5] that collects road-traffic data with low latency in access networks. In Preco, routing and scheduling decisions are made in a centralized manner in accordance with realtime network status, e.g. realtime vehicle position, speed, and the road on which a vehicle runs. To realize this, SDN is utilized. He et al. minimized the communication cost while the requirements of application can still be satisfied by utilizing SDN as a unified resource manager [6]. Originally, SDN was designed for and deployed in wired network environments. However, as in the above examples, SDN has also been applied to wireless sensor networks in some research. Especially, many studies have applied SDN to vehicular ad hoc networks (VANET) [7], [15], [16].

The system we propose also utilize SDN to control the data transfer on the basis of the data importance.

B. ACTIVE LEARNING

Active learning is a subfield of machine learning. The supervised training system needs many labeled instances, e.g., speech recognition, classification, and filtering. The purpose of active learning is to achieve high accuracy while using as few labeled instances as possible, thereby minimizing the cost of obtaining labeled data [11]. It is not necessary to obtain labeled instances in the application for delivering road-traffic information. Conversely, it is necessary to aggregate the data by spending the communication cost. The proposed method obtains the requested accuracy of the information from as little data as possible on the basis of the idea of active learning. Matrix completion (MC) is used to estimate road-traffic information and is particularly studied in relation to active learning among machine learning technologies [8], [9], [17]. MC estimates the missing value of matrix. Conventionally, active learning methods that treat the matrix form like MC have been studied for collaborative filtering for recommendation systems [18]–[21]. However, active learning methods that are not limited to recommendation systems and the naive model of MC have also been studied. They are called active matrix completion (AMC) [22]–[24].

III. PROPOSED DATA TRANSFER CONTROL SYSTEM

A. SYSTEM OVERVIEW

Fig. 2 overviews the proposed system. The service scenario we assume in this paper is that the system provides real-time road-traffic information delivery for human or robotic drivers. As shown in Fig. 2, probe vehicles, which are moving around regional areas, collect images by on-board cameras and upload them to the edge server via access and regional networks. The edge server extracts spatial information about road traffic from images received from probe vehicles. When some information is missing, as described in Fig. 2, the edge server estimates it from obtained information by using the machine-learning technology. Then, road-traffic information is provided periodically on a realtime basis.

B. DETAILED SYSTEM MODEL

Fig. 3 depicts the model of the proposed system we assume in this paper. The proposed system consists of probe vehicles, SDN routers, and an edge server, which are interconnected by communication networks. Each component in the proposed system is described in detail as follows.

1) PROBE VEHICLES

Probe vehicles work like moving sensors to collect images using their on-board cameras. This type of sensing is called

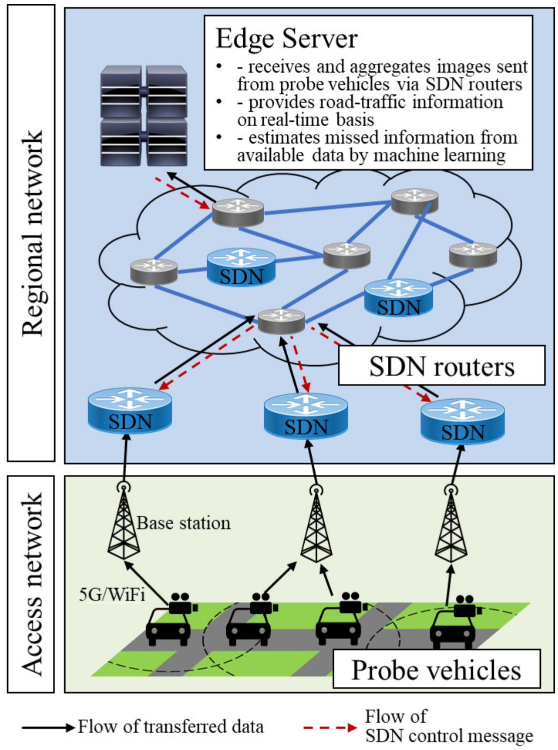


FIGURE 2. System model of proposed system.

mobile crowd sensing (MCS) [8], [9]. MCS is known as a smart way compared with the conventional way like using roadside cameras in terms of cost, coverage, and accuracy. Probe vehicles upload image data formatted as in Fig. 4(a), which includes image itself, time, and coordinate, to the edge server at every unit-time. Each probe vehicle is connected to a base station close to it in the access network and can send its data through a wireless channel. However, to simplify the discussion in our study, the wireless channel in the access network is idealized: unlimited bandwidth and no transmission loss. Furthermore in this paper, we assume that probe vehicles themselves are not controllable in the proposed system because they should be basically independent of network operation.

2) SDN ROUTERS

SDN routers, which are placed to physically interconnect between the access network and the edge server in the regional network as in Fig. 3, transfer images collected by probe vehicles to the edge server. SDN routers are the main controllable entities based on the policy of the proposed system, which will be described in detail in Section III-D. SDN routers are controlled by the SDN controller contained in the edge server, which is operated by a network operator.

3) EDGE SERVER

We consider edge computing as computational resource for processing the aggregated data. In general, since edge servers are placed in a regional network, which is closer to the network edge rather than core, it provides lower latency and

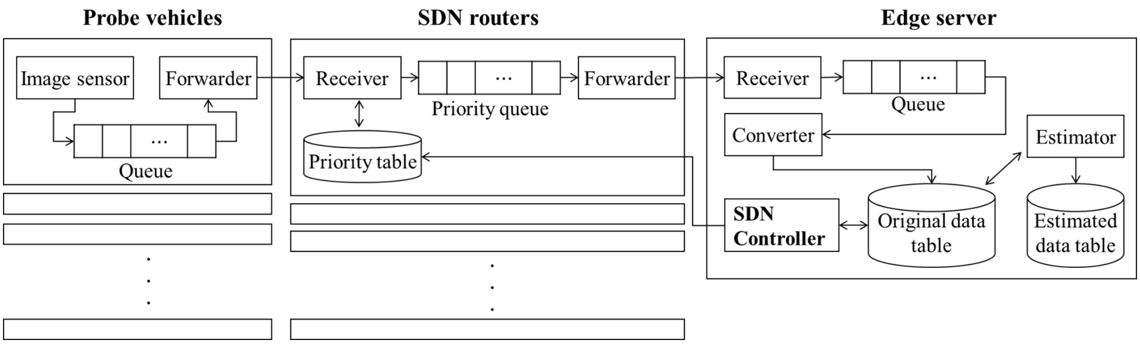


FIGURE 3. Block diagram of proposed system.

enables us to ensure realtime application more easily than cloud servers [25]. The edge server receives images uploaded by probe vehicles and extracts raw data shown in Fig. 4(b) from them. The automatic license plate recognition technology would be helpful in such data extraction [26] though the detail should be omitted in this paper. Then, the extracted data is formatted as shown in Fig. 4(c) and stored.

Fig. 4(d) shows the format of information delivered periodically on a realtime basis. The row and column in this information mean areas and time slots, while the entry of a specific set of a row and a column indicates the number of vehicles observed in the specific area during the specific time slot. As you can easily imagine, the information in Fig. 4(d) can be created by aggregating data shown in Fig. 4(c). However, when the edge server can not receive data for all the entries before the deadline for realtime delivery, it needs to estimate missing entries by using a machine-learning method. Fig. 4(e) shows the information in which missing entries have been completed after estimation. We assume edge servers are equipped with sufficient computational resources; computational overhead at edge servers is not considered and is out of the scope of this paper.

C. PROBLEM FORMULATION

As we mentioned in Section I, the technical challenge we tackle through this paper and our system solves is the tradeoff between accuracy and freshness of information in road-traffic information delivery. That is, as the bandwidth is limited more strictly or the processing overhead is larger in the regional network in Fig. 3, it takes longer to transfer image data from probe vehicles to the edge server. This section presents a mathematical formula of the above problem.

Here, we suppose that the period of delivering road-traffic information is T and the throughput of the regional network is given as the number of images the regional network can transfer during T, $\theta(T)$, which is determined by bandwidth and processing capacities of the regional network. To guarantee realtime delivery of road-traffic information, the following requirement must be satisfied:

$$|D(T)| \leq \theta(T) \tag{1}$$

where $D(T)$ denotes a set of images that probe vehicles collect during T and $|D(T)|$ denotes the number of elements of $D(T)$. If (1) is not satisfied, the system overflows and cannot deliver the information in real time. $\theta(T)$ is a time-dependent variable that could be changed by background traffic and network failure. As $\theta(T)$ becomes smaller, only a more limited number of images can be transferred from probe vehicles to the edge server during T, which results in deteriorating the accuracy of spatial information of road traffic.

Our proposed system solves the above tradeoff. The approach of the proposed system is to prioritize more 'important' data by the SDN technology when it is transferred in the regional network. The 'importance' of each image here is assessed on the basis of how much it contributes to improving the accuracy of road-traffic information when the system estimates missing information by a machine-learning method. The mathematical formulation is given below:

$$\min \quad \sum_{L=1}^{N_a} Error(D_{used}(T_c), L) \tag{2}$$

$$\text{s.t.} \quad D_{used}(T_c) = \bigcup_{T=T_c-N_u}^{T_c} \bigcup_{L=1}^{N_a} D_{aggregated}(T, L) \tag{3}$$

$$D_{aggregated}(T, L) \subseteq D(T, L)$$
$$(L = \{1, \cdots, N_a\}, \quad T \leq T_c)$$

$$\sum_{L=1}^{N_a} |D_{aggregated}(T, L)| \leq \theta(T) \tag{4}$$

where $D_{used}(T_c)$ in (2) denotes the set of images used at time T_c to estimate missing information by a machine-learning method. Formula (3) means $D_{used}(T_c)$ is the set of images aggregated over locations 1 to N_a during the period from $T_c - N_u$ to T_c. $D_{aggregated}(T, L)$ denotes the set of images collected at location L during period T. Formula (4), which comes from (1), shows the constraint for guaranteeing the realtime delivery in the system. As presented in (2), the objective of the proposed system is to maximize the accuracy of road-traffic information by minimizing the estimation error while satisfying the requirement of realtime delivery given by (4).

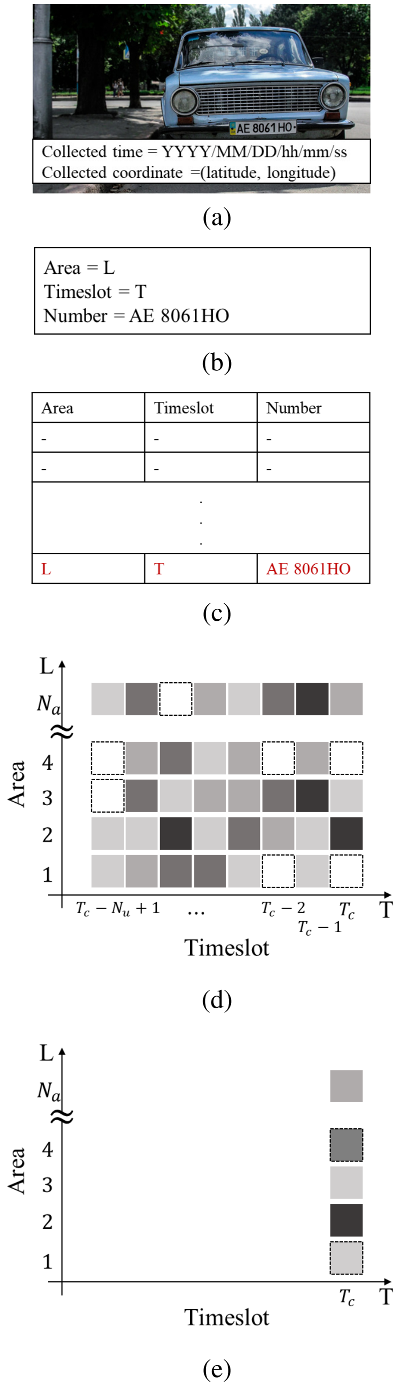


FIGURE 4. Data and information formats. (a) Image data. (b) Raw data. (c) Original data. (d) Traffic information with missing values, T_c means the timeslot of current time. N_a and N_u denotes the numbers of areas and time slots, which correspond rows and columns of the matrix, respectively. (e) Traffic information after estimation for delivering.

D. PROPOSED CONTROL METHOD

In the proposed system, the network operator operates communication networks that interconnect between probe vehicles and the edge server so that more 'important' data is transferred with higher priority. As mentioned in the previous section, the 'importance' of data is determined on the basis of how much the data contributes to the accuracy in estimating road-traffic information when there is missing information. The proposed control method consists of the following steps.

1) Each image data is associated with one of the areas based on the position where the image was captured by a probe vehicle.
2) The contribution to the estimation of data in the associated area is referred.
3) The priority is applied to the data based on the contribution of the associated area.

Thus, the prioritization in the proposed method is performed on an area-by-area basis. Active learning and SDN are key technologies in our system. The former enables us to evaluate which piece of data contributes more to machine-learning based estimation. The latter is a communication network technology that enables the network operator to control routers flexibly and dynamically in accordance with their policy.

As depicted in Fig. 3, an SDN router in the proposed method consists of a receiver, a forwarder, a priority queue, and a priority table. The receiver in an SDN router receives images and inputs them to the priority queue on the basis of the predetermined priority obtained from the priority table, which is produced by the SDN controller at the edge server using the active learning method mentioned above. The forwarder forwards images from the priority queue to the next router, which should be closer to the edge server.

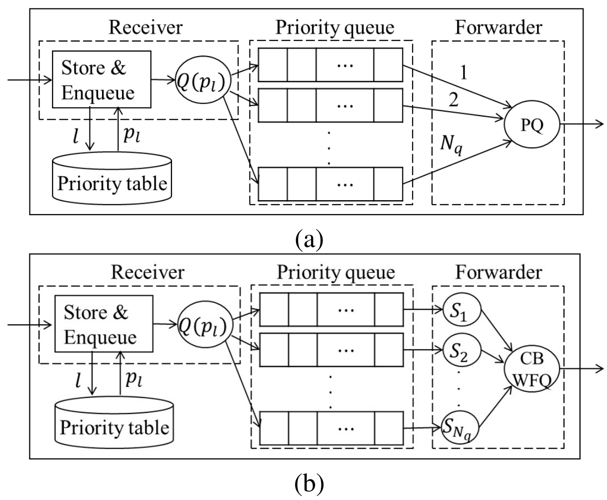


FIGURE 5. Queueing methods with proposed method. (a) PQ. (b) CBWFQ.

The following paragraphs describe how we can actually implement the proposed method using SDN routers. There are two simple but commonly used queuing methods: priority queuing (PQ) and class-based weight fair queuing (CBWFQ) [27].

1) PQ

Fig. 5(a) illustrates how the proposed method works in the case of PQ. As illustrated in Fig. 5(a), the receiver stores the received data from the previous router and enqueues it to the queue. The receiver attaches a priority label to each piece of data. Since, in road-traffic information delivery, each data is associated with a location. Therefore, priorities of data can be represented as:

$$p_L = \{1, 2, 3, \ldots, N_a\} \tag{5}$$

$$p_i \neq p_j \quad (1 \leq i < j \leq N_a) \tag{6}$$

where 1 means the highest priority and N_a means the lowest priority. The priority p_L is updated by the SDN controller periodically. Then, the receiver forwards data to the queue after attaching the priority label p_L to the data. In accordance with p_L, Q determines to which queue the forwarded data is assigned. The number of queues N_q and the function Q are predetermined before the operation. Finally, the forwarder forwards the data from the queue with the highest priority among the queues that are not empty.

2) CBWFQ

Fig. 5(b) illustrates how the proposed method works in the case of CBWFQ. Basically, the receiver in CBWFQ behaves in the same manner as that in PQ. However, the difference of CBWFQ from PQ is that opportunities to dequeue from each queue are assigned in accordance with predetermined rates S_q though the queue with the higher priority always obtains the opportunity in PQ.

IV. PERFORMANCE EVALUATION

A. EVALUATION SCENARIO

A simulation study was performed to evaluate the effectiveness of the proposed method described in the previous section, which makes it possible to solve the accuracy-freshness tradeoff in realtime road-traffic information delivery. In our simulation, the system periodically delivers road-traffic information with period Δ_d. Therefore, the number of delivering opportunities N_d is equal to P_s/Δ_d, where P_s is the total simulation time. As shown in Fig. 4(d), the form of road- traffic information in our simulation is the number of vehicles observed within each area at each time slot (15 min in our simulation). The information at each time slot is aggregated from $t_c - \Delta_d$ to delivering time t_c. In this section, the time duration from $t_c - \Delta_d$ to t_c is represented as T_c, which is the time slot at the current time t_c.

The parameters used in our simulation are listed in Table 1. We built a simulator specified for this study by using C++. In our simulation, we produced realistic vehicles movements using the epfl/mobility dataset, which is available from the CRAWDAD repository [28]. The dataset contains GPS coordinates and the time of each measurement of 536 taxis, which were collected for about 30 days in San Francisco, California. US. Some pre-processing was necessary: linear interpolation was performed to shape the interval of the raw data into uniformly 1 minute. Moreover, since 536 taxis seemed a little small as road traffic, we overlay the 2nd- and 3rd-week data to the 1st-week data and produce new 1 week data with 1608 taxis. Fig. 6 shows the geographical range of our simulation. We divided it equally and defined each of divided ones as an area.

TABLE 1. Simulation parametars.

Parameter	Value
Period of information delivery (Δ_d)	15 minutes
Simulation length (P_s)	24 hours
No. of time slots in delivered information (N_d)	96
No. of areas in delivered information (N_a)	32
No. of time slots utilized for estimation (N_u)	48
No. of access SDN routers	4
No. of intermediate SDN routers (N_I)	0, 1 or 2
Background traffic	none
No. of queues (N_q)	4
Queue size	unlimited
No. of probe vehicles (N_P)	100
Sensing radius (R_s)	100 meters
Sensing duration (Δ_s)	1 minute

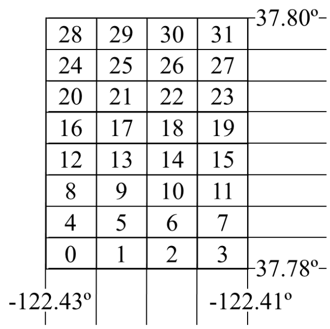


FIGURE 6. Ranges of each area in simulation.

B. SIMULATION MODEL

The simulation model in our simulation basically follows the block diagram given in Fig. 3, which consists of probe vehicles, SDN routers, and the Edge server with the SDN controller. Each of the entities is operated on the basis of different time scale and rule. The edge server receives observed data from probe vehicles via the SDN routers during a period Δ_d and produces road-traffic information periodically with Δ_d as mentioned above. It also performs machine-learning based estimation to estimate missing information, which is not received from probe vehicles by the limited throughput of the communication network. At the same timing, the priority table at SDN routers is updated. SDN routers are operated by the event-driven basis; when any of queues in a SDN router is not empty, it is operated to forward data to the next router. Each probe vehicle uploads its observed data to the edge server at every minute via the connected SDN router, which varies in accordance with its location.

1) EDGE SERVER

The form of the road-traffic information delivered by the edge server, $\boldsymbol{M}^*$, is a matrix where the number of rows and columns are equal to the numbers of time slots N_u and areas N_a. N_u denotes the number of past time slots used for estimation. Therefore, the entry of each row and each column of $\boldsymbol{M}^*$

corresponds to the number of vehicles observed within an area at a time slot. $\boldsymbol{M}^*$ at the T_c-th delivery is represented as $\boldsymbol{M}^*(T_c)$ and the entry (the number of observed vehicles) of an area $L \in \{1, 2, \ldots, N_a\}$ and time slot $T \in \{T_c - N_u + 1, T_c - N_u + 2, \ldots, T_c\}$ is represented as $M_{LT}(T_c)$.

However, in our study, $\boldsymbol{M}^*(T_c)$ can not be always produced from data observed by probe vehicles by the limited throughput of the communication network. Therefore, the edge server estimates missing entries of $\boldsymbol{M}^*(T_c)$ using a machine-learning method called Matrix Completion (MC), which allows us to estimate a data matrix from a sampling of its entries using low rank nature of the data [29]. MC is well used in road-traffic information delivery [8], [9]. The mathematical expression of MC is generally given as:

$$\min \quad \|\boldsymbol{X}\|_* \tag{7}$$

$$\text{subject to} \quad X_{ij} = M_{ij} \quad (i, j) \in \Omega \tag{8}$$

where $\boldsymbol{X}$ is the matrix that will be completed. $\|\boldsymbol{M}\|_*$ is the nuclear norm of $\boldsymbol{X}$ and $\boldsymbol{M}$ is correct data matrix. Ω denotes the set of sampled entries of $\boldsymbol{M}$.

We extended (7) for our simulation as below:

$$\min \quad \|\boldsymbol{X}(T_c)\|_* \tag{9}$$

$$\text{subject to} \quad X_{LT}(T_c) = M^*_{LT}(T_c) \quad (L, T) \in \Omega \tag{10}$$

where $\boldsymbol{M}^*(T_c)$ is the input matrix used for estimation and it is available from the aggregated data $D_{used}(T_c)$ described in Section III-C by changing the original data form to the matrix form. Note that when $D_{aggregated}(T, L) = D(T, L)$, $\boldsymbol{M}^*(T_c)$ is completely same as $\boldsymbol{M}(T_c)$. As the difference between the estimated matrix, $\boldsymbol{X}(T_c)$ and the correct matrix, $\boldsymbol{M}(T_c)$ becomes smaller, we would say the accuracy of estimation is better.

2) SDN CONTROLLER

The SDN controller is operated with period Δ_d to update the priority of data p_L of $\forall\, L$ and inform all of the SDN routers of it. The important role of the SDN controller is to evaluate the contribution of each entry of $\boldsymbol{M}(T_c)$, $M_{LT}(T_c)$, to the estimation of missing entries. As mentioned in Section III-D, the evaluation is done on an area-by-area basis. AMC, which was mentioned in Section II, should be used for such evaluation [22]–[24]. However, since discussion and implementation of AMC should be the out of the scope of our paper for the simplification of research objective, we evaluate the contribution of each entry in the following way. Algorithm 1 shows the pseudo code for the contribution evaluation in our simulation. The contribution of each area is calculated by using the actual matrix at T_c, $\boldsymbol{M}^*(T_c)$ and all the correct entries, $\{M_{1T_c}(T_c + 1), \ldots, M_{N_aT_c}(T_c + 1)\}$. Although in reality, the correct entries are not available, we assumed that they are ideally known. Algorithm 1 inputs the area that contributes to accuracy most to *UsedAreas* in the greedy manner. The function in Algorithm 1, *Accuracy* is the summation of errors between the estimated and ideal entities at $T_c + 1$ in the estimation of the matrix shown in Fig. 7. As shown

Algorithm 1 Pseudo Code for Contribution Evaluation in Simulation

```
Areas ← {1, 2, ..., N_a}
UsedAreas ← Ø
while Areas ≠ Ø do
    // Accuracy() given by Fig. 7
    NextArea = arg min_{L∈Areas} Accuracy(UsedAreas ⋃{L})
    Areas ← Areas \ {L}
    UsedAreas ← UsedAreas ⋃{L}
end while
return UsedAreas
```

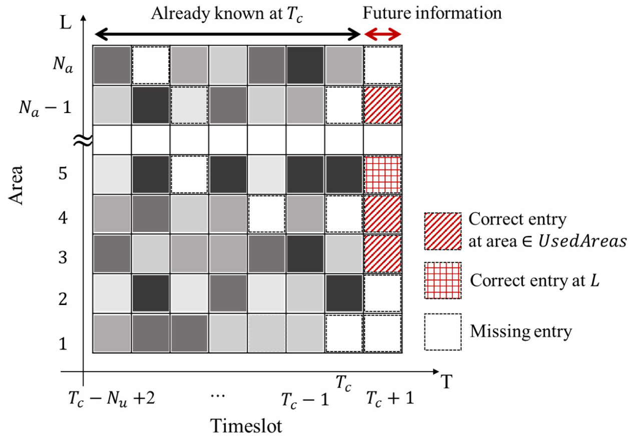


FIGURE 7. Matrix used for estimation in *Accuracy*().

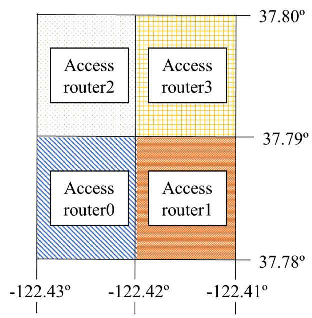


FIGURE 8. Association of access SDN routers with areas.

in this figure, this matrix is composed of the combination of the actual entries obtained until T_c and only the entries of *UsedAreas* and area L in the correct entries at $T_c + 1$ Finally, this algorithm adopts the order of area that was input to *UsedAreas* as the order of the contribution p_L.

3) SDN ROUTERS

Fig. 9 shows the network topology in our simulation, which is composed of probe vehicles, access and intermediate SDN routers, and the edge server. The number of access SDN routers is 4, while the number of intermediate SDN routers N_I, which was varied from 0 to 2 as a simulation parameter. In the network given in Fig. 9, the end-to-end

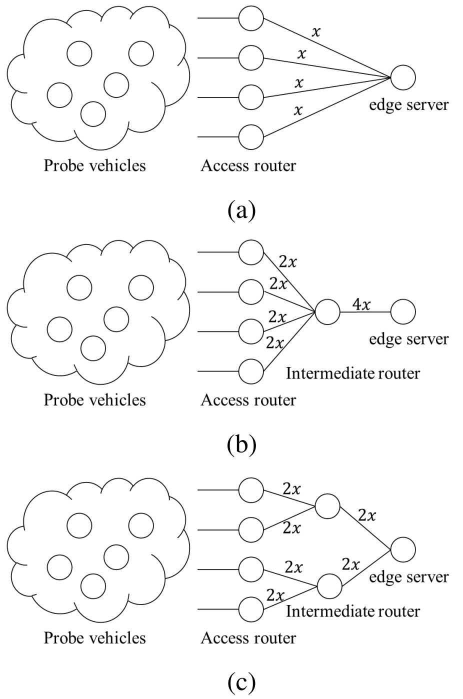


FIGURE 9. Simulation topologies. Throughput $\theta = 4x$. (a) Topology 0 ($N_I = 0$). (b) Topology 1 ($N_I = 1$). (c) Topology 2 ($N_I = 2$).

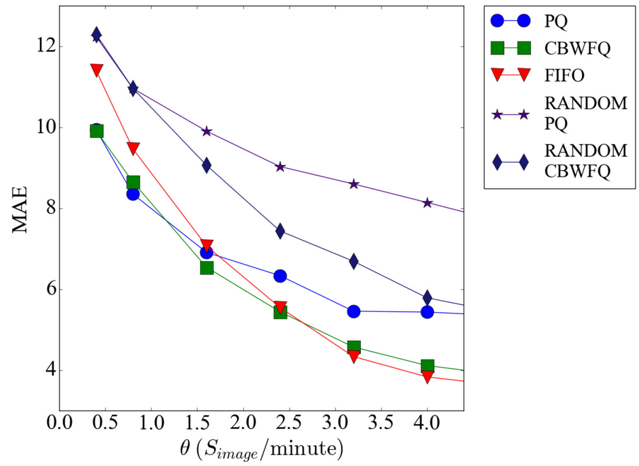


FIGURE 10. Comparison in mean absolute error of estimation vs. throughput in topology 0 ($N_I = 0$).

throughput from probe vehicles to the edge server is limited by the bottleneck links between the last-hop SDN routers and the edge server. Also, our simulation does not consider background traffic. Therefore, in our simulation, θ is given as the sum of the throughputs of those bottleneck links. Each probe vehicle is associated with one of the access SDN routers in accordance with its current position as illustrated in Fig. 8, which depicts the relation between a coordinate and the access SDN router associated with probe vehicles in the coordinate.

As described in Section III-D, SDN routers are operated in accordance with the priority table that is controlled by the SDN controller. In our simulation, PQ and CBWFQ are considered as a priority queueing method. The numbers of queues in both methods are commonly set to N_q. Then, function Q, which is used to assign data obtained from area L to the queue in accordance with its priority p_L, is defined as:

$$Q(p_L) = \begin{cases} 1 & (1 \leq p_L < \lceil \frac{1}{10} N_a \rceil) \\ 2 & (\lceil \frac{1}{10} N_a \rceil \leq p_L < \lceil \frac{1+2}{10} N_a \rceil) \\ 3 & (\lceil \frac{1+2}{10} N_a \rceil \leq p_L < \lceil \frac{1+2+3}{10} N_a \rceil) \\ 4 & (otherwise) \end{cases} \tag{11}$$

where the priority with the smaller number means the higher priority. In our simulation, the number of queues, N_q, follows the reference to the default setting of priority queuing given by Cisco Systems [30], which is four. Note that PQ always chooses data to be forwarded next from the queue that has the highest priority among the queues that are not empty. However, differently from PQ, CBWFQ gives opportunities to each queue on the basis of the speed S_q that is predetermined to each queue $q = 1, 2, 3, 4$, which also follows the reference setting given by Cisco and is represented as:

$$\{S_1, S_2, S_3, S_4\} = \{\frac{4\theta}{10}, \frac{3\theta}{10}, \frac{2\theta}{10}, \frac{\theta}{10}\} \tag{12}$$

In our simulation, the capacities of each queue are set unlimited. However, instead of the capacities, we set the deadline of holding time of data in the queues; old data are deleted every time the SDN controller updates the priority table.

4) PROBE VEHICLES

In our simulation, the epfl/mobility dataset is also used to produce movements of probe vehicles. However, only N_P vehicles are picked up as probe vehicle from all of the vehicles. Each probe vehicle senses the range within R_s from it and observes the number of vehicles in the range. It cumulates observed data during Δ_s and immediately sends it to the access SDN router to which it is connected.

C. METRIC AND BENCHMARK

The Mean Absolute Error (MAE) is utilized for evaluation in this paper. The MAE is given by [31]:

$$MAE = \frac{\sum_{T_c=1}^{N_d} \sum_{L=1}^{N_a} Error(D_{used}(T_c), L)}{N_d N_a} \tag{13}$$

$$= \frac{\sum_{T_c=1}^{N_d} \sum_{L=1}^{N_a} |X_{LT_c}(T_c) - M_{LT_c}(T_c)|}{N_d N_a} \tag{14}$$

where $\boldsymbol{X}(T_c)$ is the matrix that is estimated for T_c-th delivering and $\boldsymbol{M}(T_c)$ is the correct data matrix of $\boldsymbol{X}(T_c)$. $\sum_{L=1}^{N_a} Error(D_{used}(T_c), L)$ corresponds to the objective function in (2) and denotes the sum of all the absolute errors at the time of the T_c-th delivering. As mentioned above, N_d is the number of opportunities of delivering information during the simulation time. Therefore, the MAE is given by averaging estimation errors over N_d.

We set two benchmark methods in our performance evaluation: FIFO and RANDOM. In the former, queues in routers are operated in the first-in first-out manner without any prioritization. In the latter, priorities are assigned at random to data

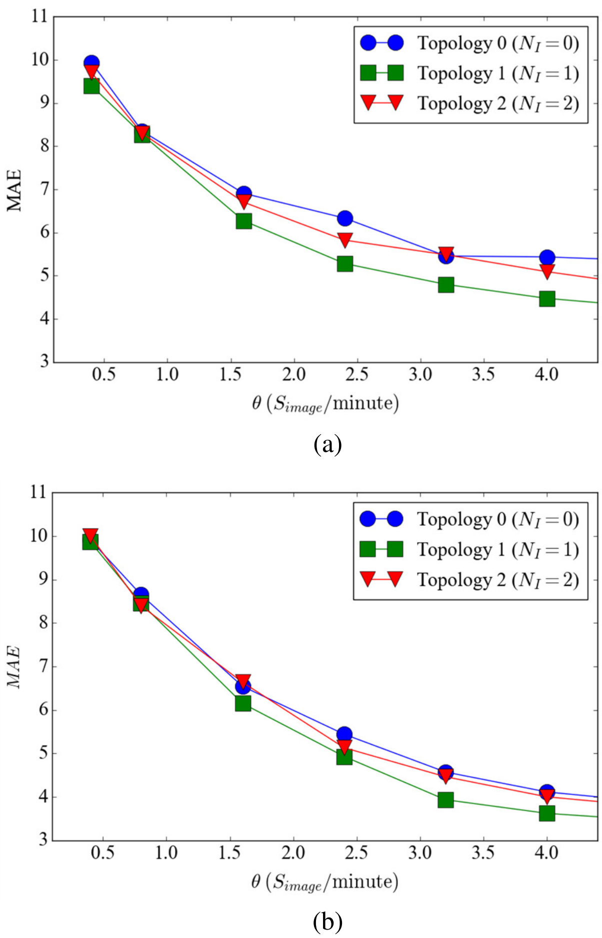


FIGURE 11. **Mean absolute error of estimation vs. throughput in topologies 0 to 2 ($N_I = 0, 1, 2$). (a) PQ (b) CBWFQ.**

though queues are operated on the basis of PQ or CBWFQ described above.

D. EVALUATION RESULTS

Fig. 10 shows the result of MAE, which was defined in (13), against the limited throughput θ in topology 0, where the number of intermediate routers was zero. In Fig. 10, we see the MAE basically increases as θ decreases in all the methods. This is obviously because it becomes harder to produce accurate road-traffic information as a more limited number of images are received by the edge server. From the figure, we should highlight that the proposed method with CBWFQ works well most stably against the wide range of θ. The proposed method with PQ worked well for the small throughputs though it became worse than FIFO for the large throughputs. In PQ, since queueing is deterministic based on priority, only data with the highest priority was always transferred, which increases the redundancy of data received by the edge server particularly when the throughput is high enough. Finally, through comparisons with RANDOM with PQ and CBWFQ, it has been validated that prioritization based on contribution to estimation accuracy, which is the key in the proposed method, works effectively.

Next, we now investigate the effect of the number of intermediate routers on the performance of the proposed method. Fig. 11 (a) and (b) show the MAE against θ in the proposed method with PQ and CBWFQ in topologies 0, 1, and 2, which corresponds to N_I = 0, 1, and 2, respectively. In both methods, the MAE became smallest in the wide range of θ when N_I = 1, followed by N_I = 2. As show in Fig. 9, when $N_I = 0$, since all data eventually passes through only the single intermediate router, the data is prioritized over different areas appropriately. However, when N_I = 1 or 2, since data from different areas are distributed to different routes before the edge server receives them, the prioritization for data between different areas does not work effectively. Considering the result in Fig. 10, in which the proposed method with CBWFQ still works better than the compared methods even when $N_I = 0$, the result in Fig. 11 suggests that the proposed method with CBWFQ would work further better than the compared methods.

V. CONCLUSION

In this paper, we proposed a data transfer control method to deliver highly accurate and fresh urban road-traffic information in real time. The proposed system solves the tradeoff between the accuracy and freshness of the delivered information on the basis of the key idea that data more 'important' for the accuracy of information is prioritized when it is transferred in communication networks. A simulation is performed to verify that the proposed method works well to ensure information accuracy while satisfying freshness requirement. The results indicate that the proposed system can deliver more accurate and fresh data than the conventional methods, especially under the severe throughput limitation in communication networks. For a more practical evaluation, in our future work, we will consider how to implement the proposed system by utilizing OpenFlow [32] as an SDN platform and AMC as the method for active learning. We will also consider other applications and queuing methods than the ones discussed in this paper and examine the scalability of the proposed system.

REFERENCES

[1] *The Future Economic and Environmental Costs of Gridlock in 2030, An Assessment of the Direct and Indirect Economic and Environmental Costs of Idling in Road Traffic Congestion to Households in the UK, France, Germany and the USA Report for INRIX*, Centre Econ. Bus. Res., London, U.K., 2016.

[2] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *IEEE Network*, vol. 27, no. 5, pp. 48–55, Sep./Oct. 2013.

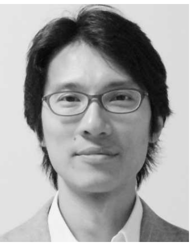
[3] P. Papadimitratos, A. De La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 84–95, Nov. 2009.

[4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[5] Z. Jiao, H. Ding, M. Dang, R. Tian, and B. Zhang, "Predictive big data collection in vehicular networks: A software defined networking based approach," in *Proc. Global Commun. Conf. (GLOBECOM)*, 2016, pp. 1–6.

[6] Z. He, D. Zhang, and J. Liang, "Cost-efficient sensory data transmission in heterogeneous software-defined vehicular networks," *IEEE Sensors J.*, vol. 16, no. 20, pp. 7342–7354, Oct. 2016.
[7] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudane, "Software defined networking-based vehicular adhoc network with fog computing," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, May 2015, pp. 1202–1207.
[8] R. Du, C. Chen, B. Yang, and X. Guan, "VANET based traffic estimation: A matrix completion approach," in *Proc. Global Commun. Conf. (GLOBECOM)*, 2013, pp. 30–35.
[9] R. Du, C. Chen, B. Yang, N. Lu, X. Guan, and X. Shen, "Effective urban traffic monitoring by vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 1, pp. 273–286, Jan. 2015.
[10] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 865–873, Apr. 2015.
[11] B. Settles, "Active learning literature survey," Univ. Wisconsin, Madison, WI, USA, Tech. Rep. 1648, p. 11.
[12] D. Kreutz, F. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
[13] L. Cui, F. R. Yu, and Q. Yan, "When big data meets software-defined networking: SDN for big data and big data for SDN," *IEEE Netw.*, vol. 30, no. 1, pp. 58–65, Jan./Feb. 2016.
[14] M. Tortonesi, J. Michaelis, N. Suri, and M. Baker, "Software-defined and value-based information processing and dissemination in IoT applications," in *Proc. IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, Apr. 2016, pp. 789–793.
[15] J. Liu *et al.*, "High-efficiency urban traffic management in context-aware computing and 5G communication," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 34–40, Jan. 2017.
[16] M. A. Salahuddin, A. Al-Fuqaha, and M. Guizani, "Software-defined networking for RSU clouds in support of the Internet of vehicles," *IEEE Internet Things J.*, vol. 2, no. 2, pp. 133–144, Apr. 2015.
[17] Q. Zhao, C. Chen, R. Du, S. Bi, and B. Yang, "HaTTC: An urban traffic sensing method based on tensor completion technique," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 175–180.
[18] C. Boutilier, R. S. Zemel, and B. Marlin, "Active collaborative filtering," in *Proc. 19th Conf. Uncertainty Artif. Intell.*, 2002, pp. 98–106.
[19] R. Jin and L. Si, "A Bayesian approach toward active learning for collaborative filtering," in *Proc. 20th Conf. Uncertainty Artif. Intell.*, 2004, pp. 278–285.
[20] A. S. Harpale and Y. Yang, "Personalized active learning for collaborative filtering," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2008, pp. 91–98.
[21] I. Rish and G. Tesauro, "Active collaborative prediction with maximum margin matrix factorization," in *Proc. ISAIM*, 2008, p. 20.
[22] S. Chakraborty, J. Zhou, V. Balasubramanian, S. Panchanathan, I. Davidson, and J. Ye, "Active matrix completion," in *Proc. IEEE 13th Int. Conf. Data Mining (ICDM)*, Dec. 2013, pp. 81–90.
[23] N. Ruchansky, M. Crovella, and E. Terzi, "Matrix completion with queries," in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1025–1034.
[24] C. Lan, Y. Deng, and J. Huan, "A disagreement-based active matrix completion approach with provable guarantee," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2016, pp. 4082–4088.
[25] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st ed. MCC Workshop Mobile Cloud Comput.*, 2012, pp. 13–16.
[26] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas, "A license plate-recognition algorithm for intelligent transportation system applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 377–392, Sep. 2006.
[27] X. Chen, C. Wang, D. Xuan, Z. Li, Y. Min, and W. Zhao, "Survey on QoS management of VoIP," in *Proc. Int. Conf. Comput. Netw. Mobile Comput. (ICCNMC)*, 2003, pp. 69–77.
[28] CRAWDAD. *Epfl/Modify Dataset*. Accessed: Apr. 13, 2016. [Online]. Available: http://crawdad.org/
[29] E. J. Candés and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, p. 717, 2009.
[30] *Cisco IOS Quality of Service Solutions Configuration Guide, Release 12.2*. Accessed: Sep. 30, 2017. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/ios/12_2/qos/configuration/guide/fqos_c/qcfpq.html
[31] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 4, Aug. 2009, Art. no. 421425.
[32] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Apr. 2008.

SHINGO KATO received the B.E. degree in electrical and electronic engineering from Kyoto University, Kyoto, Japan, in 2016. He is currently working toward the M.S. degree in communications and computer engineering at the Graduate School of Informatics, Kyoto University. His research interests include network modeling and system-design in IoT applications.



RYOICHI SHINKUMA received the B.E., M.E., and Ph.D. degrees in communications engineering from Osaka University, Osaka, Japan, in 2000, 2001, and 2003, respectively. In 2003, he joined the Communications and Computer Engineering Department, Graduate School of Informatics, Kyoto University, as an Associate Professor. He was a Visiting Scholar at the Wireless Information Network Laboratory, Rutgers, the State University of New Jersey, USA, from 2008 to 2009. His researches include network design and control criteria, particularly inspired by economic and social aspects. He is a Senior Member of the IEICE. He received the Young Researchers' Award from the IEICE in 2006 and the Young Scientist Award from Ericsson Japan in 2007, respectively. He also received the TELECOM System Award from the Telecommunications Advancement Foundation in 2016. He has been the Chairperson with the Mobile Network and Applications Technical Committee of the IEICE Communications Society since 2017.