

Efficient On-Demand Data Service Delivery to High-Speed Trains in Cellular/Infostation Integrated Networks

Hao Liang, *Student Member, IEEE*, and Weihua Zhuang, *Fellow, IEEE*

Abstract—In this paper, we investigate on-demand data services for high-speed trains via a cellular/infostation integrated network. Service requests and acknowledgements are sent through the cellular network to a content server, while data delivery is achieved via trackside infostations. The optimal resource allocation problem is formulated by taking account of the intermittent network connectivity and multi-service demands. In order to achieve efficient resource allocation with low computational complexity, the original problem is transformed into a single-machine preemptive scheduling problem based on a time-capacity mapping. As the service demands are not known a priori, an online resource allocation algorithm based on Smith ratio and exponential capacity is proposed. The performance bound of the online algorithm is characterized based on the theory of sequencing and scheduling. If the link from the backbone network to an infostation is a bottleneck, a service pre-downloading algorithm is also proposed to facilitate the resource allocation. The performance of the proposed algorithms is evaluated based on a real high-speed train schedule. Compared with the existing approaches, our proposed algorithms can significantly improve the quality of on-demand data service provisioning over the trip of a train.

Index Terms—Cellular/infostation integrated network, high-speed train, on-demand data service, resource allocation.

I. INTRODUCTION

Recently, the high-speed rail has been rapidly developing all over the world [2]. The rail not only can significantly shorten journey times, but also can improve passenger comforts by high-speed Internet services [3]. The cellular network deployed near the rail lines can provide seamless coverage. However, the data transmission rate is limited for trains moving at extremely high speeds because of the Doppler effect [4]. With hundreds of passengers onboard and an ever-growing data-intensive service demand such as audio/video clip downloading and bulk data retrieval, high information traffic congestion in the cellular network is inevitable.

An alternative or complementary solution is proposed in [4]–[7], where trackside infostations (or repeaters) are deployed in close vicinity to the rail lines and connected to content servers in the Internet. Powerful antennas are installed on each train to communicate with the infostations. The antennas are further connected to a vehicle station which can be accessed by the passenger devices based on wireless local

area network (WLAN) technologies. Various medium access control (MAC) protocols are studied for the communications between the infostations and vehicle stations. For instance, the IEEE 802.11p MAC can be used for video broadcasting in metro passenger information systems [7], while the MAC frame structure proposed in [4] can support data delivery to a high-speed train with a speed up to 360 km/h.

In this work, we consider a cellular/infostation integrated network architecture to better utilize the resources of both network infrastructures [8] [9]. Data services are provided in an on-demand manner. A cellular network with seamless coverage is considered to support control channels for service requests and acknowledgements to minimize their delay and avoid congestion, while data traffic is delivered via trackside infostations to achieve a high data transmission rate. For a large number of onboard passengers, the resource contention among multiple services should be resolved. Further, the coverage provided by the infostations may not be seamless for a low deployment cost. As a high-speed train travels along a rail line, the wireless link from an infostation to a vehicle station is highly dynamic and subject to periodic disconnections, which makes the resource allocation challenging.

In the literature, the optimal on-demand broadcast scheduling is investigated for satellite and cellular networks [10] [11]. The proposed algorithms can resolve the resource contention among multiple services. However, the approaches based on a constant rate broadcast link are not applicable to data delivery via infostations. On the other hand, the data delivery in a vehicular network with intermittent links is studied based on the mobility patterns of vehicles [8] [12]. Service pre-downloading approaches are proposed to reduce the data fetching delay when the link from the backbone network to an infostation is a bottleneck [5] [13]. The solutions deal with single service delivery for each vehicle [5] [8] [12] or offline resource allocation based on service popularity [13], which cannot be directly applied to on-demand data delivery to mass transportation vehicles such as high-speed trains. A service scheduling problem is discussed in [1] under the assumption that the bandwidth from the backbone network to an infostation is sufficiently large, while the service pre-downloading is not addressed. How to efficiently allocate resources among multiple on-demand data services in such a network with intermittent connectivity is an open issue.

In this paper, we investigate the resource allocation for on-demand data delivery to high-speed trains, taking account of both intermittent network connectivity and multi-service demands. Specifically, the contribution of this paper is three-fold: i) The optimal resource allocation problem is formulated based

Manuscript received 15 May 2011; revised 25 October 2011. This research is supported by a research grant from the Natural Science and Engineering Research Council (NSERC). This work is presented in part at IEEE Globecom 2011 [1].

The authors are with the Department of Electrical and Computer Engineering, University of Waterloo, 200 University Avenue West, Waterloo, Ontario, Canada N2L 3G1 (e-mail: {h8liang, wzhuang}@uwaterloo.ca).

Digital object identifier XXXXXX

TABLE I: Summary of important symbols used.

Symbol	Definition
$A_{h,k}$	The capacity of the k th frame within the h th infostation
B	The size of each data block
c_n	The partitioning point of virtual period
$G_s (D_s)$	The request time (deadline) of service s
H	The number of trackside infostations
K_h	The number of frames within the h th infostation
$M_{h,s} (M_{h,s}^d)$	The number of blocks to be (already) pre-downloaded at the h th infostation for service s
N	The number of virtual periods
Q_s	The size of service s
$Q_{h,k,s}^r (\bar{Q}_{h,k,s}^r)$	The numbers of remaining blocks of service s before (after) the resource allocation for the k th frame within the h th infostation
$Q_{\hat{s},s}^r$	The numbers of remaining blocks of service s when service \hat{s} is requested
S	The set of on-demand data services
T_F	Frame duration
$T_I (T_O)$	The starting (ending) time of a trip
$T_h^i (T_h^o)$	The time for a train to come into (go out of) the transmission range of the h th infostation
W_h	The bandwidth of the link from the backbone network to the h th infostation
$x_{h,k,s}$	The number of blocks delivered to the vehicle station during the k th frame within the h th infostation for service s
$y_{n,s}$	The number of blocks delivered to the vehicle station within the n th virtual period for service s
ω_s	The reward of service s

on the trajectory of a train, data service demands, and network resources. In order to achieve efficient resource allocation with low computational complexity, the original frame-based formulation is transformed into a capacity-based formulation, which is a single-machine preemptive scheduling problem with integer request times, processing times, and deadlines. The transformation is based on a time-capacity mapping which exploits the predetermined high-speed train schedule; ii) An online resource allocation algorithm is proposed to address the uncertainties in service demands, and the performance bound is characterized based on the theory of sequencing and scheduling. Given the link from the backbone network to an infostation is a bottleneck, we analyze the pre-downloading capacity and propose a service pre-downloading algorithm to facilitate the resource allocation; iii) The performance of our proposed algorithms is evaluated based on a real high-speed train schedule. It is shown that our proposed resource allocation algorithms can significantly improve the total reward of delivered services as compared with existing algorithms. By tuning a redundant factor, different tradeoff can be achieved between the overhead and reward of service pre-downloading. As many symbols are used in this paper, Table I summarizes the important ones. Proofs of Theorem 1 and all the Lemmas are given in Appendix.

II. SYSTEM MODEL

The network topology is shown in Fig. 1. Several trackside infostations are deployed along the rail line, whereas a cellular network provides a seamless coverage over the region. The

base stations of the cellular network and the infostations are connected to the content servers in the Internet via wireline links¹. When a passenger requests an on-demand data service, the service request is sent from the vehicle station to the corresponding content server via the cellular network. The data traffic of the requested service is delivered from the content server to the vehicle station via the infostations. After the service is delivered, an acknowledgement is made by the vehicle station via the cellular network. For simplicity, we assume that the probability of traffic congestion in the cellular and backbone networks is low, so that the service requests and acknowledgements can be delivered with a negligible delay. Moreover, the data transmission rate from a vehicle station to a passenger device is sufficiently large. A data block can be successfully delivered to a passenger device if it is delivered to the vehicle station.

For the communications between the infostations and the vehicle station, we consider the MAC frame structure proposed in [4] which is specifically designed for high-speed trains with a speed up to 360 km/h. Time is partitioned into frames with equal duration T_F . At the beginning of each frame, one of the powerful antennas which are installed on the train is selected as the master antenna to broadcast a beacon signal to the infostations in the vicinity. The infostations which can detect the beacon signal transmit their unique identification signals as acknowledgments. Each of the antennas on the train uses the acknowledgements for channel estimation and tunes to the

¹Note that a wireless link is possible for an infostation with two sets of wireless transceivers [6].

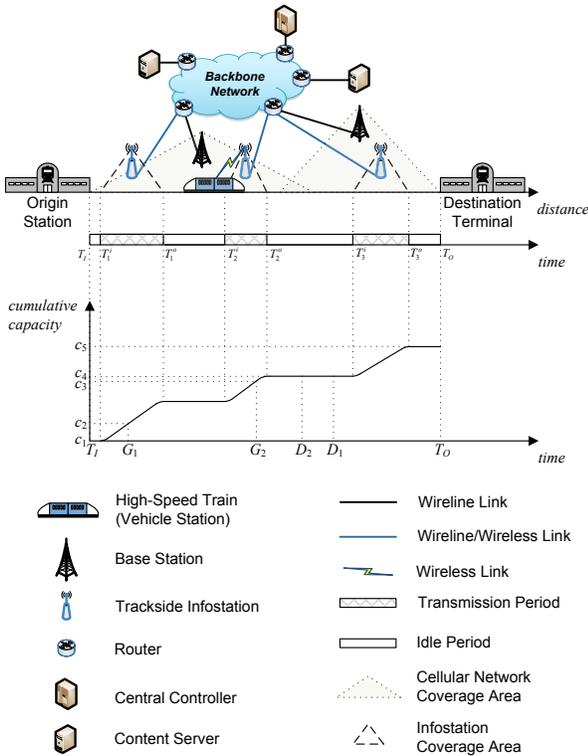


Fig. 1: System model and time-capacity mapping.

infostation with the highest link gain. Then all the infostations which have detected the beacon signal start to broadcast data blocks. This scheme is referred to as the blind information raining. If a group of infostations are deployed in close vicinity with overlapped coverage area, an additional zone controller [4] should be deployed to control the group of infostations and schedule the broadcasting to reduce interference and improve data throughput. In this paper, we mainly focus on a network with isolated infostations and intermittent link connectivity. However, the analytical model can be directly extended to a network with some densely deployed infostations, by replacing each infostation in the current model with a zone controller to take charge of scheduling the group of infostations in close vicinity.

A central controller is deployed and can communicate with the cellular network, infostations, and content servers. The central controller allocates the network radio resources based on the train trajectory and data service demands. The train trajectory defines the location of a train at a specific time, while the radio resources depend on the wireless channel condition from an infostation to a vehicle station. Since each train moves on a predetermined rail line and the schedule of a high-speed train is highly stable², the information of train trajectory and network resources can be obtained by the central controller in advance with high accuracy. However, the demand of a data service is not known a priori until the service request is received by the content server and delivered to the central controller.

²According to a recent report, the accuracy of train departure times of Huhang high-speed railway (also known as the Shanghai-Hangzhou high-speed railway) is about 99.5% [14] [15].

A. Train Trajectory

Consider a single trip of a train from an origin station to a destination terminal within the time duration $[T_I, T_O]$. A total number of H trackside infostations are deployed along the rail line. For instance, we have $H = 3$ in Fig. 1. Each infostation covers a segment of the rail line based on its wireless transmission range. Denote T_h^i and T_h^o as time instants for the train to come into and go out of the transmission range of the h th ($h \in [1, \dots, H]$) infostation, respectively. For isolated infostations, we have $T_h^o \leq T_{h+1}^i$ for $1 \leq h \leq H-1$. Taking into account the duration of a trip, we have $T_I \leq T_1^i$ and $T_H^o \leq T_O$. In Fig. 1, the transmission period and idle period are the time durations when the train is in and out of the coverage area of an infostation, respectively.

B. Data Service Demands

A set S of on-demand data services are supported over the trip. The request of service s ($s \in S$) is received by the content server at time G_s . If service s is delivered to the vehicle station before its deadline D_s , a reward ω_s can be obtained by the service provider. We consider $G_s \geq T_I$ and $D_s \leq T_O$, assuming that all other services can be delivered to the passengers when they are off-board. Erasure coding based service delivery is considered [4] [13]. The information data of service s is encoded and segmented into a large number \tilde{Q}_s of blocks, each having an equal size of B bits. Service s can be decoded when at least Q_s ($Q_s < \tilde{Q}_s$) distinct blocks are received. The advantage of using erasure coding is that no recovery scheme is required for the transmission error or loss of a specific block. The infostations only need to keep transmitting (or “raining” according to [4]) the encoded blocks until the service can be decoded at the vehicle station, which significantly simplifies the protocol design for high-speed train applications subject to a highly dynamic wireless channel condition.

C. Network Resources

The duration that the train is within the coverage of the h th infostation corresponds to a number of frames, given by $K_h = \lfloor (T_h^o - T_h^i)/T_F \rfloor$. Note that the small difference between T_h^i and the beginning time of the first frame is omitted. The k th frame begins and ends at times $T_h^i + (k-1)T_F$ and $T_h^i + kT_F$, respectively. We define the capacity $(A_{h,k})$ of the k th frame as the maximum number of blocks that can be delivered from the h th infostation to the vehicle station within this frame. The value of $A_{h,k}$ is determined by the wireless channel condition according to (3) in [4]. If the wireless channel is in deep fading such that no block can be delivered to the vehicle station, we have $A_{h,k} = 0$. A round-robin scheduler³ is applied when multiple trains are present in the coverage area of an infostation. If the k th frame within the h th infostation coverage is not allocated to the vehicle station

³A scheduler with prioritization can potentially improve the performance of resource allocation when multiple trains are simultaneously within the coverage of an infostation. However, as the time for two high-speed trains to meet is extremely short according to the real train schedule (e.g., the one considered in Section VI), the performance improvement can be very limited.

under consideration, we have $A_{h,k} = 0$. Since the MAC is frame based, service request time (or deadline) is rounded to the beginning time of a frame.

Full-duplex infostations are considered such that data fetching from the content server and data delivery to the vehicle station can be achieved simultaneously. Two cases are considered for the link from the backbone network to an infostation: 1) The bandwidth of the link (e.g., a high data-rate wireline link [4]) is sufficiently large so that the capacity of each frame can be fully utilized; 2) The link (e.g., a T1 based wireline link at 1.5Mbps [16] or an IEEE 802.16j based wireless link [6]) is a bottleneck with limited bandwidth W_h to the h th infostation. For the second case, data services can be pre-downloaded at the infostations to achieve high radio resource utilization [8]. We consider an unlimited buffer space at the infostations.

III. PROBLEM FORMULATION AND TRANSFORMATION

In this section, we first formulate the optimal resource allocation problem. Then we introduce a time-capacity mapping to transform the original problem formulation into a capacity-based problem formulation.

A. Problem Formulation

The objective in service provisioning is to maximize the total reward of delivered services over a trip of the train. Define $x_{h,k,s}$ as the number of blocks delivered to the vehicle station during the k th frame within the h th infostation coverage for service s . The resource allocation variable over the trip of the train is given by $X = \{x_{h,k,s} | h \in \{1, 2, \dots, H\}, k \in \{1, 2, \dots, K_h\}, s \in S\}$. For a specific X , define $\psi_{X,s}$ as a delivery indicator of service s , which equals 1 if service s is delivered before its deadline and 0 otherwise. Based on erasure coding, we have $\psi_{X,s} = 1$ if $\sum_{h=1}^H \sum_{k=1}^{K_h} x_{h,k,s} = Q_s$, and $\psi_{X,s} = 0$ if $\sum_{h=1}^H \sum_{k=1}^{K_h} x_{h,k,s} < Q_s$. Note that we do not consider the case $\sum_{h=1}^H \sum_{k=1}^{K_h} x_{h,k,s} > Q_s$ because, for erasure coding based service delivery, the resources are underutilized by delivering more than Q_s blocks for service s . The optimal resource allocation problem is formulated as

$$(\mathbf{P1}) \max_X \sum_{s \in S} \omega_s \psi_{X,s} \quad (1)$$

$$\text{subject to} \quad x_{h,k,s} \in \mathbb{Z}^+, \quad h \in \{1, 2, \dots, H\}, s \in S \\ k \in \{1, 2, \dots, K_h\} \quad (2)$$

$$\sum_{h=1}^H \sum_{k=1}^{K_h} x_{h,k,s} \leq Q_s, \quad s \in S \quad (3)$$

$$x_{h,k,s} = 0, \quad \text{if } G_s \geq T_h^i + kT_F \\ \text{or } D_s \leq T_h^i + (k-1)T_F, \\ h \in \{1, 2, \dots, H\}, s \in S \\ k \in \{1, 2, \dots, K_h\} \quad (4)$$

$$\sum_{s \in S} x_{h,k,s} \leq A_{h,k}, \quad h \in \{1, 2, \dots, H\}, \\ k \in \{1, 2, \dots, K_h\}. \quad (5)$$

Constraint (2) implies that negative resource allocation is not allowed, where $\mathbb{Z}^+ = \mathbb{N} \cup \{0\}$ represents the set of

nonnegative integers. Constraint (4) states that the blocks of a service can only be delivered after the request time and before the deadline. With (5), the number of blocks that can be delivered to the vehicle station during the k th frame within the h th infostation coverage is limited by the capacity $A_{h,k}$ of the frame.

Problem P1 is a mixed integer programming (MIP) problem which cannot be solved efficiently [17]. The main difficulty of analyzing problem P1 comes from the integer nature of constraint (2). However, based on further investigation, we observe that problem P1 can be potentially considered as a problem of sequencing and scheduling [18] because of the constant request time, deadline, size (in terms of the number of blocks), and reward of each service. However, the existing theory of sequencing and scheduling cannot be directly applied to analyze problem P1 since the services are not schedulable continuously over time because of the intermittent link connectivity. Moreover, the data transmission rate (in terms of $A_{h,k}$) from infostations to a vehicle station is not a constant. Therefore, the duration to complete each service is dependent on the time when the service is requested and scheduled.

In order to better characterize problem P1 and develop efficient algorithms to solve it, we consider a problem transformation in the rest of this section such that the time indices are virtually mapped to cumulative capacity values, as shown in Fig. 1. Here we define the cumulative capacity at time t ($t \in [T_I, T_O]$) as the summation of the capacities of all frames within $[T_I, t]$. The problem transformation consists of two steps, i.e., time-capacity mapping and capacity-based problem formulation as presented in the following. Here problem P1 is referred to as the frame-based problem formulation.

B. Time-Capacity Mapping

For a specific trip of the train, the maximum number of blocks that can be delivered to the vehicle station is limited by $\sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k}$. Define a time-capacity mapping function $f(t) : [T_I, T_O] \rightarrow [0, 1, \dots, \sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k}]$ which maps time t to the corresponding cumulative capacity. Based on the information of train trajectory and network radio resources, we have the following lemma.

Lemma 1. *The value of $f(t)$ is given by*

$$f(t) = \begin{cases} \sum_{j=1}^{\lfloor (t-T_{h_t}^i)/T_F \rfloor} A_{h_t,j} + \sum_{l=1}^{h_t-1} \sum_{j=1}^{K_l} A_{l,j}, & \text{if } h_t \geq 1 \text{ and } t \leq T_{h_t}^o \\ \sum_{l=1}^{h_t} \sum_{j=1}^{K_l} A_{l,j}, & \text{otherwise} \end{cases} \quad (6)$$

where $h_t = \arg \max_h \{T_h^i \leq t\}$ if $t \geq T_1^i$, and $h_t = 0$ otherwise. Without loss of generality, we consider the summation $\sum_{l=a}^b (\cdot)$ equals zero if $b < a$.

Intuitively, more blocks can be potentially delivered to the vehicle station as time t increases. This property is inherent for $f(t)$ and stated by the following lemma.

Lemma 2. *The mapping $f(t)$ is a non-decreasing function with respect to t ($t \in [T_I, T_O]$).*

In problem P1, only constraint (4) is directly related to the time indices. Therefore, we can apply the time-capacity mapping function $f(t)$ on constraint (4) to transform it into a capacity-based constraint. Based on Lemma 1 and Lemma 2, the following theorem holds.

Theorem 1. For problem P1, (4) is equivalent to a capacity-based constraint given by

$$\begin{aligned} x_{h,k,s} &= 0, \text{ if } G_s^c \geq \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j} \\ \text{or } D_s^c &\leq \sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}, \quad s \in S \\ h &\in \{1, 2, \dots, H\}, k \in \{1, 2, \dots, K_h\} \end{aligned} \quad (7)$$

where $G_s^c = f(G_s)$ and $D_s^c = f(D_s)$.

In (7), G_s^c and D_s^c can be considered as the virtual request time and deadline of service s , respectively, which are defined based on the cumulative capacity.

C. Capacity-Based Problem Formulation

By replacing (4) in problem P1 with (7), we can obtain problem P2. Since all constraints of problem P2 are defined based on the number of blocks, we can simplify problem P2 by introducing a capacity-based formulation.

By definition, we have $f(T_I) = 0$ and $f(T_O) = \sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k}$. Then the set $\{T_I, T_O, G_s, D_s | s \in S\}$ of time indices can be represented by a set C of unique cumulative capacity, given by

$$\begin{aligned} C &= \cup_{s \in S} \{f(G_s), f(D_s)\} \cup \{f(T_I), f(T_O)\} \\ &= \cup_{s \in S} \{G_s^c, D_s^c\} \cup \left\{ 0, \sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k} \right\}. \end{aligned} \quad (8)$$

Let $|C| = N + 1$ ($N \geq 1$) and c_n ($1 \leq n \leq N + 1$) be the cardinality and elements of set C , respectively. Without loss of generality, we consider an ascending order of the elements in C , i.e., $c_1 < c_2 < \dots < c_{N+1}$. An example is shown in Fig. 1, where two services are considered with request times G_1 and G_2 , and deadlines D_1 and D_2 , respectively. Then we have $N = 4$ and c_n ($1 \leq n \leq 5$) given by

$$\begin{aligned} c_1 &= 0, \quad c_2 = G_1^c, \quad c_3 = G_2^c, \\ c_4 &= D_1^c = D_2^c, \quad c_5 = \sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k} \end{aligned} \quad (9)$$

where D_1 and D_2 are mapped to the same cumulative capacity c_4 since no block can be delivered during an idle period. Note that if $G_2^c < D_1^c \neq D_2^c < \sum_{h=1}^H \sum_{k=1}^{K_h} A_{h,k}$, we have $N = 5$, while each element in C (other than c_1 and c_6) corresponds to the request time or deadline of a service.

We partition the trip of the train into N non-overlapped virtual periods according to the cumulative capacity values in C . Within the n th virtual period (defined by $[c_n + 1, c_{n+1}]$), no new service is requested and no existing service expires since all service request times and deadlines are considered in

the calculation of set C . Therefore, for a feasible resource allocation, changing the sequence of service scheduling within a virtual period does not affect the service delivery performance. This property is formally stated by the following lemma.

Lemma 3. Consider a feasible resource allocation variable X with four elements $x_{h_1, k_1, s_1}, x_{h_1, k_1, s_2}, x_{h_2, k_2, s_1}, x_{h_2, k_2, s_2}$. Suppose $x_{h_1, k_1, s_1}, x_{h_2, k_2, s_2} \geq 1, x_{h_1, k_1, s_2}, x_{h_2, k_2, s_1} \geq 0$. All blocks of the two frames (i.e., the k_1 th frame within the h_1 th infostation coverage and the k_2 th frame within the h_2 th infostation coverage) belong to the same virtual period, while the two frames are not identical. Construct another resource allocation variable X' by replacing the elements $x_{h_1, k_1, s_1}, x_{h_1, k_1, s_2}, x_{h_2, k_2, s_1}, x_{h_2, k_2, s_2}$ in X with $x_{h_1, k_1, s_1} - 1, x_{h_1, k_1, s_2} + 1, x_{h_2, k_2, s_1} + 1, x_{h_2, k_2, s_2} - 1$ and keeping all other elements unchanged. Then we have the same feasibilities and objective function values for X and X' .

Based on Lemma 3, the optimal resource allocation can be achieved by considering the total number of blocks delivered for each service within each virtual period. Define $y_{n,s}$ as the number of blocks delivered to the vehicle station for service s within the n th virtual period. Then the resource allocation variable over the trip of the train is given by $Y = \{y_{n,s} | n \in \{1, 2, \dots, N\}, s \in S\}$. Define the delivery indicator of service s as $\eta_{Y,s}$. We have $\eta_{Y,s} = 1$ if $\sum_{n=1}^N y_{n,s} = Q_s$, and $\eta_{Y,s} = 0$ if $\sum_{n=1}^N y_{n,s} < Q_s$. Then problem P2 can be transformed into a capacity-based formulation as follows

$$(P3) \quad \max_Y \quad \sum_{s \in S} \omega_s \eta_{Y,s} \quad (10)$$

$$\text{subject to} \quad y_{n,s} \in \mathbb{Z}^+, \quad n \in \{1, 2, \dots, N\}, s \in S \quad (11)$$

$$\sum_{n=1}^N y_{n,s} \leq Q_s, \quad s \in S \quad (12)$$

$$y_{n,s} = 0, \text{ if } G_s^c \geq c_{n+1} + 1 \text{ or } D_s^c \leq c_n, \\ n \in \{1, 2, \dots, N\} \quad (13)$$

$$\sum_{s \in S} y_{n,s} \leq c_{n+1} - c_n, \quad n \in \{1, 2, \dots, N\} \quad (14)$$

where (14) states that the number of blocks which can be delivered to the vehicle station during the n th virtual period is limited to $c_{n+1} - c_n$.

IV. RESOURCE ALLOCATION

Based on the problem transformation, time indices are virtually transformed to the cumulative capacity values, over which the services are continuously schedulable. According to the theory of sequencing and scheduling, problem P3 defines a single-machine preemptive scheduling problem with integer request (or release) times (G_s^c), processing times (Q_s), and deadlines (D_s^c) (formal notation: $1|G_s^c, \text{preemption}|\sum \omega_s \eta_{Y,s}$), which can be solved by a dynamic programming algorithm at complexity $O(|S|L_G^2 L_\omega^2)$, where L_G is the number of distinct request times, and L_ω represents the sum of the integer reward [18]–[21]. The complexity is known as pseudo-polynomial [21] since the representation of all rewards ω_s ($s \in S$) by integers may result in a large L_ω and a high computational complexity accordingly. Moreover,

for on-demand data services, the service demands are not known a priori. In order to achieve efficient resource allocation for on-demand data service delivery to high-speed trains, we devise an online algorithm in this section. As the online algorithm is devised based on problem P3, its performance bound can be characterized based on the theoretical results of the single-machine preemptive scheduling problem, to be discussed in the following.

As the train moves from the origin station to the destination terminal, the online algorithm allocates the network resources to multiple services frame-by-frame. Consider the k th frame within the h th infostation coverage, with $x_{h,k,s}$ blocks delivered for service s ($s \in S_{h,k}^g$), where $S_{h,k}^g = \{s | s \in S, G_s \leq T_h^i + (k-1)T_F\}$ represents the set of requested services. The resource allocation algorithm is detailed in Algorithm 1, where $Q_{h,k,s}^r$ and $\tilde{Q}_{h,k,s}^r$ are the numbers of remaining blocks of service s before and after the k th frame, respectively. The algorithm needs to be performed only when $S_{h,k}^g \neq \emptyset$. For a newly requested service s , i.e.,

$$s \in \begin{cases} S_{h,k}^g, & \text{if } h = 1, k = 1 \\ S_{h,k}^g \setminus S_{h-1, K_{h-1}}, & \text{if } h \neq 1, k = 1 \\ S_{h,k}^g \setminus S_{h,k-1}^g, & \text{otherwise} \end{cases} \quad (15)$$

we have $Q_{h,k,s}^r = Q_s$; Otherwise, we have

$$Q_{h,k,s}^r = \begin{cases} \tilde{Q}_{h-1, K_{h-1}, s}^r, & \text{if } k = 1 \\ \tilde{Q}_{h,k-1, s}^r, & \text{if } k > 1. \end{cases} \quad (16)$$

In (15) and (16), $k = 1$ corresponds to the first frame within an infostation coverage. In (16), if $k = 1$, we have $h \neq 1$ since the service is considered to be new in one of the previous frames.

Algorithm 1 iteratively allocates the capacity of a frame ($A_{h,k}$) to the on-demand data services in descending order of their utilities, until the capacity of the frame is fully utilized. In step 3, S_A represents the set of active services which can possibly be delivered before their deadlines, given by

$$S_A = \left\{ s | s \in S_{h,k}^g, \tilde{Q}_{h,k,s}^r > 0, D_s^c \geq \tilde{Q}_{h,k,s}^r + \sum_{j=1}^k A_{h,j}, \right. \\ \left. + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j} - m \right\}. \quad (17)$$

In step 7, U_s represents the utility of service s . We consider two kinds of utilities, i.e., Smith ratio and exponential capacity [20]. For Smith ratio based algorithm, $U_s = \omega_s / Q_s$. Intuitively, a service with a higher reward or smaller size can obtain a higher utility. For the exponential capacity based algorithm, the utility function incorporates the number of remaining blocks ($\tilde{Q}_{h,k,s}^r$) which corresponds to the current condition of each service, and is given by

$$U_s = \omega_s \left(1 - \frac{\ln(\max_{s \in S_{h,k}^g} Q_s)}{\max_{s \in S_{h,k}^g} Q_s} \right)^{\tilde{Q}_{h,k,s}^r - 1}. \quad (18)$$

The complexity of Algorithm 1 is $O(\max_{h,k} \{A_{h,k}\} |S|)$.

Define the competitive ratio of Algorithm 1 as the maximal

Algorithm 1 Resource Allocation Algorithm

Input: $k, h, G_s^c, D_s^c, \omega_s, Q_s, Q_{h,k,s}^r$ ($s \in S_{h,k}^g$)

Output: $x_{h,k,s}, \tilde{Q}_{h,k,s}^r$ ($s \in S_{h,k}^g$)

- 1: Initialize $x_{h,k,s} = 0, \tilde{Q}_{h,k,s}^r = Q_{h,k,s}^r$ for $s \in S_{h,k}^g$,
 $m = A_{h,k}$;
 - 2: **while** $m \neq 0$ **do**
 - 3: S_A calculation;
 - 4: **if** $S_A = \emptyset$ **then**
 - 5: **break**;
 - 6: **end if**
 - 7: U_s calculation, for $s \in S_A$;
 - 8: $s^* = \arg \max_{s \in S_A} \{U_s\}$;
 - 9: Update $x_{h,k,s^*} \leftarrow x_{h,k,s^*} + 1, \tilde{Q}_{h,k,s^*}^r \leftarrow \tilde{Q}_{h,k,s^*}^r - 1,$
 $m \leftarrow m - 1$;
 - 10: **end while**
-

ratio (corresponds to the worst-case performance of Algorithm 1 with respect to the randomness in service requests) of the total reward of delivered services based on the optimal solution of problem P3 to that of the delivered services based on Algorithm 1. The competitive ratio of the Smith ratio and exponential capacity based algorithm is given by $2 \max_{s \in S} \{Q_s\}$ and $\max_{s \in S} \{Q_s\} / \ln(\max_{s \in S} \{Q_s\})$, respectively [20]. Since we have $1 / \ln(\max_{s \in S} \{Q_s\}) < 2$ for typical on-demand data services which consist of a large number of blocks, the exponential capacity based algorithm can improve the performance of resource allocation in worst-case scenarios, at the cost of higher computational complexity in step 7. However, the computational complexity can be potentially reduced. For instance, a straightforward approach is to maintain a queue of all active services and sort the services in descending order of their utilities. In this way, the head-of-line (HOL) service always represents the service with the highest utility to be scheduled. For the resource allocation in each frame, the order is updated only for the newly requested services or the services with some blocks being delivered. As a result, step 7 and step 8 do not need to be recalculated for each active service during each iteration with respect to m .

V. SERVICE PRE-DOWNLOADING

When the link from the backbone network to an infostation is the bottleneck of service delivery, the capacity $A_{h,k}$ of a frame is underutilized if less than $A_{h,k}$ blocks are fetched from the content server within the frame. In order to address this problem, a service pre-downloading mechanism can be implemented [5], [8], [13]. In the cellular/infostation integrated network, after a service request is received by the content server, the data blocks of the service can be pre-downloaded to the infostations to be visited by the vehicle station, and then delivered to the vehicle station upon its arrival. A simple service pre-downloading approach is to buffer all data blocks of the available services at each infostation. However, this approach is not only infeasible (because of the limited bandwidth of the bottleneck link) but also inefficient (as some pre-downloaded blocks cannot be transmitted to the vehicle station during its short visit to each of the infostations).

In the following, we propose a service pre-downloading approach to facilitate the resource allocation of Algorithm 1. Let $S_s^g = \{s | s \in S, G_s \leq G_s^g\}$ denote the set of requested services at time G_s^g . We want to determine the number of blocks ($M_{h,s}$) to be pre-downloaded at the h th ($h \in [1, \dots, H]$) infostation for service $s \in S_s^g$. Next, we first analyze the pre-downloading capacity to determine the maximum number of blocks to be pre-downloaded at each infostation, and then present a service pre-downloading algorithm to calculate $M_{h,s}$.

A. Pre-Downloading Capacity and Redundant Factor

Taking account of the limited capacity of each frame, the number of blocks to be pre-downloaded at the h th infostation ($\sum_{s \in S} M_{h,s}$) should be limited by the sum capacity of all frames within the infostation coverage ($\sum_{k \in K_h} A_{h,k}$). On the other hand, for a given time t ($t < T_h^o$), the duration of service pre-downloading to the h th infostation is given by $T_h^o - t$. Note that the transmission period with duration $T_h^o - T_h^i$ is taken into account since full-duplex infostations are considered. With the bandwidth W_h of the bottleneck link, the maximum number of blocks that can be fetched from the content server during $T_h^o - t$ is $\lfloor (T_h^o - t)W_h/B \rfloor$. Then the pre-downloading capacity of the h th infostation at time t is

$$Q_{h,t}^c = \min \left\{ \sum_{k \in K_h} A_{h,k}, \lfloor (T_h^o - t)W_h/B \rfloor \right\}. \quad (19)$$

As discussed in Section II, service s can be successfully decoded if Q_s blocks are received by the vehicle station before D_s . However, because of the resource contention among multiple services, the number of pre-downloaded blocks of service s is dependent on other service demands. Therefore, we introduce a redundant factor β ($\beta \geq 0$) for service pre-downloading. Specifically, for each service, the number of pre-downloaded blocks at the infostations to be visited is β times the number of remaining blocks to be delivered. The larger the β , the more blocks can be pre-downloaded for each service and the less number of services can be pre-downloaded. Note that β can be larger than one since some pre-downloaded blocks of a service may not be delivered to the vehicle station when new services with high priorities arrive.

B. Service Pre-Downloading Algorithm

Based on the pre-downloading capacity and redundant factor, a service pre-downloading algorithm is devised. When the request of a service \hat{s} is received at time $G_{\hat{s}}$, the service pre-downloading variables ($M_{h,s}$) are calculated according to Algorithm 2, where $Q_{\hat{s},s}^r$ represents the number of remaining blocks of service s at time $G_{\hat{s}}$. In step 5, only the set of active services which can possibly be delivered before their deadlines is considered. The minimum operation in step 9 is performed over two terms corresponding to the pre-downloading capacity and redundant factor, respectively. For the first term, a summation $\sum_{i=1}^{j-1} M_{h,s_i}$ is subtracted since the corresponding capacity is used to pre-download services with higher priorities, while for the second term, a summation $\sum_{l=h_{G_{\hat{s}}+1}}^{h-1} M_{l,s_j}$ is subtracted since this amount of blocks is

Algorithm 2 Service Pre-Downloading Algorithm

Input: $G_s, D_s, G_s^c, D_s^c, \omega_s, Q_s, Q_{\hat{s},s}^r$ ($s \in S_s^g$)
Output: $M_{h,s}$ ($h \in \{1, \dots, H\}$, $s \in S_s^g$)

- 1: Initialize $M_{h,s} = 0$ for $h \in \{1, \dots, H\}$, $s \in S_s^g$;
- 2: U_s calculation, for $s \in S_s^g$;
- 3: Sort services in S_s^g in descending order of their utilities and obtain the ordered set $\{s_1, s_2, \dots, s_{|S_s^g|}\}$;
- 4: **for** $j = 1$ to $|S_s^g|$ **do**
- 5: **if** $Q_{\hat{s},s_j}^r = 0$ or $D_{s_j}^c < G_{\hat{s}}^c + Q_{\hat{s},s_j}^r$ **then**
- 6: Continue;
- 7: **end if**
- 8: **for** $h = h_{G_{\hat{s}}} + 1$ to H **do**
- 9: $M_{h,s_j} = \min \left\{ Q_{h,G_{\hat{s}}}^c - \sum_{i=1}^{j-1} M_{h,s_i}, \beta Q_{\hat{s},s_j}^r - \sum_{l=h_{G_{\hat{s}}+1}}^{h-1} M_{l,s_j} \right\}$;
- 10: **end for**
- 11: **end for**

to be pre-downloaded to the infostations before infostation h . The utility (U_s) of service s is given by the resource allocation Algorithm 1. The complexity of Algorithm 2 is $O(H|S|)$.

Let $M_{h,s}^d$ denote the number of blocks already pre-downloaded at infostation h for service s . Because of the arrival of services with higher priorities, we may have $M_{h,s} < M_{h,s}^d$. Therefore, the number of blocks to be pre-downloaded for service s at infostation h is given by $\max\{0, M_{h,s} - M_{h,s}^d\}$. For each infostation, the data blocks of the services are fetched from the content server in descending order of their utilities. When multiple trains are traveling on parallel high-speed rails between the origin station and destination terminal [14], Algorithm 2 is applied by letting S_s^g represent the set of services requested by all the trains.

When a train comes into the transmission range of an infostation, the pre-downloaded blocks are scheduled for transmission according to Algorithm 1. After all pre-downloaded blocks are delivered, the remaining blocks of available services are directly fetched from the content servers.

VI. NUMERICAL RESULTS

In order to evaluate performance of the proposed resource allocation algorithms, we consider a real train schedule based on the Huhang high-speed railway [22]. The railway is specially designed for high-speed trains with a maximum speed of 350 km/h. There are ten stations on the railway and the location of each station (in terms of the distance from the Shanghai station) is given in Table II.

Since no mobility trace is available, we consider a synthetic train mobility model proposed in [23]. Each train moves at a constant speed when it travels from one station to another. When a train leaves (arrives at) a station, it accelerates (decelerates) according to a constant acceleration (deceleration). For simplicity, we consider the deceleration equals to the negative value of the acceleration (α). A typical value for the acceleration α of a high-speed train is given by 0.4 m/s² [24]. Five sample trajectories of the high-speed trains are shown in Fig. 2, for two trains from Hangzhou (G7403 and G7302) and three trains from Shanghai (G7401, G7301, and G7362). The

TABLE II: Locations of stations on the railway.

Station	Shanghai	Hongqiao	Songjiang
Distance (km)	0	33	64
Station	Jinshan	Jiashan	Jiaxing
Distance (km)	81	100	117
Station	Tongxiang	Haining	Yuhang
Distance (km)	145	166	177
Station	Hangzhou		
Distance (km)	202		

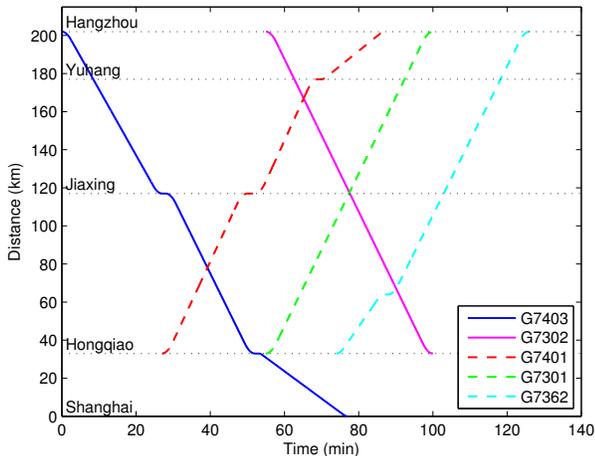


Fig. 2: Trajectory of the high-speed trains.

starting time of train G7403 (06:05 am) based on the February 2011 schedule is chosen to be time 0. Note that the trains G7301 and G7302 need less time to travel between Hongqiao and Hangzhou since they do not stop at two intermediate stations, Jiaxing and Yuhang.

For the wireless channel condition, we use a typical setting for a high-speed train [4], with $T_F = 53\mu s$, $B = 240$ bits, and $H = 40$. The wireless communication between an infostation and the vehicle station is established based on a carrier frequency of 2.4GHz and an approximate data rate of 50Mbps. On each train operating on Huhang high-speed railway, 13 antennas can be deployed with a separation distance of 15 m between adjacent antennas. The distance between each infostation and the rail line is 3 m. The transmission range of each infostation is approximately 500 m. The service requests arrive at a train according to a Poisson process with average rate λ . The number of blocks of each service (Q_s) is uniformly distributed within $[Q_{\min} = 50000, Q_{\max} = 500000]$ (corresponding to a service size within [1.5, 15] Mbytes). The lifetime ($D_s - G_s$) of each service is exponentially distributed with average value 2 minutes. The reward of each service (ω_s) is uniformly distributed within $[1, 10]^4$.

⁴In reality, the reward of each service may depend on many factors such as the service size, urgency (delivery deadline), and priority. How to map these factors to the reward for practical high-speed train applications is still an open issue and left for our future work.

In addition to the proposed resource allocation algorithms, we consider three existing algorithms for comparison, i.e., first-in-first-out (FIFO), earliest due date (EDD), and RAPID [12]. For the FIFO and EDD algorithms, the services are scheduled according to an ascending order of their request times and deadlines, respectively. RAPID is a typical single-service resource allocation algorithm for a network with intermittent links. Since the RAPID algorithm is originally proposed for randomized node mobility, we have modified the algorithm to incorporate the pre-determined train schedule for fair comparison. To calculate the utility function, we modify the algorithm by replacing the time indices with cumulative capacity values based on the time-capacity mapping. Moreover, the transfer opportunity [12] (which determines the maximum number of blocks that can be delivered from an infostation to the vehicle station) is changed from a constant defined by the original work to the sum of the capacity of all frames within each infostation, which is a variable with respect to different infostations according to the train schedule.

A. Performance of Resource Allocation Algorithms

The performance of our proposed resource allocation algorithm is evaluated by extensive simulations under different system parameters, such as service arrival rate, train schedule, service size, and service lifetime. The total reward of delivered services versus average service arrival rate (λ) is shown in Fig. 3 for train G7302. The standard deviations are illustrated for reference. The total reward is low for the FIFO and EDF algorithms since they do not incorporate the train trajectory and data service demands. Although the EDF algorithm performs well when most services can be delivered before their deadlines, its performance degrades as λ increases [25]. For a large λ , the total rewards achieved by the RAPID algorithm and our proposed algorithm improve since more services can be potentially scheduled. However, the increment dwindles since the network throughput becomes saturated. The RAPID algorithm performs better than the FIFO and EDF algorithms since the mobility information of the train is taken into account. By further incorporating the demands of multiple services, our proposed resource allocation algorithm achieves the best performance. In comparison with the existing algorithms, the performance gain achieved by our proposed algorithm improves as the service arrival rate increases, which is a desirable property for high-speed trains with hundreds of passengers onboard and an ever-growing service demand. Although the competitive factor of the exponential capacity based algorithm is higher than that of the Smith ratio based algorithm, the performance of resource allocation is comparable since a worst case scenario for the algorithm (i.e., there is a large service with high reward which consumes most of the network resources [20]) happens with a low probability for the high-speed train applications.

Fig. 4 shows the total reward of delivered services versus λ for train G7401. For the same λ value, train G7401 has a higher total reward than train G7302, because the former has a longer trip duration, as shown in Fig. 2, resulting in more delivered services.

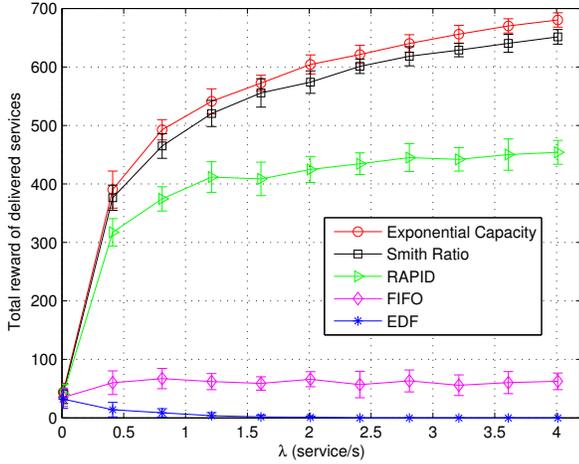


Fig. 3: Impact of service arrival rate (G7302).

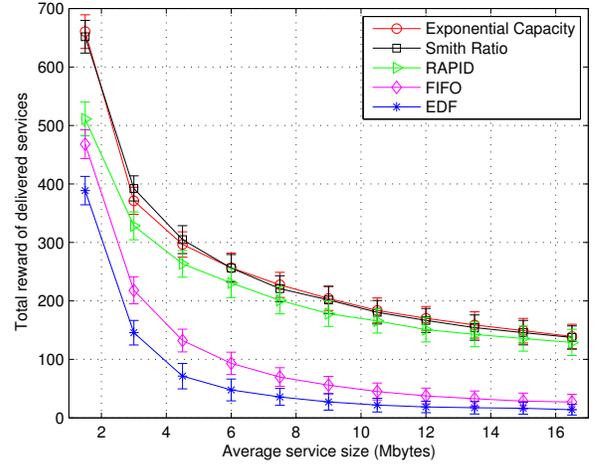


Fig. 5: Impact of service size.

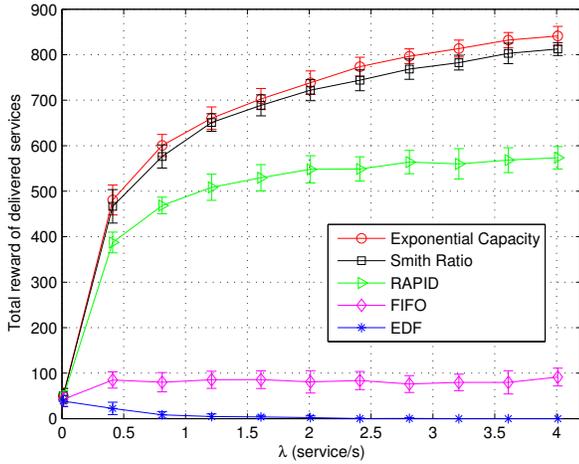


Fig. 4: Impact of service arrival rate (G7401).

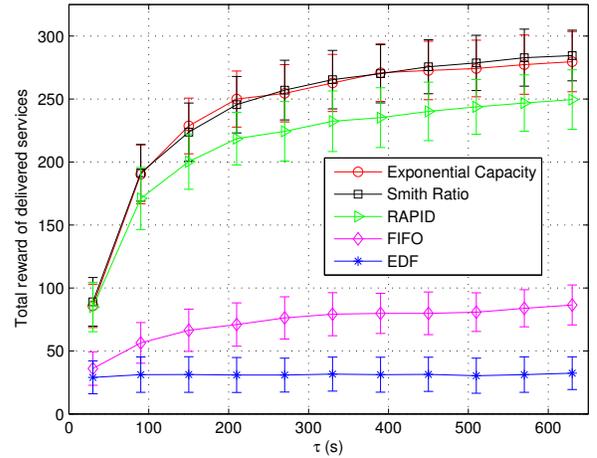


Fig. 6: Impact of service lifetime.

Fig. 5 shows how the total reward of delivered services changes with the average service size, with $Q_{\min} = 50000$ and Q_{\max} varying according to the average. We can see that the total reward decreases as the average downloading file size increases. For a larger average service size, more resources are needed to deliver each service. As a result, a less number of services can be delivered under the limited resources.

The total reward of delivered services increases with the average service lifetime (τ), as shown in Fig. 6. For a larger τ , more infostations can be visited by a vehicle station before a service expires, which increases the probability of delivering the service. However, the increment dwindles when τ is large since the network throughput becomes saturated. From Figs. 4-6, we can see that our proposed resource allocation algorithms outperform the existing algorithms under different system parameters. As similar performance is observed for Smith ratio and exponential capacity utilities, in the following performance evaluation, we consider the Smith ratio based algorithm as an example.

B. Performance of Service Pre-Downloading Algorithm

The effect of the bottleneck link bandwidth (W_h) in service pre-downloading is shown in Fig. 7 for train G7403, where all infostations have the same bandwidth $W_h = W$ ($h \in [1, \dots, H]$). The total reward is significantly improved by the service pre-downloading. As the bandwidth increases, the total reward improves slowly without service pre-downloading since all data blocks need to be fetched directly from the content servers upon the arrival of a vehicle station, which underutilizes the capacity of the wireless channel (in frames). On the other hand, with service pre-downloading, a higher reward can be achieved even for a small W_h since the disconnected period of a vehicle station is effectively exploited by the infostations to pre-download data blocks.

Fig. 8 and Fig. 9 show that the redundant factor (β) has a critical impact on the resource allocation performance. As demonstrated in Fig. 8, the number of pre-downloaded blocks increases as β increases. However, the increment decreases for a large β because of a saturated throughput of the bottleneck

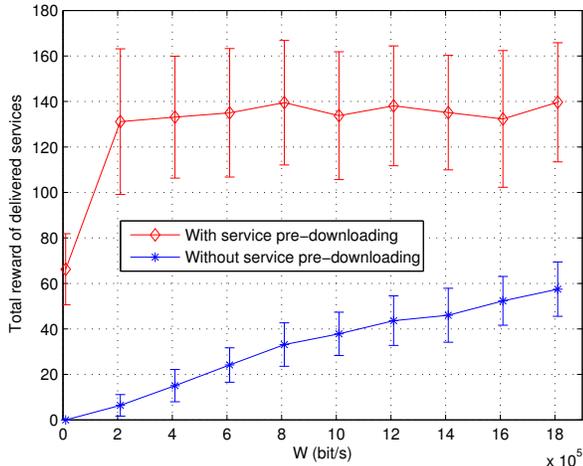


Fig. 7: The total reward with and without service pre-downloading ($\lambda = 0.02$ service/s).

link. Also, the number of pre-downloaded blocks increases with λ . From Fig. 9, we can see that, as β increases, the total reward first increases and then decreases. When β is small, more services can be pre-downloaded at each infostation while the number of blocks pre-downloaded for each service is small. The highest total reward is achieved at $\beta = 0.6$ and $\beta = 1.4$ for $\alpha = 0.05$ service/s and $\alpha = 0.01$ service/s, respectively. This observation indicates that, for a lower service arrival rate, more blocks should be pre-downloaded for each service, and vice versa. However, further investigation is needed to determine the optimal value of β and to strike a balance between the overhead of service pre-downloading and the total reward of delivered services.

VII. CONCLUSIONS

In this paper, we formulate an optimal resource allocation problem for on-demand data delivery to high-speed trains in a cellular/infostation integrated network. The problem is transformed into a single-machine preemptive scheduling problem with integer request times, processing times, and deadlines. An online resource allocation algorithm with the Smith ratio and exponential capacity based utility functions is proposed. The performance bound of the online algorithm is characterized based on the theory of sequencing and scheduling with respect to the single-machine preemptive scheduling problem. Further, a service pre-downloading algorithm is presented to achieve efficient resource allocation when the link from the backbone network to an infostation is a bottleneck. It is demonstrated that the proposed resource allocation algorithm can improve the total reward of delivered services over the existing approaches such as FIFO, EDF, and RAPID, and that the service pre-downloading algorithm can significantly improve the efficiency of resource allocation when the bandwidth of the link to the infostation is a limiting factor. By tuning the redundant factor, different tradeoff can be achieved between the overhead of service pre-downloading (in terms of the number of pre-downloaded blocks) and total reward of delivered services.

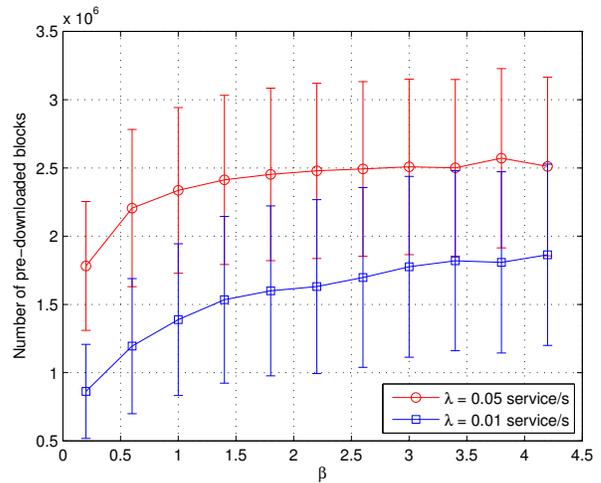


Fig. 8: Number of pre-downloaded blocks versus redundant factor ($W = 10^4$ bit/s).

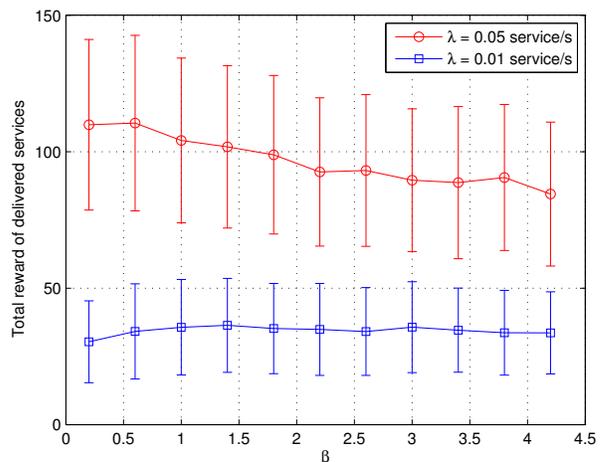


Fig. 9: Total reward of delivered services versus redundant factor ($W = 10^4$ bit/s).

Further work includes a joint formulation of the service pre-downloading problem and resource allocation problem, and the design of a more efficient utility function for the online algorithm. Moreover, for practical high-speed train applications, how to map the quality of service (QoS) parameters such as the service size, relative deadline, and importance/priority to the reward of each service is an interesting topic and left for our future work.

APPENDIX A: PROOF OF LEMMA 1

Two cases are considered for a given t . Case 1: t is in a transmission period (i.e., $\exists h, T_h^i \leq t \leq T_h^o$); Case 2: t is in an idle period (i.e., $\nexists h, T_h^i \leq t \leq T_h^o$).

Case 1: At time t , the infostation with which the vehicle station can communicate is $h_t = \arg \max_h \{T_h^i \leq t\}$. The number of frames within the h_t th infostation coverage before time t is $\lfloor (t - T_{h_t}^i)/T_F \rfloor$, and the sum capacity of these frames is given by $\sum_{j=1}^{\lfloor (t - T_{h_t}^i)/T_F \rfloor} A_{h_t, j}$. On the other hand,

if $h_t > 1$, the sum capacity of the frames within the coverage of infostations $[1, \dots, h_t - 1]$ is given by $\sum_{l=1}^{h_t-1} \sum_{j=1}^{K_l} A_{l,j}$.

Case 2: The infostation most recently visited by the train is $h_t = \arg \max_h \{T_h^i \leq t\}$. Since no block can be delivered during an idle period, the cumulative capacity is given by the sum capacity of all frames in infostations $[1, \dots, h_t]$, i.e., $\sum_{l=1}^{h_t} \sum_{j=1}^{K_l} A_{l,j}$.

APPENDIX B: PROOF OF LEMMA 2

Consider two time instants t_1 and t_2 , such that $T_I \leq t_1 < t_2 \leq T_O$. There are four cases. Case 1: Both t_1 and t_2 are in a transmission period; Case 2: Both t_1 and t_2 are in an idle period; Case 3: t_1 is in a transmission period while t_2 is in an idle period; Case 4: t_1 is in an idle period while t_2 is in a transmission period. Since the proof is similar for the different cases, we show the proof of Case 1 in the following.

If both t_1 and t_2 are in the same transmission period, i.e., $h_{t_1} = h_{t_2}$, we have

$$\begin{aligned} f(t_1) &\leq \sum_{j=1}^{\lfloor (t_2 - T_{h_{t_1}}^i) / T_F \rfloor} A_{h_{t_1},j} + \sum_{l=1}^{h_{t_1}-1} \sum_{j=1}^{K_l} A_{l,j} \\ &= \sum_{j=1}^{\lfloor (t_2 - T_{h_{t_2}}^i) / T_F \rfloor} A_{h_{t_2},j} + \sum_{l=1}^{h_{t_2}-1} \sum_{j=1}^{K_l} A_{l,j} \\ &= f(t_2). \end{aligned} \quad (20)$$

The inequality in (20) is due to $\lfloor (t - T_{h_{t_1}}^i) / T_F \rfloor$ being a non-decreasing function of t , and $A_{h,j}$ being non-negative.

If t_1 and t_2 are in different transmission periods, i.e., $h_{t_1} + 1 \leq h_{t_2}$, we have

$$\begin{aligned} f(t_1) &\leq \sum_{j=1}^{\lfloor (T_{h_{t_1}}^o - T_{h_{t_1}}^i) / T_F \rfloor} A_{h_{t_1},j} + \sum_{l=1}^{h_{t_1}-1} \sum_{j=1}^{K_l} A_{l,j} \\ &= \sum_{l=1}^{h_{t_1}} \sum_{j=1}^{K_l} A_{l,j} \leq \sum_{l=1}^{h_{t_2}-1} \sum_{j=1}^{K_l} A_{l,j} \\ &\leq \sum_{j=1}^{\lfloor (t_2 - T_{h_{t_2}}^i) / T_F \rfloor} A_{h_{t_2},j} + \sum_{l=1}^{h_{t_2}-1} \sum_{j=1}^{K_l} A_{l,j} \\ &= f(t_2). \end{aligned} \quad (21)$$

The first inequality in (21) holds as $T_{h_{t_1}}^i \leq t_1 \leq T_{h_{t_1}}^o$.

APPENDIX C: PROOF OF THEOREM 1

For sufficiency, we first consider the condition $G_s \geq T_h^i + kT_F$. Since $f(t)$ is a non-decreasing function with respect to t according to Lemma 2, we have

$$\begin{aligned} G_s^c &= f(G_s) \geq f(T_h^i + kT_F) \\ &= \sum_{j=1}^{\lfloor (T_h^i + kT_F - T_h^i) / T_F \rfloor} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j} \\ &= \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}. \end{aligned} \quad (22)$$

Similarly, we can obtain $D_s^c \leq \sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ for $D_s \leq T_h^i + (k-1)T_F$.

For necessity, we cannot derive (4) directly from (7) since $f(t)$ is not a bijective function and thus is not reversible. Instead, we resort to (2) and (5) of problem P1. We first prove the inequality part based on contradiction. Consider $G_s^c > \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ in (7). Suppose $G_s < T_h^i + kT_F$, since $f(t)$ is a non-decreasing function of t , we have

$$G_s^c = f(G_s) \leq f(T_h^i + kT_F) = \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}. \quad (23)$$

As (23) contradicts with $G_s^c > \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$, we have $G_s \geq T_h^i + kT_F$.

Next, consider $G_s^c = \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ in (7). Suppose $G_s < T_h^i + kT_F$. Since the service request time is rounded to the beginning time of a frame, we have $G_s \leq T_h^i + (k-1)T_F$. By applying function $f(t)$ on both sides of the inequality, we have

$$\begin{aligned} G_s^c &= f(G_s) \leq f(T_h^i + (k-1)T_F) = \sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j} \\ &\leq \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}. \end{aligned} \quad (24)$$

With $G_s^c = \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$, the first and second inequalities in (24) should take equal signs. Based on the second equality $\sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j} = \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$, we have $A_{h,k} = 0$. According to (5), the summation of the resource allocation variables for the k th frame within the h th infostation coverage is upper-bounded by $A_{h,k}$, i.e., $\sum_{s \in S} x_{h,k,s} \leq A_{h,k}$. Moreover, since $x_{h,k,s}$ can take only non-negative values as stated by (2), we have $x_{h,k,s} = 0, s \in S$. This result indicates that for $G_s^c = \sum_{j=1}^k A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ in (7), we already have $x_{h,k,s} = 0$ for $G_s < T_h^i + kT_F$ in problem P1. The discussion on $D_s^c \leq \sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ is similar and omitted here. Since both sufficiency and necessity are satisfied, (4) is equivalent to (7) for problem P1.

APPENDIX D: PROOF OF LEMMA 3

Define $c_{h,k}^p = \sum_{j=1}^{k-1} A_{h,j} + \sum_{l=1}^{h-1} \sum_{j=1}^{K_l} A_{l,j}$ as the cumulative capacity of all frames prior to the k th frame within the h th infostation coverage. Since $x_{h_1,k_1,s_1}, x_{h_2,k_2,s_2} \geq 1$, the two frames under consideration should have non-zero capacity, i.e., $A_{h_1,k_1}, A_{h_2,k_2} > 0$. Without loss of generality, we consider all blocks of the two frames belong to the n th virtual period, i.e.,

$$\begin{aligned} &\left\{ c_{h_1,k_1}^p + 1, c_{h_1,k_1}^p + 2, \dots, c_{h_1,k_1}^p + A_{h_1,k_1} \right\} \\ &\subseteq [c_n + 1, c_{n+1}] \end{aligned} \quad (25)$$

$$\begin{aligned} &\left\{ c_{h_2,k_2}^p + 1, c_{h_2,k_2}^p + 2, \dots, c_{h_2,k_2}^p + A_{h_2,k_2} \right\} \\ &\subseteq [c_n + 1, c_{n+1}]. \end{aligned} \quad (26)$$

The value of the objective function (1) is the same for X and X' since the total numbers of blocks delivered for services s_1 and s_2 respectively are the same. For feasibility, (2) and (3) hold for X' straightforwardly. Constraint (5) holds for X' since we have

$$x_{h_1, k_1, s_1} + x_{h_1, k_1, s_2} = (x_{h_1, k_1, s_1} - 1) + (x_{h_1, k_1, s_2} + 1) \quad (27)$$

$$x_{h_2, k_2, s_1} + x_{h_2, k_2, s_2} = (x_{h_2, k_2, s_1} + 1) + (x_{h_2, k_2, s_2} - 1). \quad (28)$$

Based on (7), for a feasible resource allocation variable X and $x_{h_1, k_1, s_1} \neq 0$, we have

$$G_{s_1}^c < \sum_{j=1}^{k_1} A_{h_1, j} + \sum_{l=1}^{h_1-1} \sum_{j=1}^K A_{l, j} = c_{h_1, k_1+1}^p. \quad (29)$$

Inequality (29) is equivalent to $G_{s_1}^c \leq c_{h_1, k_1}^p$ since the service request time is rounded to the beginning time of a frame. Similarly, $D_{s_1}^c \geq c_{h_1, k_1}^p + A_{h_1, k_1}$, $G_{s_2}^c \leq c_{h_2, k_2}^p$, and $D_{s_2}^c \geq c_{h_2, k_2}^p + A_{h_2, k_2}$. By (25) and (26), we have

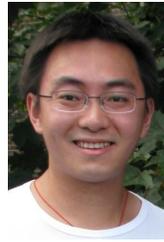
$$\begin{aligned} G_{s_2}^c &\leq \max \{G_{s_1}^c, G_{s_2}^c\} \leq c_n \leq c_{h_1, k_1}^p < c_{h_1, k_1}^p + A_{h_1, k_1} \\ &\leq c_{n+1} \leq \min \{D_{s_1}^c, D_{s_2}^c\} \leq D_{s_2}^c \end{aligned} \quad (30)$$

$$\begin{aligned} G_{s_1}^c &\leq \max \{G_{s_1}^c, G_{s_2}^c\} \leq c_n \leq c_{h_2, k_2}^p < c_{h_2, k_2}^p + A_{h_2, k_2} \\ &\leq c_{n+1} \leq \min \{D_{s_1}^c, D_{s_2}^c\} \leq D_{s_1}^c. \end{aligned} \quad (31)$$

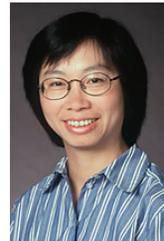
The second and sixth inequalities in (30) (and (31)) hold since no further partition (by G_s^c or D_s^c) exists within each virtual period according to definition (8). As a result, (7) holds for X' .

REFERENCES

- [1] H. Liang and W. Zhuang, "Resource allocation for on-demand data delivery to high-speed trains via trackside infostations," submitted to IEEE GLOBECOM 2011.
- [2] International union of railways. <http://www.uic.org/>.
- [3] Thales selects BelAir networks Wi-Fi for Bergen light rail project. <http://www.belairnetworks.com/>.
- [4] D. H. Ho and S. Valaee, "Information raining and optimal link-layer design for mobile hotspots," *IEEE Trans. Mobile Comput.*, vol. 4, no. 3, pp. 271–284, May–Jun. 2005.
- [5] S. Motahari, E. Haghani, and S. Valaee, "Spatio-temporal schedulers in IEEE 802.16," in *Proc. IEEE GLOBECOM'05*, pp. 566–570, Nov. 2005.
- [6] C. Sue, S. Sorour, Y. Youngsoo, and S. Valaee, "Network coded information raining over high-speed rail through IEEE 802.16j," in *Proc. IEEE PIMRC'09*, pp. 1138–1142, Sept. 2009.
- [7] L. Zhu, F. R. Yu, B. Ning, and T. Tang, "Cross-layer design for video transmissions in metro passenger information systems," *IEEE Trans. Veh. Tech.*, vol. 60, no. 3, pp. 1171–1181, Mar. 2011.
- [8] B. B. Chen and M. C. Chan, "MobTorrent: a framework for mobile Internet access from vehicles," in *Proc. IEEE INFOCOM'09*, pp. 1404–1412, Apr. 2009.
- [9] J. C.-P. Wang, H. El Gindy, and J. Lipman, "On cache prefetching strategies for integrated infostation-cellular network," in *Proc. IEEE LCN'06*, pp. 185–192, Nov. 2006.
- [10] J. Xu, X. Tang, and W. Lee, "Time-critical on-demand data broadcast, algorithms, analysis, and performance evaluation," *IEEE Trans. Paralle. Distr. Sys.*, vol. 17, no. 1, pp. 3–14, Jan. 2006.
- [11] R. Dewri, I. Ray, I. Ray, and D. Whitley, "Optimizing on-demand data broadcast scheduling in pervasive environments," in *Proc. ACM EDBT'08*, pp. 559–569, Mar. 2008.
- [12] A. Balasubramanian, B. N. Levine, and A. Venkataramani, "Replication routing in DTNs: a resource allocation approach," *IEEE/ACM Trans. Netw.*, vol. 18, no. 2, pp. 596–609, Apr. 2010.
- [13] Y. Huang, Y. Gao, K. Nahrstedt, and W. He, "Optimizing file retrieval in delay-tolerant content distribution community," in *Proc. IEEE ICD-CIS'09*, pp. 308–316, Jul. 2009.
- [14] Shanghai-Hangzhou High-Speed Railway. <http://en.wikipedia.org/>.
- [15] Zhejiang news. <http://zjnews.zjol.com.cn/>.
- [16] M. Papadopoulos and H. Schulzrinne, *Peer-to-Peer Computing for Mobile Networks - Information Discovery and Dissemination*. New York: Springer, 2008.
- [17] G. L. Nemhauser and L. A. Wolsey, *Integer and Combinatorial Optimization*. New York: John Wiley & Sons, 1988.
- [18] K. R. Baker and D. Trietsch, *Principles of Sequencing and Scheduling*. New Jersey: John Wiley & Sons, 2009.
- [19] P. Baptista, L. Peridy, and E. Pinson, "A branch and bound to minimize the number of late jobs on a single machine with release time constraints," *Eur. J. Oper. Res.*, vol. 144, no. 1, pp. 1–11, Jan. 2003.
- [20] C. Durr, L. Jez, and K. T. Nguyen, "Online scheduling of bounded length jobs to maximize throughput," *Lect Notes Comput. Sci.*, vol. 5893, pp. 116–127, May 2010.
- [21] E. L. Lawler, "A dynamic programming algorithm for the preemptive scheduling of a single machine to minimize the number of late jobs," *Ann. Oper. Res.*, vol. 26, no. 1, pp. 125–133, Dec. 1990.
- [22] Train Schedule. <http://lieche.58.com/>.
- [23] A. Ahmad, "Simulation-based local train mobility model," in *Proc. ACM SpringSim'10*, pp. 1–8, Apr. 2010.
- [24] R. Liu and Y. Deng, "Comparing operating characteristics of high-speed rail and maglev systems: case study of Beijing-Shanghai corridor," *Transp. Res. Rec.*, vol. 1863, pp. 19–27, Jan. 2004.
- [25] S. K. Baruah, J. Haritsa, and N. Sharma, "On-line scheduling to maximize task completions," in *Proc. IEEE RTSS'94*, pp. 228–236, Dec. 1994.



Hao Liang (S'09) received the B.Sc. degree from Nanjing University of Science and Technology, China, in 2005, and the M.Sc. degree from Southeast University, China, in 2008, both in electrical engineering. He is currently working toward a Ph.D. degree at the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His current research interests are in the areas of wireless communications, wireless networking, and smart grid. He received the Best Student Paper Award from IEEE 72nd Vehicular Technology Conference (VTC Fall-2010), Ottawa, Canada. He served as the Technical Program Committee (TPC) Member for IEEE VTC Fall-2011 and IEEE VTC Fall-2010. He is the System Administrator of IEEE Transactions on Vehicular Technology.



Weihua Zhuang (M'93-SM'01-F'08) received the B.Sc. and M.Sc. degrees from Dalian Maritime University, China, and the Ph.D. degree from the University of New Brunswick, Canada, all in electrical engineering. Since October 1993, she has been with the Department of Electrical and Computer Engineering, University of Waterloo, Canada, where she is a Professor and a Tier I Canada Research Chair in wireless communication networks. Her current research focuses on resource allocation and QoS provisioning in wireless networks.

Dr. Zhuang is a co-recipient of the Best Paper Awards from the IEEE Multimedia Communications Technical Committee in 2011, IEEE VTC Fall-2010, IEEE WCNC 2007 and 2010, IEEE ICC 2007, and the International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine) 2007 and 2008. She received the Outstanding Performance Award 4 times since 2005 from the University of Waterloo for outstanding achievements in teaching, research, and service, and the Premier's Research Excellence Award in 2001 from the Ontario Government for demonstrated excellence of scientific and academic contributions.

Dr. Zhuang is the Editor-in-Chief of the IEEE Transactions on Vehicular Technology, and the TPC Symposia Chair of the IEEE Globecom 2011. She is a Fellow of the Canadian Academy of Engineering (CAE) and the IEEE, and an IEEE Communications Society Distinguished Lecturer.