Transmission Scheduling for 
Data Wireless Networks in 
Mass Transit Systems

Thesis submitted in partial fulfillment of the requirements for the 
master degree "M.Sc." at the School of Computer Sciences 
Tel-Aviv University 
by 
Elisha Rosensweig

Prepared under the supervision of Prof. Hanoch Levy
 onload=window.location='https://www.google.com/'>זמנון ייצוגים ברשאות סלולריות
עבורה מערכות תעבורה המובילות

ה埂שה חלקגעמיידה בדרישות לקבלת תואר שני בביית הסדר למediator המוחשב
אוניברסיטת תל אביב
על ידי
אלישע רובנცווי

ה埂נ תחת הדריכות של פרופ' חנוך לי

תנומתنشיאור
Abstract

Scheduling of down-link transmission for data applications in wireless networks has been the focus of recent research. Channel-aware scheduling algorithms were shown to achieve significant performance gains by accounting for the time varying reception capacities and exploiting their independence across the mobile users. We deviate from these studies by considering mass-transit systems, in which reception capacities of the mobile users are not independent but are rather positively correlated to each other and thus pose a potential problem due to the bursty traffic requirements they inflict on the cells. We study the performance of these systems focusing on trains and aiming at providing a solution to this bursty traffic. We propose a down-link scheduling algorithm that maximizes the resources allocated to the mobile users outside the train while obeying the delay constraint of the train users. We further propose a new architecture for train wireless support systems, based on spatially separated reception train antennas. The combination of the proposed scheduling algorithm with the proposed architecture results in significant improvement of system performance.

This work was supported in part by REMON - a grant from the Israeli Ministry of Industry and Trade.
## Contents

1 Introduction and Related Work ........................................ 4

2 Model description .................................................... 8
   2.1 Cellular Network ................................................. 8
   2.2 Cellular Users .................................................. 10

3 Efficient transmission Scheduling Algorithm: Load Minimization under
   Delay Constraints .................................................. 12
   3.1 Motivation ......................................................... 12
   3.2 Basic Model Properties .......................................... 14
   3.3 Buffer-and-Burst algorithm (B&B) basic structure .......... 16
   3.4 Buffer-and-Burst algorithm (B&B) specification ............ 19
      3.4.1 Segment $\alpha$ ........................................... 20
      3.4.2 Segment $\beta$ ............................................. 21
      3.4.3 Segment $\gamma$ ........................................... 23
      3.4.4 Segment $\delta$ ........................................... 24
   3.5 Algorithm specification and complexity ...................... 28
   3.6 Implementation issues .......................................... 30

4 The MASS Architecture for Capacity Increase ................... 31
   4.1 MASS Architecture .............................................. 31
   4.2 Notation ......................................................... 34
   4.3 Min Slot-Capacity ................................................ 35
   4.4 Total Slot-Capacity .............................................. 35
   4.5 Symmetric model ................................................ 37

5 Buffer-and-Burst and MASS combined .............................. 38
   5.1 Preliminaries ..................................................... 40
   5.2 3 antenna deployment ........................................... 42
   5.3 $n > 3$ antennas ................................................ 48
6 System Evaluation

6.1 The effects of MASS .............................................................. 49
6.2 B&B and B&B+MASS performance ........................................... 51

7 Appendix .............................................................................. 55
1 Introduction and Related Work

There has been a growing interest in data applications within wireless networks, due to the increase in systems which support them. One topic which has been the subject of recent research, is scheduling of down-link transmission for data applications in wireless networks. It has been shown ([1]-[15]) in numerous settings that using channel-aware scheduling on the base-station down-link, where service is given at each point in time to a mobile user based on his channel capacity, can result in a dramatic increase of both the QoS of each mobile user and the overall system performance.

The improvement in performance achieved by channel-aware policies is due to the variability of the mobile users’ reception capacity, which is an inherent property in wireless systems due to user mobility and signal fading. Simplistic algorithms may end up spending significant amount of transmission resources (transmission time or slots) attempting to deliver data to users whose capacity is momentarily very low. Conversely, channel-aware policies follow the basic principle of aiming to transmit to any given user during times in which his reception capacity is relatively high, and allowing other users to receive service when his reception capacity is low. They are therefore most suitable for more delay-tolerant types of traffic, such as data applications, which allow the scheduler the freedom to hold off transmission until the effective transmission rate (“capacity”) to the specific user will be high. Since transmitting at every slot to the user with the highest capacity would eventually lead to the starvation of users with low capacity, many policies have been suggested which achieve high capacity gains while still retaining some measure of QoS to all users.

Previous research in this field has been mostly based on two simplifying assumptions: the independence of capacity fluctuations across users, and a static user population per cell. The first assumption is essential for all such approaches, since it is this variation which is exploited by them in order to increase the available capacity. The second is based on the observation that the scheduling algorithms operate at the packet level, at which the user population evolves relatively slowly. In recent years, there has been some discussion regarding dynamic user populations [1][25]. These focused on cases in which users leave the user population when the action they were performing in the system is completed, i.e. the
The growing scope of wireless services introduces, however, systems and situations in which both of the assumptions mentioned above are problematic. One such scenario is support for wireless services on board mass-transit vehicles, such as trains. The main approach for providing wireless coverage onboard trains is to equip the train with a central antenna unit. The central antenna unit communicates with the terrestrial wireless base-stations in a manner which is similar to the way a standard user does. Additionally, the central antenna unit connects to an internal Onboard Wireless Network (OWN). Each user on board the train connects to the OWN, and all the wireless traffic is channeled through the central antenna unit to/from the base-station. This is the approach taken by many companies, such as PointShot, Icomera, 21Net and more (for a sample list of these, see [24]). Such an approach solves many difficulties resulting from traveling at high speeds and the need to penetrate the external hull of the train. Our interest is in the traffic scheduling carried out at the terrestrial base-station which is the bottleneck of the wireless system and which must concurrently serve both the standard users and the train users.

The introduction of a train system into the wireless system poses significant challenges on the scheduling of the wireless transmission at the terrestrial base-station. The challenges are due to the high number of active users residing on board a train whose total traffic volume can create a significant load on the base-station, since all of them appear in the cell at the same time. Of course, one could easily solve this problem by increasing the base-stations capacity or adding base-stations along the train tracks. However, since the train remains within the cell for only a small fraction of the time (i.e. an order of a few tens of seconds, at a frequency of once every few minutes), such an increase is not economical, as the base-station will remain at very low utilization most of the time. Thus, due to the train passage, the base-station is subject to frequent drastic increases of load (load bursts), during which it must maintain proper QoS for the standard users as well as for the train users. Also, from the trains’ perspective this issue is a persistent one, since in every cell passed by the train the overall load on the system can be high.

\[^1\text{In some systems the OWN can be connected also to a satellite system. Such connection is out of the scope for this study and is left for future research.}\]
Considering this wireless-support architecture, we note that the assumptions used in previous work to construct scheduling policies are also somewhat questionable in this new scenario. To begin with, all the train users experience the exact same capacity, thus limiting the ability of existing channel-aware policies to supply high transmission rates. One possible approach is to recognize that the architecture previously described may allow us to relate to all the users on board the train as a single, virtual user. This is so since all users on board the train can be transmitted to in the same slot, allowing the internal OWN to direct the incoming wireless traffic to the different users. Based on this observation, we may return to the original model of users roaming the cell independently, viewing all the train users as a single such user. Such a representation raises additional difficulties, however, since it is clear that this virtual user requires special treatment due to the large number of actual users it represents. This could be done using a weighted system, where each user is assigned a weight which is used to create a service scale, allocating more service time to users with higher weights. Since the number of users on the train is dynamic and can grow to be quite large, its’ relative weight compared to all other users can result in denial-of-service to all the cell users, which is an undesirable outcome.

Additionally, the need to address dynamic user population is amplified by the introduction of such systems. As the speed of such trains is very high, it is no longer clear that the packet-level QoS is insensitive to the movement of these clients between cells. This is a problem not so much on an objective level but rather a relative one. For example, channel-aware policies which use stochastic analysis to model the behavior of a system rely heavily on the fact that each user is in the cell long enough to balance out his QoS. However, in cases where the user population is mostly static, and a single user (i.e. train) moves much faster between cells, the performance of this user will be relatively erratic. A similar problem is expected to occur from the admission control standpoint. Some admission control policies give precedence to existing users over newly-arrived users, in order to lower the call-dropping rate. With a train moving in and out of cells at high frequency, the users on the train are at risk of being constantly and consistently denied service. This problem could be partially dealt with by differentiating between call blocking and handoff blocking, as has been done previously at [27].
Both of these problems lead us to suggest that, in the case of trains and similar mass-transit vehicles, the correct approach requires adopting a *multi-cell* view of the QoS experienced by the train users. This view of things, once accepted, raises many problems which need to be addressed, the most central being how to coordinate and compare the services given to cell-based users to those given to train-based users. In this paper we claim that the correct way to deal with this is to give train users a degree of precedence over all cell-based users when allocating services. Such an approach is justified when one considers that the trains’ high-speed causes it to spend a relatively short time in each cell, thus ensuring that the overload within each cell, though undesirable, will be relatively short. With smart allocation algorithms, the negative effects of train-precedence can be mitigated.

Our purpose is to study this system and deal specifically with its bursty traffic nature. We aim at proposing mechanisms, as well as base-station scheduling algorithms, for coping with these problems and for providing efficient system operation. To achieve this end, we make use of two factors in which trains are unique. The first is the predetermined route through the network which the train follows, which has been previously employed on trains in an attempt to alleviate the problem of power outages [19] [20]. This also fits into the general framework of using direction prediction in order to allocate services to users in a network [26]. The second is the long body of the train, which is spread over a significant length.

Following a description of the model (Section 2) we start this work (Section 3) with constructing a base-station scheduling algorithm. Our approach aims at optimizing the scheduling of the train transmissions so as to minimize the amount of transmission time dedicated to the train in each cell, while obeying the delay constraint of the transmissions to the train. We recognize that an optimal solution must account for all network cells and therefore is too complex to solve directly. We therefore propose an approach that decomposes the problem into a group of single-cell problems, and provide a mechanism for combining cell-oriented solutions, which can be used to construct a network-wide solution without greatly increasing the allocation cost. We then move on to design a scheduling algorithm, called *Buffer-and-Burst* (B&B), which is shown to provide a $(1 + \epsilon)$-competitive allocation of slots in a given cell. Such a solution is commonly referred to throughout the paper as
quasi-optimal.

Aiming at further alleviation of system overload, we propose (Section 4) to take advantage of the special physical properties of the train, namely its considerable length. This implies that, at any given moment in time, the potential reception capacities at two different parts of the train significantly vary. Aiming at exploiting this property, we propose a new train architecture called Multi-Antenna Spatially-Separated (MASS), in which the train uses a multiplicity of antennas, spread over the train, to which the base-station may transmit, and a switching mechanism that, at every moment, selects only the antenna with the (momentarily) highest reception level to be operative. Focusing on the effects of the distance between base-stations and users on the reception capacity, we derive the optimal deployment of these antennas, w.r.t. several performance measures.

Though deploying multiple antennas over the train hull can improve performance in many ways, we specifically note the large potential it has with regards to the Buffer-and-Burst Policy. We therefore proceed to discuss the possible advantages of combining the two components of this work (Section 5). Here as well the question arises as to the optimal deployment of the multiple antennas. Under several simplifying assumptions we present the optimal such antenna deployment as well as an overview of a proof for its optimality. Finally, in Section 6 we present simulation results demonstrating the performance gains achieved by these solutions, independently and when employed together.

This work has appeared in [28].

2 Model description

2.1 Cellular Network

A wireless (cellular) network consists of transmission base-stations, which are geographically spread, and users, which roam the cells defined by these base-stations. The connection of a user to the network is via a wireless link, typically to and from the base-station that is closest to the user.

The effective transmission rate ("capacity") at which a base-station can transmit to a
user depends on many factors. One central factor is the distance of the user from the base-station, responsible for free-space loss. The impact of distance on the reception level is believed to be modeled by the function

\[ \text{Capacity}(x) \propto x^\eta \]  

for some \(-4 \leq \eta \leq -2\), where \(x\) is the distance between user and base-station. Thus, the user, as well as the system, can benefit from having the user relatively close to a base-station.

As described earlier, our work attempts to take advantage of the special characteristics of trains, in order to improve the quality of services in the cell. One such property is the predictability of the route taken by the train. Focusing primarily on the effects of distance on the reception level, this route can be translated into the available transmission capacity at each point on the track. Combined with the expected speed of the train at each point along its route - also a largely predictable property - it is possible to predict with relative accuracy the expected capacity a user on the train will experience at each point in time.

There are, of course, other factors which affect the user reception capacity, such as topography, weather conditions, multi-path fading and others. These issues are discussed extensively in the literature, and many solutions have been proposed and applied in the field. Solutions to these problems are generally orthogonal to our focus in this paper, and can be combined with our solution.

Let \( S := (s_i \in \mathbb{R}^2)_{i \in \mathbb{Z}} \) be an infinite series of base-station locations. We consider only those base-stations which define the cells through which the train moves. We list the base-stations according to their order along the train tracks, which are assumed to be of infinite length and laid out in a relatively straight line coinciding with the \(x\)-axis on an imaginary plane. For base-station \( s_i \), let \( d_i \) represent the minimal (euclidian) distance of \( s_i \) from the train tracks, and \( x_{d_i} \) the point on the tracks which is closest to this \( s_i \). For each pair of base-stations \( s_i \) and \( s_j \) let \( z_{i,j} = d(x_{d_i}, x_{d_j}) \) be the distance along the tracks between two adjacent minimal distance points. Figure 1 portrays this model, and Figure 2 reflects the fluctuations in the reception capacity over time as the train moves through the cells.
2.2 Cellular Users

Mobile users are categorized into cell users, which are standard mobile users roaming freely in their respective cells, and train users, which are train passengers connected to the OWN onboard the train. All train users can be transmitted to in parallel, yet must share capacity. We therefore treat the train as a single (large) user denoted $Tr$, and commonly refer to the train itself as a user, and to packets transmitted to train users as transmitted to the train itself.

We use a time-slotted model, where time is divided into short transmission slots, and during each slot the base-station transmits to a single user selected according to some predefined policy. Ours is an early-arrival system, in which transmission takes place only after packets have arrived, so packets can be transmitted without delay during the slot in which they arrive. At the beginning of each slot $t$, the base-station assesses the capacity of each user, in terms of the number of data (packets) which can be transmitted to that user during the specific slot. It then uses this information to determine the allocation of $t$ for down-link transmission. This type of scheduling is commonly termed ”channel-aware”.

Each user $u$ is associated with two time-dependent functions and a constant delay con-
straint which is service-dependant. The first function is the data-arrival rate, denoted $In^u(t)$, representing the number of packets which arrive at the base-station for transmission to user $u$ at slot $t$ (all packets are assumed to be of equal size). Packets which are not forwarded immediately to $u$ are stored in a buffer until the base-station allocates a slot for transmitting them.

The second function is the user’s reception capacity at slot $t$, expressed in terms of packets, and denoted $C^u(t)$. When a slot is allocated to a specific user, the base-station transmits the buffered packets according to order of arrival (FIFO), up to the user’s reception capacity limit. If $S = \{t_1, ..., t_k\}$, $C^u(S) = \{C^u(t_1), ..., C^u(t_k)\}$. In a similar manner, if $p$ is a point on the tracks, $C^u(p)$ is used to denote the capacity of user $u$ at point $p$.

Packet granularity is chosen in such a way as to ensure that both transmission capacity and data-arrival rate have only integer values. Each packet is therefore transmitted in full during a single slot. Additionally, when these functions are considered to be constant, they shall be commonly denoted $C^u$, $In^u$.

The delay constraint $w^u$ reflects the largest allowable interval between the time a packet arrives at the base-station and the time it is transmitted (i.e. congestion delay). This is determined based on the type of service currently requested by the user. In this work we assume a maximal delay constraint, denoted $w_{max}$, is equally applied to all train users. The actual delay constraint depends on the type of service being rendered: while voice transmission would require tight delay constraint, for data-transfer a reasonable delay could be several seconds [1]. Though the theoretical approach taken in this paper is valid for all types of traffic, the suggestions made in this paper are practical and useful mainly for supporting data applications.
3 Efficient transmission Scheduling Algorithm: Load Minimization under Delay Constraints

3.1 Motivation

When considering the problem of allocating services to users on board trains, there are several reasons to adopt a different approach than that taken in dealing with classic mobile users. As discussed in the introduction, the large number of users on board the train creates a situation in which a large body of users is moving rapidly through the network, creating short bursts of system overload in their passing. Such behavior is uncharacteristic of standard wireless users, and therefore demands special treatment. Specifically, aside for the system overload it creates in each cell, the users on board the train will tend to suffer from the fact that they are not in the cell for long enough in order to experience the same QoS the cell users do, or to recieve service under several admission control schemes.

It turns out that the fast pace in which trains move between cells can itself be used in order to solve the problem created by this same quality. Specifically, it is clear that a user, who spends only a short period of time in a cell, will require less support from the base-station, which in turn ensures it will have less of a negative impact on the service tendered to the rest of the users in the cell. When these users move in and out of the cell in unison, as in the case of train passengers, a policy which gives them some degree of priority over the rest is justified, since their negative effect on system performance is limited.

This approach takes form in our work in the following way. For each packet destined to be transmitted to a train user, we require that the congestion delay, i.e. the duration between its arrival at the base-station and the actual transmission to the user, is no more than $w_{\text{max}}$ slots. Any allocation of service which meets this requirement is considered legal, regardless of the negative effects on the cell users. However, our goal is to minimize these negative effects and therefore we strive to develop an allocation policy which will bring the number of slots used in service of the train to a minimum.

Instead of dealing directly with the multi-cell optimization problem, which would seem to be intractable due to the large number of variables to account for, we decompose the problem
and deal separately with each cell. During each epoch, we focus on the cell through which the train is passing at the time. Once the optimal solution is devised per cell, we demonstrate how localized solutions can be combined to form a quasi-optimal global solution, encompassing the entire train route.

We treat the train as a single (large) user $Tr$ and assume the data arrival rate to be a constant $In^{Tr}$. Note that the fact that the train represents a large number of users supports the assumption that the total train arrival rate is basically constant, as fluctuations in the train user population will tend to cancel each other out. Applying Buffer-and-Burst to more dynamic settings is possible with some slight modifications, as is discussed later in this section (3.6).

In this paper we use an early-arrival model, in which transmission occurs after the arrival of packets in the slot. Packets are stored in a buffer until transmission. Let $B^u_{alg}(t)$ denote the number of packets in the buffer at slot $t$, before any transmission may take place in slot $t$. When the data arrival rate is constant, the delay constraint can also be formulated in terms of maximal legal buffer size - the buffer may not exceed $(w_{max} + 1) \cdot In^{Tr}$ packets \(^2\) at all times.

We further denote $Ex^u(t) := \max\{0, C^u(t) - In^u\}$ to be the excess capacity at slot $t$, which can be thought of as the amount of packets from the buffer which can be transmitted during slot $t$ if also the packets arriving during slot $t$ were to be transmitted at $t$. For a set of consecutive slots we define in similar manner $Ex^u[t_1, t_2] = \sum_{t=t_1}^{t_2} Ex^u(t)$ be the maximal number of packets that can be transmitted between slots $t_1$ and $t_2$ (inclusive), if all the packets arriving during that period are transmitted as well. Note that there is no guarantee that all this excess capacity can be put to use while meeting the delay constraint for all packets.

**Definition 1** A user service allocation is a vector $A^u \in \{0, 1\}^T$ where $T$ is the number of slots available for allocation and $A^u(t) = 1$ iff user $u$ is transmitted to at slot $t$. For a user service allocation $A^u$, the allocation size $|A^u| = \sum_{t=1}^{T} A^u(t)$ is the number of slots allocated

\(^2\)We allow $(w_{max} + 1) \cdot In^{Tr}$ packets to be transmitted during a single slot, due to the fact that we deal here with an early-arrival model. Packets which are transmitted during the slot in which they arrived are considered to have experienced a delay of 0.
for service in $A^u$. A service block in $A^u$ is a set of consecutive slots, all of which are marked for transmission. We denote a service block from slot $t_1$ to slot $t_2$ as $A^u[t_1, t_2]$.

A legal user service allocation is a service allocation which, when followed, ensures each packet obeys its delay constraint. One characteristic of such allocations is that, for any $t$ s.t. $A^u(t) = 1$ in a legal user service allocation $A^u$, the base-station transmits, at slot $t$,

$$\min\{C^u(t), B_{A^u}^u(t), (w_{\text{max}} + 1) \cdot I^u\}$$

packets at slot $t$ according to the order of arrival (FIFO). Note here that the buffered data is dependant on the user service allocation vector $A^u$, which can be matched to some algorithm $\text{alg}$ that produces $A^u$.

**Definition 2** An optimal allocation of a packet set $P$ transmitted to user $u$ is the minimal-sized legal user service allocation for all packets in $P$.

Throughout this section we shall only be referring to train service allocations and thus omit to mention this explicitly, using simply $A$ to refer to $A^{Tr}$.

### 3.2 Basic Model Properties

Though we have imposed no explicit limitations on the relationship between the data arrival rate and the reception capacity values, the introduction of delay constraints into the model allows us to limit discussion to specific capacity functions. For any general reception capacity function $C^{Tr}()$ and a constant data arrival rate $I^{Tr}$, define a delay-bounded capacity function as

$$C^{Tr^*}_*(t) = \min\{C^{Tr}(t), (w_{\text{max}} + 1) \cdot I^{Tr}\}.$$  

**Theorem 1** Given a capacity function $C^{Tr}()$, a service allocation $A$ is legal w.r.t $C^{Tr}$ iff $A$ is a legal service allocation w.r.t $C^{Tr^*}_*$.

**Proof:** As expressed in Equation 2, in any legal service allocation $A$ every slot may transmit up to $(w_{\text{max}} + 1) \cdot I^{Tr}$ packets, since otherwise there is a packet which arrived more than $w_{\text{max}}$ slots earlier and which is only transmitted now, thus exceeding the allowable delay. This implies that if $C^{Tr}(t) > (w_{\text{max}} + 1) \cdot I^{Tr}$, the additional capacity cannot be put to use in any legal context and thus has no effect on the resulting service allocation.  


We can thus limit ourselves to delay-bounded capacity functions. Given a general capacity function, slots with capacity $> (w_{\text{max}} + 1) \cdot \text{In}^{Tr}$ are considered to have capacity $(w_{\text{max}} + 1) \cdot \text{In}^{Tr}$.

Theorem 1 leads us to recognize that the actual potential of a slot is determined by the ratio between its capacity and the buffer size at the time. When a slot is allocated for transmission, the number of packets which can be transmitted during the slot is the minimum between the number of packets in the buffer and the slot capacity. Generalizing this rule to the case of a block of slots, we realize that this rule can be formulated as the relationship between the number of packets in the buffer at the beginning of the block and the excess capacity available in the block, as shown in the following lemma:

**Lemma 1** Assume some legal allocation until slot $t_1$. If for all $t_1 \leq t \leq t_2$ $C^u(t) \geq \text{In}^u$, then allocation of all slots $t_1, \ldots, t_2$ will result in a legal allocation until $t_2$ and $B_{\text{alg}}^u(t_2 + 1) = \max \{0, B_{\text{alg}}^u(t_1) - E x^u[t_1, t_2]\} + \text{In}^u$.

**Proof:** Since for all $t_1 \leq t \leq t_2$ $C^u(t) \geq \text{In}^u$, the buffer cannot grow during the transmission block, so legality is assured. At every slot the buffer size is reduced by $\min \{E x^u(t), B_{\text{alg}}^u(t)\}$, so overall the buffer size at the end of the transmission block is $\max \{0, B_{\text{alg}}^u(t_1) - E x^u[t_1, t_2]\}$.

Each slot is associated with the cell through which the train passes at that time. For each cell $i$ define $M_{ij}$ to be a set of the $j$ slots with highest capacity cell $i$, using as a tiebreaker the rule that earlier slots are chosen over later slots. This series of sets maintains the rule that $M_{ij}^{j+1} \supset M_{ij}^j$. Let $M_{ij}^j(h)$ ($1 \leq h \leq j$) be the $h$-th slot in this set according to sequential order (See Figure 3). Since the reception capacity of the train is a function of its distance from the base station, and we assume the train tracks are laid out in a straight line, $M_{ij}^j$ is always a contiguous block of slots, which take place normally around the time the train is half-way through the cell$^3$. We further denote $M_{max}^i = M_k^i$ for $k$ defined as

$$k = \min_{E x^{Tr}(M_{ij}^h) \geq w_{\text{max}} \cdot \text{In}^{Tr}} h.$$  

---

$^3$Recall we are dealing here with delay-bound capacity functions. If the original function was not delay bound, we still choose the slots to make up $M_{ij}^j$ according to the order in which they were in the original function.
Figure 3: $M_i^t$

$M_{\text{max}}^i$ is therefore the smallest such set with enough capacity to transmit a full buffer combined with the packets arriving during the set itself. In theory, such a $k$ for which Equation 4 is true may not exist in any given cell. This might occur when the data arrival rate is relatively high and/or when the delay constraint is very relaxed (i.e. $w_{\text{max}}$ is very large). However, such cases are generally impractical and uncommon. High data arrival rates are usually not manageable within reasonable delay constraints, and extremely relaxed delay constraints tend to lower the QoS of users. In this paper we assume, therefore, that such a $k$ exists, though it is possible to modify the algorithm to accommodate situations in which this requirement is dropped.

Observation 1 For all $t \in M_{\text{max}}^i$, $C^{Tr}(t) > I^{Tr}$. This is true since for any slot $t$ in which $C^{Tr}(t) \leq I^{Tr}$ we know $E^{Tr}(t) = 0$, so from minimality of $|M_{\text{max}}^i|$ such a slot would not be included in $M_{\text{max}}^i$.

From Observation 1 and Lemma 1 it follows that for every $M_j^i \subseteq M_{\text{max}}^i$, if the buffer contains $E^{Tr}[M_j^i(1), M_j^i(j)]$ packets, all the slots in $M_j^i$ could be fully and legally (i.e. no delay constraint is violated) utilized.

3.3 Buffer-and-Burst algorithm (B&B) basic structure

Figure 2 qualitatively depicts the train reception corresponding to the train-tracks and cell structure given in Figure 1, which reflect the situation when the tracks form approximately a straight line through the cell. The reception level increases as the train approaches the
base-station and drops as it moves away from it. The reception level has a local minimum at the intersection points between the cells where handoff between the base-stations is assumed to occur. In a multi-cell environment the train goes through cycles of such reception level increase and decrease. As Figures 1 and 2 suggest, our analysis is not limited to symmetric base-station positioning w.r.t. the train tracks.

In order to deal with the allocation of service over several cycles, we use a "divide and conquer" approach, treating each cycle separately, devising an optimal cycle allocation, and then combining these local allocations into one overall allocation. Cycle allocations are similarly achieved by subdivision and recombination. Such allocations, which are designed for the optimal transmission of a sub-set of packets, will be referred to as locally optimal.

Formally, we define the transition points between cycles as all the capacity local minima points. However, when the capacity at a local minima slot $t_{min}$ drops below the incoming data rate, the transition point moves to the closest point preceding $t_{min}$ at which the capacity is equal to or greater than the incoming rate. Figure 4 depicts a cycle when this scenario takes place.

Combining locally optimal allocations can result, of course, in a suboptimal allocation overall. We rely on the following theorem to ensure the resulting allocation exceeds the size of the optimal allocation by, at most, a small constant per cycle, resulting in a quasi-optimal combined allocation:

**Theorem 2** Let $P$ be a set of packets arriving over a contiguous set of slots, transmitted by some algorithm LocOpt using the "divide and conquer" structure described above. Let $\{P_1, ..., P_n\}$ be the partition of $P$ into $n$ sets of consecutive packets, where $P_i$ is transmitted in locally-optimal fashion by LocOpt. Denote by $OPT(P_i)$ the allocation LocOpt generates for packets $P_i$. If these locally-optimal allocations are disjoint (i.e. no slot is allocated in two different local allocations), then

$$\sum_{i=1}^{n} |OPT(P_i)| \leq |OPT(P)| + (n - 1).$$

**Proof:** Let $OPT^*(P_i)$ be the allocation for packets in $P_i$ according to some globally optimal allocation denoted $OPT(P)$. Since $OPT(P_i)$ is the locally-optimal allocation,
obviously $|OPT(P_i)| \leq |OPT^*(P_i)|$. Therefore, the only way in which it is possible that $|OPT(P)| < |\bigcup_{i=1}^n OPT(P_i)|$ is if there is an overlap between allocations the $OPT^*(P_i)$.

Due to the FIFO transmission order, all the slots allocated in $OPT^*(P_i)$ precede the slots allocated in $OPT^*(P_{i+1})$. Thus, for every two consecutive transmission sets there can be, at most, one slot which is allocated in both for transmission: the last slot allocated in $OPT^*(P_i)$ might also be the first slot allocated in $OPT^*(P_{i+1})$. For similar reasons, no overlap can exist between allocations which are not transmitting consecutive sets of packets. There are $n-1$ pairs of consecutive packet sets, and thus we get $\sum_{i=1}^n |OPT^*(P_i)| \leq |OPT(P)| + (n-1)$ which bounds the performance decrease as stated in the theorem.

In our case, we divide each cycle into 4 such sets and thus achieve an allocation of cardinality $\leq |OPT| + 3$ per cycle, which is asymptotically equivalent to $OPT$, since the length of each slot has no theoretical lower bound. This equivalence holds also for all practical purposes, since the number of slots in a standard micro-cell cycle is well over $10^4$. We shall refer to such performance by our algorithm as $(1 + \epsilon)$-competitive, where $\epsilon = \frac{3}{|OPT(P)|}$.

The remaining difficulty in the algorithm construction is to find a useful packet-partitioning, such that the locally-optimal allocation can be construed for each packet set in a manner which will ensure disjoint locally-optimal allocations for all packets.

We will achieve such a packet partitioning in an implicit manner. In each cycle (displayed in Figure 2), at most four disjoint slot sets are defined. Each set is made up of a contiguous block of slots, and thus shall be referred to from here on as a segment. Each such segment $X$ is assigned a transmission policy, which in turn defines the set of packets $P_X$ transmitted during that segment. Since every slot belongs to at most one such segment, disjointness is guaranteed. Using the same argument, cycle allocations are disjoint, and can be combined into a global allocation.

For each cycle the division we propose is roughly shown in Figure 4. In this figure, slots in segment $X = \alpha, \beta, \gamma, \delta$ transmit all the packets $P_X$, which arrive in the delimited sections in the figure. This slot assignment is roughly described in the following manner:

- Segment $\alpha$ - all slots s.t. $C^{Tr}(t) < In^{Tr}$. These slots are grouped around the time handoff occurs.
Segment $\beta$ - monotonically increasing capacity, until the beginning of segment $\gamma$.

Segment $\gamma$ - $M_j^i$, where $j$ is chosen according to considerations described later on.

Segment $\delta$ - monotonically decreasing capacity, from the end of segment $\gamma$ until the next cycle.

From the definition of segment $\alpha$, and as portrayed in Figure 4, segments $\beta$, $\gamma$ and $\delta$ contain only slots for which the following condition is met:

$$C_{Tr}(t) \geq I_{Tr}^T.$$  \hspace{1cm} (5)

The exact boundaries between each pair of segments is defined in what follows. For now, observe that it is possible that under certain conditions, some segments do not exist. For example, segment $\alpha$ might not exist in a cycle with very low incoming data rate. Also, some slots might not belong to any of these segments. These slots are never allocated for transmission by our algorithm.

### 3.4 Buffer-and-Burst algorithm (B&B) specification

When viewed in the context of a single cycle, Buffer-and-Burst generates an allocation with a strongly intuitive structure. During the slots with the lowest capacity (Seg. $\alpha$) it allocates as few slots as is legally possible, allowing the buffer to fill up. Then (Seg. $\beta$) it aims only at keeping the buffer within legal parameters until the highest capacity slots in the cycle are reached (Seg. $\gamma$). A block of these slots is allocated in order to completely clear the buffer of
all pending packets. Finally, the remainder of the the cycle (Seg. \( \delta \)) is spent transmitting as few slots as possible while ensuring the buffer remains almost empty, in preparation for the next cycle. In this manner, transmission is focused as much as possible during the strongest slots in the cycle, a policy which obviously will decrease the total number of slots required for servicing the train users.

We now continue and present the locally optimal allocation for each packet set \( P_X (X = \alpha, \beta, \gamma, \delta) \). The resulting allocations are combined into a \( (1+\epsilon) \)-competitive allocation based on Theorem 2. The optimality proof is then completed by proving that every packet is transmitted legally by one of the transmission segments (Theorem 4).

### 3.4.1 Segment \( \alpha \)

Segment \( \alpha \) contains all the slots whose capacity is low and conforms to \( C_{Tr}(t) < In_{Tr} \), and these slots only. Let \( \alpha(-) \subset \alpha \) consist of the first \( k := T_{\alpha} - w_{max} \) slots, where \( T_X \) denotes the number of slots in segment \( X \). For the cases in which \( k > 0 \) we define \( P_{\alpha} \) as all the packets which arrive during \( \alpha(-) \), and in all other cases \( T_{\alpha} \) is the empty set.

**Lemma 2** The optimal allocation for transmission of \( P_{\alpha} \) allocates the highest capacity slots in \( \alpha \) until their total capacity \( \geq |P_{\alpha}| = k \cdot In_{Tr} \). If no such allocation exists, there is no legal allocation for some packet(s) in \( P_{\alpha} \).

The allocation is roughly depicted in Figure 5.

![Figure 5: \( \alpha \)-Policy](image)

**Proof:** If no allocation exists which can transmit all the packets in \( P_{\alpha} \) by the end of \( \alpha \), there is no legal allocation for some packets in \( P_{\alpha} \), since they all came at least \( w_{max} \) slots prior to the end of \( \alpha \). Also, since we are using the highest capacity slots of segment \( \alpha \), the
allocation is optimal. What is left to show is that, when some legal allocation does exist, the allocation described in the lemma is indeed always feasible and legal, i.e. that each packet can be transmitted within delay constraints by these allocated slots.

Denote \( \alpha = \{ t_1, \ldots, t_n \} \), \( \alpha(-) = \{ t_1, \ldots, t_k \} \), where \( t_i = t_1 + (i - 1) \). Also, let \( B&B_\alpha \) be the allocation of this segment as described in the lemma. Since during segment \( \alpha \) the incoming data rate is greater than the capacity at each slot, the buffer is constantly increasing in size regardless of any allocation which takes place during the segment, so every allocated slot is fully utilized regardless of the number of allocated slots. Let us assume by way of contradiction that there is some packet \( p \) arriving at \( t_i \in \alpha(-) \) which is not transmitted within its delay constraint by \( B&B_\alpha \). This means that by the end of slot \( t_i + w_{\text{max}} \), \( p \) has not been transmitted, and due to the fact that the capacity of all allocated slots is fully utilized, this means

\[
\sum_{t=t_1}^{t_i+w_{\text{max}}} C^{Tr}(t) \cdot B&B_\alpha(t) < i \cdot I_n^{Tr}.
\]

The total capacity of the allocated slots is \( \geq k \cdot I_n^{Tr} \), so we get that the remaining allocated slots, which come after \( t_i+w_{\text{max}} \), must conform to

\[
\sum_{t=t_i+w_{\text{max}}+1}^{t_n} C^{Tr}(t) \cdot B&B_\alpha(t) > (k - i) \cdot I_n^{Tr}.
\]

However, since \( T_\alpha - (i + w_{\text{max}}) = k - i \), this would require one or more of these remaining slots to have capacity greater than \( I_n^{Tr} \), contradicting the definition of segment \( \alpha \).

3.4.2 Segment \( \beta \)

Segment \( \beta \) consists of slots in the monotonically increasing section of the capacity cycle, beginning with the first such slot which abides by Eq. 5. Such a segment always exists, unless it is ”taken over” by segment \( \gamma \), and ends when segment \( \gamma^i \) begins. These issues shall be clarified later on when segment \( \gamma \) is defined.

In order to define the range of packets to be transmitted by segments \( \beta^i \) and \( \gamma^i \), define for each slot \( t \) in this section \( r^i(t) \) to be

\[
r^i(t) = \min_{t'} \forall \ t' \leq t'' \leq t \ C^{Tr}(t') \geq C^{Tr}(t'')
\]

(6)
Let $\text{start}(\beta^i)$ be the slot for which $r^i(t)$ is minimal, and define

$$cov^i = \max\{r^i(\text{start}(\beta^i)), \text{start}(\beta^i) - w_{\max}\}.$$  

In what follows we shall demonstrate that all packets arriving prior to $cov^i$ have already been transmitted by the time segment $\beta^i$ is reached. Note that $cov^i$ is a slot in segment $\alpha^i$ or in some previous cycle. Segment $\beta^i$ ends when segment $\gamma^i$ begins, as defined later on.

**Definition 3** (\textit{$\beta$-policy}) During segment $\beta$, transmit at slot $t$ only when a delay constraint will otherwise be violated at slot $t$ for some packet, i.e. when

$$B_{B&B}^T(t) > w_{\max} \cdot I_n^T.$$  

The motivation behind this policy in segment $\beta$ is that as time passes, the capacity available at each slot increases. Thus, by waiting till the last possible moment, we increase the packets-per-allocated-slot ratio, which improves the performance of the allocation. If we transmit packets arriving only after $cov^i$, this optimality is guaranteed also for all the buffered packets, since until transmission they do not pass a stronger slot. This is formally stated thus:

**Lemma 3** If $\beta$ transmits only packets arriving from $cov^i$ and on, $\beta$-policy yields an optimal allocation for all packets transmitted by it (denoted $P_\beta$).

**Proof:** First, we show that this is a legal allocation. Due to the fact that all capacities in segment $\beta$ conform to Equation 5, the arrival rate is lower than the capacity at every slot in this segment. This ensures that each allocated slot has the capacity to transmit all the packets which would have otherwise violated their delay constraint. What remains to be shown is that this allocation scheme is optimal, namely that packets $P_\beta$ could not be transmitted using less slots than those transmitted by Buffer-and-Burst.

Let $\text{first}_A(t)$ be the first packet transmitted by slot $t$ using allocation $A$. Optimality of our policy is shown by demonstrating that for any two slots $t_1, t_2$ allocated using this policy, $p_1 = \text{first}_{B&B}(t_1)$ and $p_2 = \text{first}_{B&B}(t_2)$ are transmitted in different slots no matter what alternative allocation is being used, so the resulting allocation using the above policy is of minimal size and thus locally-optimal.

Assume by way of contradiction that in some other allocation $p_1, p_2$ are transmitted in the
same slot $t$. $t \leq t_1$ since by the policy definition $p_1$ would violate its delay constraint after $t_1$. This also means $p_2$ arrives before $t_1$ (since we assume it is transmitted at $t \leq t_1$), so it is available for transmission at $t_1$. Since $p_1$ arrived earlier than $p_2$, $p_2$ is within its delay constraint at $t_1$. Therefore, since we use FIFO transmission policy and $p_1 = \text{first}_{B&B}(t_1)$, $p_2$ should also be transmitted at this slot when using $\beta$-policy, since $C^{Tr}(t_1) \geq C^{Tr}(t)$. This contradicts the given fact that $p_2 = \text{first}_{B&B}(t_2)$, so the contradicting assumption is refuted. 

3.4.3 Segment $\gamma$

Segment $\gamma^i$ is a minimal transmission block of type $M^i_j \subseteq M^{i}_{\text{max}}$, selected in such a manner that by its end the buffer is cleared of all packets arriving before $\text{cov}^{i+1}$. Formally, $\gamma^i = M^i_k$ s.t.

$$|B_{B&B}^{Tr}(M^i_k(1) - 1)| \leq Ex^{Tr}(M^i_k) + \max\{0, (M^i_k(k) - \text{cov}^{i+1}) \cdot \text{In}^{Tr}\}. \tag{8}$$

If there are several such blocks, we choose the shortest one.

Denote $P_{\gamma^i}$ to be all packets in the buffer right before this segment begins, in addition to all the packets arriving during $M^i_j$ before $\text{cov}^{i+1}$. These are all transmittable by the transmission block defined above as segment $\gamma^i$.

**Lemma 4** If $P_{\gamma^i}$ does not contain packets arriving before $\text{cov}^i$, then packets $P_{\gamma^i}$ are transmitted optimally.

**Proof:** Since we chose the minimal block of slots which meets the specified condition, we know all but the last slot transmit at full capacity. Combined with the fact that segment $\gamma^i$ is made up of the strongest slots in the cell, it is clear that any other allocation of slots within the cycle would require at least the same number of slots in order to transmit $P_{\gamma^i}$. Thus optimality within the cycle is proven.

Additionally, since we transmit only packets arriving in the interval $\text{cov}^i \leq t < \text{cov}^{i+1}$, these packets cannot be transmitted by future or past cells using a stronger slot, so the allocation is globally optimal.
In what follows we shall show that that design of Buffer-and-Burst is such that each cycle $i$ transmits all packets arriving until $\text{cov}^i$, thus ensuring that the conditions of Lemmas 3 and 4 regarding the buffer are met and the allocation of $\beta^i, \gamma^i$ is optimal.

**Observation 2** Regardless of the buffer size at the beginning of segment $\gamma^i$, by its end the buffer has been emptied of all packets arriving prior to $\text{cov}^{i+1}$. Specifically, if $\gamma^i = M^i_k$ and $M^i_k(k) < \text{cov}^{i+1}$, the buffer will be cleared by the end of the segment.

### 3.4.4 Segment $\delta$

Segment $\delta^i$ includes all slots in the monotonically decreasing section of the cycle, located between segments $\gamma^i$ and $\alpha^{i+1}$, and which meets the following condition:

$$\forall \ t < t' < t + w_{\text{max}} \ C^{Tr}(t) \geq C^{Tr}(t')$$

We treat this segment in the following manner. We begin by making two observations regarding optimal policies in this segment (Lemmas 5 and 6), which lead us to formulate our transmission policy. The optimality of this policy is proven by way of induction (Lemmas 7 and 8): we show that at each of the allocated slots, our policy is no worse off than some optimal policy.

**Lemma 5** Let $t_1, \ldots, t_T$ be the slots of segment $\delta$ ($t_i := t_i + i - 1$), and let $\text{opt}$ be some optimal allocation. If $t_i$ is the last slot such that

$$B_{\text{opt}}^{\text{Tr}}(t_i) \leq \text{Ex}^{Tr}[t_{i+1}, t_T]$$

then allocating all the slots $t_{i+1}, \ldots, t_T$ for transmission results in an optimal allocation for all packets transmitted during this allocation block.

**Proof:** Since $t_i$ is the last slot for which Eq. 10 holds, we know that using a subset of this transmission block will not be enough to transmit the packets transmitted by this allocation block. From Equation 9 we know that these slots are stronger than all succeeding slots within legal delay. Therefore, since they are all (but the last) fully utilized by this allocation we may conclude that any alternative transmission of this set of packets will require at least the same number of slots.

---

4If $|\alpha^{i+1}|=0$, the segment ends at the next local minimum, i.e. the handoff slot.
Lemma 6 There exists an optimal allocation of packets arriving within segment $\delta$, for which the following two conditions are met:

- slot $t$ is not allocated if $C^{Tr}(t + 1) \geq B^{Tr}_{opt}(t) + In^{Tr}$ and $t + 1 \in \delta$
- slot $t$ is allocated for transmission if $C^{Tr}(t + 1) \leq B^{Tr}_{opt}(t)$.

Proof: Since we are dealing only with delay-bound capacity functions, the above transmission rules also ensure legality.

For the first case, we compare an allocation at slot $t$ to an allocation at the succeeding slot. Though in both cases a single slot is allocated, $In^{Tr}$ more packets are transmitted if slot $t + 1$ is allocated, thus reducing the number of remaining packets required for transmission at a later stage. Additionally, transmitting at slot $t + 1$ will clear the buffer completely, so the buffer state at the end of slot $t + 1$ is optimal and cannot be improved upon by transmitting at both slots $t, t + 1$. Thus, there is no reason to transmit at slot $t$ in such a case.

For the second case, from the definition of segment $\delta$ we know that all succeeding slots $t'$ within legal range ($= w_{max}$ slots) of slot $t$ will have $C^{Tr}(t') \leq B^{Tr}_{opt}(t)$. Therefore the specified condition implies that any allocation in the future that substitutes allocation of $t$ will require at least one entire slot dedicated for transmission of packets transmittable by slot $t$. By transmitting earlier we only clear the buffer sooner of these packets, which can only improve the allocation performance.

The question of where to transmit next is therefore limited to choosing between slots $t$ and $t + 1$ when

$$B^{Tr}_{opt}(t) < C^{Tr}(t + 1) < B^{Tr}_{opt}(t) + In^{Tr}$$

(11)

Definition 4 ($\delta$-policy) Allocate slot $t$ iff $C^{Tr}(t) \leq B^{Tr}_{B&B}(t)$, until the requirements of Lemma 5 are met. When this happens, allocate all the remaining slots of the segment.

Denote the resulting allocation from this policy as $B&B_{\delta}$. Note that the first part of the policy is equivalent to transmitting always at slot $t + 1$ in the case described in Eq. 11.

Theorem 3 $B&B_{\delta}$ is an optimal allocation of packets in $P_{\delta}$.
This theorem is proven by demonstrating that any allocation abiding by Lemmas 5 and 6, can do no better than this policy. Specifically, denote by \( t^* \) the slot in which \( B&B_δ \) last allocates before meeting the conditions for Lemma 5. We show that any optimal algorithm abiding by Lemma 6 will require the same number of allocated slots until slot \( t^* \), and will reach this slot with a buffer no smaller than the one achieved by \( B&B_δ \). From this we conclude \( δ \)-policy is optimal, since all policies can abide by Lemma 5 which allocates less slots the smaller the starting buffer is.

Denote by \( \text{slot}(p) \) the slot in which packet \( p \) arrived, and by \( \text{last}_{\text{alg}}(t) \) the last packet transmitted by slot \( t \) using allocation \( \text{alg} \).

**Observation 3** When \( \text{alg} \) abides by Lemma 6, we get \( \text{slot}(\text{last}_{\text{alg}}(t)) = t \).

Let \( u_1, ..., u_k \) be the slots allocated for transmission by Buffer-and-Burst, \( u_k = t^* \). Let \( v_1, ..., v_h \) be the slots allocated by some optimal allocation \( \text{opt} \) which conforms to Lemmas 5 and 6.

**Lemma 7** Let \( \text{alg}_1, \text{alg}_2 \) be two algorithms which abide by Lemma 6, and \( t_1 \) and \( t_2 \) are allocated by \( \text{alg}_1 \) and \( \text{alg}_2 \) (respectively). Then if \( \text{last}_{\text{alg}_1}(t_1) \leq \text{last}_{\text{alg}_2}(t_2) \), we get \( t_1 \leq t_2 \).

**Proof:** This is a direct result from observation 3.

**Lemma 8** If \( \text{last}_{\text{opt}}(v_i) \leq \text{last}_{B&B}(u_i) \) then \( \text{last}_{\text{opt}}(v_{i+1}) \leq \text{last}_{B&B}(u_{i+1}) \).

**Proof:** First, note that from Lemma 7 we know \( v_i \leq u_i < u_{i+1} \), so there is some packet transmitted by \( B&B \) at \( u_{i+1} \) and not by \( \text{opt} \) at \( v_i \), which ensures also \( \text{opt} \) will allocate some additional slot \( v_{i+1} \) for the transmission of these future packets.

Since \( \text{last}_{\text{opt}}(v_i) \leq \text{last}_{B&B}(u_i) \), we know that \( v_i \leq u_i \) and that if \( v_i = u_i \) then \( B_{B&B}^{Tr}(u_i) \leq B_{\text{opt}}^{Tr}(u_i) \), so after transmission at this slot \( \text{opt} \) does not have a smaller buffer than \( B&B \).

If \( \text{opt} \) would not transmit between slots \( v_i \) and \( u_{i+1} \), at slot \( u_{i+1} \) we would get \( B_{\text{opt}}^{Tr}(u_{i+1}) \geq B_{B&B}^{Tr}(u_{i+1}) \). Since \( B&B \) allocated this slot, we know that

\[
B_{B&B}^{Tr}(u_{i+1}) \geq C^{Tr}(u_{i+1})
\]

and therefore also

\[
B_{\text{opt}}^{Tr}(u_{i+1}) \geq C^{Tr}(u_{i+1})
\]

26
opt abides by Lemma 6, so opt will allocate some slot $\leq u_{i+1}$ and from Lemma 7 our claim is proven.

\section*{Proof of Theorem 3.} As opt abides by Lemma 6, and B&B always transmits at the later slot $(t + 1)$, we have the base of the inductive proof, namely $v_1 \leq u_1$. By way of induction, we get that $last_{\text{opt}}(v_k) \leq last_{\text{B&B}}(u_k)$, and thus B&B is optimal up to slot $u_k$, since any other allocation will require at least $k$ slots in order to transmit all the packets transmitted by B&B in $k$ slots.

Also, from Observation 3 it is clear that all packets in the segment not transmitted until $u_k$ (inclusive) arrived from slot $u_k$ and on, so they could not be transmitted earlier by some alternative allocation, and the optimality of their transmission is unaffected by the allocations until $u_k$. This is true also of the packets which arrived during $u_k$ and were not transmitted until later: On the one hand, there are less than $In^{Tr}$ such packets, so the price of their postponement is at most a single allocated slot. On the other hand, to transmit these packets earlier would require $k + 1$ slots allocated until $u_k$, as we know from the optimality of our policy until this slot. Thus, by postponing their transmission the allocation can only decrease in size.

Finally, from slot $u_k$ until transmission according to Lemma 5 the algorithm continues to follow the policy defined in Lemma 6, which ensures that refraining from transmission during this period is optimal. They are therefore transmitted in optimal fashion, so B&B is optimal. ■

We come now to the final stage in our analysis, in which we show that every packet is transmitted during one of the four segments discussed previously. To do so, we require a helping lemma:

\section*{Lemma 9} If $|P_{\alpha^{i+1}}| > 0$, segment $\alpha^{i+1}$ is adjacent to either segment $\delta^i$ or segment $\gamma^i$.

\section*{Proof:} Denote by $t_{\text{pre}}$ the last slot before segment $\alpha^{i+1}$, which ensures $C^{Tr}(t_{\text{pre}}) \geq In^{Tr}$ (Eq. 5). Since $|P_{\alpha^{i+1}}| > 0$, we know $|\alpha^{i+1}| > w_{\text{max}}$, the slots of which all have capacity lower than $In^{Tr}$. $t_{\text{pre}}$ thus meets the requirements of slots in $\delta$ (Eq. 9), so if $t_{\text{pre}}$ is not part of $\gamma^i$, $t_{\text{pre}} \in \delta^i$. ■
**Theorem 4** Every packet \( p \) is transmitted during one of the segments.

**Proof:** Assume \( cov^i \leq slot(p) < cov^{i+h} \), such that all other \( cov^i \) slots are not in this interval. Wlog, \( h = 1 \). Denote

\[
P_{i,\beta\gamma\delta\alpha} = \bigcup_{X \in \{\beta^i, \gamma^i, \delta^i, \alpha^{i+1}\}} P_X.
\]

We prove the theorem by showing that \( p \in P_{i,\beta\gamma\delta\alpha} \).

In Lemma 9 we discussed the adjacency of \( slot \) segments. In a similar manner, two packet segments are said to be adjacent if the first packet of one immediately follows the last packet of the other. An empty segment is said to be adjacent to all segments. Based on the \( B&B \) policy, \( P_{\gamma^i} \) and \( P_{\gamma^i} \) are adjacent, as are \( P_{\gamma^i} \) and \( P_{\delta^i} \). From Lemma 9 we also know that \( P_{\alpha^{i+1}} \) is adjacent to either \( P_{\gamma^i} \) or \( P_{\delta^i} \). From here we conclude that if \( p \) is not included in \( P_{i,\beta\gamma\delta\alpha} \), it must arrive either before or after \( P_{i,\beta\gamma\delta\alpha} \).

By way of induction, assume all packets arriving during the until \( cov^i \). Let \( p_f, p_l \) be the first and last packets in \( P_{i,\beta\gamma\delta\alpha} \) respectively. \( slot(p_f) = cov^i \), so \( p \geq p_f \). If, on the other hand, \( p_l < p \), we get \( C^{Tr}(slot(p)) < C^{Tr}(t) \) for some \( slot(p_l) < t \leq slot(p_l) + w_{\text{max}} \) (otherwise, \( slot(p) \) would be included in segment \( \delta^i \)). Let \( t' \) be the slot with highest capacity within this range, and note that \( r^{i+1}(t') \leq slot(p) \).

\[C^{Tr}(t') \geq In^{Tr} \], since otherwise this would imply that \( |\alpha^{i+1}| > |\alpha^{i+1}(\gamma)| + w_{\text{max}} \), violating the definition of these segments. Therefore, since \( r^{i+1}(t') \leq slot(p) \) we get \( cov^{i+1} \leq slot(p) \).

This, however, would require \( p \) to exceed the range of slots in which \( p \) is said to arrive. We therefore know \( p_l \leq p \leq p_f \), so \( p \in P_{i,\beta\gamma\delta\alpha} \).

\[\blacksquare\]

### 3.5 Algorithm specification and complexity

**Algorithm Specification**

- Assumed knowledge of each base-station:
  - \( C^{Tr}(t) \) for all \( t \) \(^5\).

---

\(^5\)This is just a simplifying assumption. It is actually sufficient for each base station to have the capacity between \( cov^i \) and the first slot of segment \( \beta^{i+1} \)
Allowable delay $w_{\text{max}}$ for all packets.

- Offline (preprocessing)
  1. Assume $In^{Tr} = 0$, and define each cycle as the period between two consecutive capacity local minima.
  2. For each cycle $i$, find the first slot of segment $\beta^i$ and $cov^i$. Note that as long as we assume $In^{Tr} = 0$, slot $cov^i - 1$ is the last slot of $\delta^{i-1}$, so preliminary cycle boundaries have been defined by this step.
  3. For each cycle $i$ calculate the capacity of $M^j_i$ for all $j \leq |Cyc^i|$, and order these values according to increasing value of the first slot in $M^j_i$. ("key-value map")
  4. For each cycle $i$, calculate the total capacity of cycle suffixes, starting from the peak-capacity slot. Order these values according to increasing value of the first slot $i$ the suffix. ("key-value map")

- Online - Train is on the move
  1. For every $i$, locate the boundaries of $\alpha^i$, which is a straightforward process of finding the first and last slots in the cycle in which capacity is less than $In^{Tr}$.
  2. Use this information in order to redraw the boundaries of the adjoining segments, and calculate $cov^i$.
  3. During each segment, apply its respective policy once train arrives.

**Algorithm Complexity** The complexity of the online stage is $O(T_i)$ per cycle $i$, where $T_i$ is the number of slots in the cycle. Step 1 requires one sweep over the cycle capacities, and step 2 can be done in $O(1)$. Step 3 can be performed iteratively by selecting at each round the strongest slot not previously selected. Since the capacity in segment $\alpha$ first decreases and then increases, this process has linear complexity as it is equivalent to parallel sequential iteration over two sorted lists and choosing at each point the stronger value.

Regarding step 4, let us review the complexity during each segment. Slots in segment $\alpha$ are allocated in advance, so transmission during that segment has linear complexity. Slots in segment $\beta$ are allocated only when delay constraints are about to be violated, which can
be checked at each slot in constant time. Checking when to switch over to $\gamma$-policy can be done in $O(1)$ due to the preprocessing stage (offline step 3), and once policy transition has been performed all the slots are allocated until the buffer is empty. Finally, $\delta$-policy begins in $O(1)$ allocation policy, and ends with transmitting all the slots from some point and on. Once again, checking if the transition should be made within the segment can be done in $O(1)$ based on the information gathered in the preprocessing stage (offline step 4).

### 3.6 Implementation issues

We conclude this section with some short, high-level comments on the implementation of Buffer-and-Burst in practice. We focus here on three issues: train-tracks layout, base-station coordination and dealing with varying data arrival rates.

Though we have assumed thus far that the train tracks are laid out in a straight line, it is clear from the algorithm analysis that this is not a strict requirement. Minor curves in the tracks are to be expected in real-world scenarios, and are supported by the algorithm as described. The only requirement is that the basic wave-like pattern of the capacity function is maintained to the degree that the time-scale with which the capacity fluctuates remains slow enough that the algorithm can continue to cluster slots into cycles. This is a reasonable assumption, as train-tracks change direction in relatively wide angles.

With regards to base-station coordination, we note that this is required only in a limited fashion and mainly in the offline stage. In the online stage, base-station $i$ requires only the incoming data rate, $cov^i$ and $cov^{i+1}$, and then it can compute which packets it needs to transmit. By notifying the neighboring base-station(s) of the incoming data rate and the buffer state, they will be able to compute their allocation schemes as well.

Till this point we have assumed the incoming data-rate to be a constant, which is an approximately reasonable yet somewhat limiting assumption. In practice, slight changes in the incoming data rate are to be expected. Reviewing the Buffer-and-Burst policy, we note that such fluctuations must be dealt with differently in segments $\beta$ and $\delta$ compared to segments $\alpha$ and $\gamma$. The policy during the first couple is purely online, in that no global view of the capacity function or incoming data rate is required. The Buffer-and-Burst policy can, therefore, be applied as-is in both segments, and the optimality proofs can be shown to hold
here as well. In the second couple, and at the end of segment $\delta$, we require knowledge of upcoming capacity and incoming rate values in order to determine the optimal allocation. Since we are considering a scenario in which data-arrival rate fluctuations are unpredictable, one may use the expected arrival rate instead. When only slight variations in this rate are to be expected, this should ensure a legal transmission of all packets, though optimality may be slightly compromised.

4 The MASS Architecture for Capacity Increase

4.1 MASS Architecture

In the next two sections we strive to further improve the QoS experienced by the users on and off the train by exploiting the second unique property of the train, namely its significant length. This section is devoted to presenting a new architecture which utilizes this property of the train, and constructing its optimal configuration w.r.t. the general increase in the capacity available to users on the train. The next section discusses the manner in which this new architecture can be combined optimally with the $B&B$ policy.

Trains can reach considerable length, spanning several hundred meters, and as a result at any given moment in time the potential reception capacities at two distinct points along the train significantly vary from each other. Based on this observation, we propose a new train wireless architecture, termed Multi-Antenna Spatially-Separated (MASS), and roughly depicted in Figure 6. It consists of a multiplicity of reception antennas placed on the external train hull, with the base-station transmitting at each allocated slot only to the antenna with the highest reception capacity $6$, denoted $C^{Tr}(t)$. Denote the antennas mounted on the external hull $a_0, ..., a_{n-1}$, indexed in order of placement, where $a_i(t)$ is the location of antenna $a_i$ at slot $t$. We refer to $a_0$ and $a_{n-1}$ as the end antennas, and to all other antennas as the intermediate antennas. $d(a_i, a_j)$ represents the distance between antennas $a_i$ and $a_j$.

---

6One might propose to improve the system by allowing parallel transmission to all the antennas from different base-stations. This alternative policy is, however, not comparable to the presented MASS policy, since the increased bandwidth the train users would enjoy comes at the expense of requiring an additional base-station to transmit.
which is constant over time.

![Figure 6: MASS architecture](image)

Employing such a system would obviously increase the resources available to the train users, but raises the question of the optimal antenna deployment of these antennas over the expanse of the train. This section and the next deal with devising the optimal deployment of these antennas.

In the model we use here, where handoff between base-stations occurs only due to changes in the relative distance of an antenna to base-stations, employing an architecture such as MASS can raise difficult problems. Theoretically, the existence of multiple antennas could cause a rapid succession of handoffs back and forth between two neighboring base-stations, as at each slot a different antenna can become closer to a different base-station and require handoff. This would create additional load on the base-stations involved and tend to disrupt communications to the train at a high frequency. In order to avoid such scenarios, the deployment of the antennas on the train should be such which would ensure a monotonic handoff sequence. By this we mean that over the course of the train’s passage through the network, each base-station performs at most a single handoff to some other base-station. The following lemma defines a set of antenna deployments which ensures this property, solely based on the position of the two end antennas:

**Lemma 10** Let \( x_{d_j} \) be the position on the tracks closest to \( s_j \), and \( t_{d_j}^i \) be the first slot \( t \) where \( a_i(t) \geq x_{d_j} \). If for all \( j \) a handoff occurs at slot \( t \) which complies with

\[
t_{d_j}^{i-1} \leq t \leq t_{d_{j+1}}^i
\]

then the handoff sequence is monotonic.
Proof: The condition specified in Eq. 12 implies that the handoff occurs when all the antennas are within the range \([x_{d_j}, x_{d_{j+1}}]\). This ensures the first handoff switched control from \(a_{n-1}\) to \(a_0\), since until \(t_{d_{j+1}}^0\) \(a_0\) is closer to \(s_{j+1}\) then all other antennas. Thus, the handoff is monotonic if we can prove that after this handoff no transmission takes place from \(s_j\) to the train.

Since \(a_{n-1}\) is the last to be transmitted to via \(s_j\), from the first handoff and on it remains the closest antenna to \(s_j\). Therefore, in order to verify that \(s_j\) is not revisited, it is enough to show that \(a_0\) will always be closer to \(s_{j+1}\) then \(a_{n-1}\) will be to \(s_j\) from after the first handoff.

Let \(x_j = a_{n-1}(t) - x_{d_j} > 0\) and \(x_{j+1} = a_0(t) - x_{d_{j+1}}\), at the point \(t\) where \(a_0\) is stronger than \(a_{n-1}\). At the handoff we know

\[
d_{j+1}^2 + x_{j+1}^2 \leq d_j^2 + x_j^2
\]  

(13)

If \(x_{j+1} \leq x_j\), we can ensure \(s_{j+1}\) remains preferable to \(s_j\): after some \(x\) slots pass,

\[
d_{j+1}^2 + (x_{j+1} + x)^2 = (d_{j+1}^2 + x_{j+1}^2) + (x^2 + 2 \cdot x \cdot x_{j+1}) \leq (d_j^2 + x_j^2) + (x^2 + 2 \cdot x \cdot x_j) \leq d_j^2 + (x_j + x)^2.
\]

So all that is left is to prove that \(x_{j+1} \leq x_j\):

- If \(d_{j+1} \geq d_j\), from equation 13 we know \(x_{j+1} \leq x_i\).
- Otherwise \(d_{j+1} < d_j\). We’ve seen that from the condition of the lemma it follows that \(d(x_{d_j}, x_{d_{j+1}}) > d(a_0, a_{n-1})\). Since \(s_{j+1}\) is closer to the tracks than \(s_j\), it will support higher capacity than the highest supplied by \(s_{j+1}\), before \(a_0\) reaches its peak at \(x_{d_{j+1}}\), so \(x_{j+1} < 0 \leq x_j\).

Reviewing the requirements of Lemma 10, we note that a monotonic-handoff antenna deployment can be achieved by correct positioning of the end antennas alone, and that changes in the positions of the intermediate antennas has no effect on this property of the deployment. This implies that throughout the duration of the train in a given cell, it switches control to each and every one of its antennas, a property which we shall term here complete. It is possible, however, to construct an antenna deployment which ensures monotonic handoff
but does not meet the requirements of Lemma 10. Such alternatives would cause some base-
stations to transmit only to a sub-set of \( \{a_0, ..., a_{n-1}\} \), depending on the deployment of the
intermediate antennas (i.e. an incomplete switching sequence). Such antenna deployments
are outside the scope of this work.

In what follows we shall therefore commonly differentiate between the positioning of the
end antennas and that of the intermediate ones. Specifically, since the position of the end
antennas is determined by the length of the train, the network structure and the layout of the
train tracks through the network, we tend to focus here only on optimizing the deployment
of the intermediate antennas.

In this section, we are interested in maximizing the following two natural performance
measures, with respect to the antenna configuration:

- **Min Slot-Capacity** - \( \min_{t \in \{1, ..., T\}} C_{Tr}(t) \).

- **Total Slot-Capacity** - \( \sum_{t \in \{1, ..., T\}} C_{Tr}(t) \).

In this work we demonstrate the uniform antenna deployment - an antenna deployment
where the distances between all neighboring antennas are identical - to be optimal for both
performance measures previously defined, for monotonic-handoff antenna deployments.

In a monotonic-handoff antenna deployment which abides by Lemma 10, all transmission
to the train between the handoff slots of the end antennas is covered by a single base-
station. Thus, any analysis of such systems can focus on the performance of a given antenna
deployment with regards to a single base-station, and then combine the results. In addition,
by focusing on the capacity available to the train from a single base-station, note that all
the antennas move along the same route and therefore the capacity fluctuations experienced
by each antenna are identical, offset by several slots.

### 4.2 Notation

We adapt the previously used notation and denote by \( C_i^j(t) \) the reception capacity of antenna
\( i \) relative to base-station \( j \) at time \( t \), omitting the base-station index when dealing with a sin-
gle base-station. Since we transmit only to the train antenna with highest reception capacity,
the overall capacity of the train at time \( t \) is defined as \( C_{Tr}(t) = \max_{j \in \mathbb{Z}} \max_{0 \leq i \leq n-1} C_i^j(t) \).
4.3 Min Slot-Capacity

When viewing the capacity fluctuations of the train throughout its passage through the network, the minimal capacity value is reached at a slot which is either a handoff slot or a slot in which the train switches between antennas. The handoff slots are determined (in monotonic-handoff antenna deployments) by the positioning of the end antennas, while the antenna-switching slots are determined by the relative distance between neighboring antennas. Here we are interested in optimizing the deployment of the intermediate antennas, maximizing the minimal capacity available to the train between the peak capacity slots of the end antennas. Formally, for each base-station $i$, we are interested in maximizing the minimal capacity only during the interval $t^0_i, ..., t^{n-1}_i$.

**Theorem 5** Given a positioning of the end antennas which abides by Lemma 10, the uniform antenna deployment maximizes the Min Slot-Capacity measure between peak capacity slots of the end antennas.

**Proof:** The minimal capacity slot of each configuration is a slot during which antenna-switching occurs, where the distance from the base station to two adjacent antennas is equal. Specifically, from Pythagoras we know this to occur between the two antennas with the largest distance between them on the train. Therefore, achieving $\max_{i \in \{1,...,T\}} \min_{t \in \{1,...,T\}} C^{Tr}(t)$ is equivalent to achieving $\min_{i \in \{1,...,n-1\}} d(a_i, a_{i+1})$, which is reached in the uniform antenna deployment.

4.4 Total Slot-Capacity

In a monotonic-handoff deployment which abides by Lemma 10, each base station can be dealt with on its own. Focusing on a single base-station, we note that once MASS switches between $a_i$ and $a_{i+1}$, $a_i$ no longer receives transmission from the base-station, since from that point on $a_{i+1}$ is always closer to the base-station. We note that both switches which involve $a_i$ must occur somewhere between the peak capacity slots of its neighboring antennas, $a_{i-1}$ and $a_{i+1}$. Thus, moving $a_i$ between it’s neighboring antennas, $a_{i-1}$ and $a_{i+1}$, can only affect the capacity available to the train between the respective peak capacity slots of those antennas.
In the following lemma we show that moving \( a_i \) closer to the mid-point between its two neighbors can only increase the total capacity available to the train.

**Lemma 11** Let \( A = \{a_1, a_2, a_3\} \) be three adjacent train antennas placed according to index. Let the distance \( d(a_1, a_3) \) be set, and assume \( d(a_1, a_2) - d(a_2, a_3) = \epsilon > 0 \). For some \( \rho \leq \epsilon/2 \), define \( a'_2 := a_2 - \rho \) an antenna position which is closer than \( a_2 \) to \( a_1 \). Then using \( a'_2 \) instead of \( a_2 \) increases the total available capacity of a train moving at constant speed.

**Proof:** As stated earlier, the exact position of \( a_2 \) can affect the values of \( C^T(t) \) only between the peak capacities of \( a_1 \) and \( a_3 \). So let \( p_1 = d(a_1, a_2)/2 \) and \( p_2 = d(a_2, a_3)/2 \). Denote by \( d \) the distance of the base-station from the tracks and \( x \) the location of the train on the tracks as a function of time. Since all the antennas go through the same capacity fluctuations, the performance gain by repositioning as described is (see Figure 7)

\[
- \frac{2}{\rho} \int_{p_1-\rho/2}^{p_1} (d^2 + x^2)^{-\eta/2} dx + \frac{2}{\rho} \int_{p_2}^{p_2+\rho/2} (d^2 + x^2)^{-\eta/2} dx = \\
- \frac{2}{\rho} \int_{p_1-\rho/2}^{p_1} (d^2 + x^2)^{-\eta/2} dx + \frac{2}{\rho} \int_{p_1-\rho/2}^{p_1-\epsilon/2+\rho/2} (d^2 + x^2)^{-\eta/2} dx = \\
- \frac{2}{\rho} \int_{p_1-\rho/2}^{p_1} [(d^2 + x^2)^{-\eta/2} - (d^2 + (x + (-\epsilon + \rho)/2)^2)^{-\eta/2}] dx,
\]

which is greater than zero since \( \epsilon > \rho \) and \( \eta \geq 2 \).

![Figure 7](image-url)  

Figure 7: The circled areas correspond to the actual changes in the total capacity. The dotted lines refer to the situation after the intermediate antenna is shifted towards the middle.

**Theorem 6** Given that the end antennas are positioned in some monotonic-handoff deployment according to Lemma 10, then the uniform antenna deployment maximizes the Total Slot-Capacity measure.
Proof: In any non-uniform setting, from Lemma 11 there exists a triplet for which the middle antenna can be moved to improve overall capacity, so the optimal deployment is the uniform.

\[ \text{4.5 Symmetric model} \]

Until now we did not present a way in which to determine the optimal positioning of the end antennas, as this would require considering the exact topology of the network which may be very complex. Here we analyze a simplified model, for which we design the optimal antenna deployment for both end and intermediate antennas. The properties of this simplified model are: 1) All base-stations are at an equal distance \( d \) from the train tracks, and w.l.o.g. are positioned the same side of the tracks. They thus coincide with the line \( y = d \), parallel to the \( x \)-axis, and 2) Distances between neighboring stations are all equal to each other (\( = z \)). A model where these conditions are met will be called a symmetric model.

In the symmetric model, each antenna spends exactly \( z \) slots in each cell, and it’s capacity fluctuations repeat themselves at a frequency of \( z \) slots. The capacity at slot \( t \) of \( a_{n-1} \) is therefore equal to that of an imaginary antenna, placed at distance \( z \) from \( a_{n-1} \). It is therefore clear that placing an additional antenna at \( a_{n-1} - z \) (i.e. in the direction of \( a_0 \)) cannot improve the performance of the system, since only one antenna is transmitted to at any given moment and both antennas have identical capacity at every slot.

Given some monotonic-handoff deployment of antennas, we now add an antenna, denoted \( a_{-1} \), placed at said distance \( z \) from \( a_{n-1} \). The train never switches to this antenna, instead switching to \( a_{n-1} \) which is of identical capacity whenever these antennas have the strongest capacity among antennas. In a symmetric network, Lemma 10 requires that the distance between the end antennas should be \( \leq z \) in order to remain monotonic, so the resulting deployment ensures monotonic-handoff and antennas \( a_{-1} \) and \( a_{n-1} \) are the end antennas. By applying Theorem 6 to this new set of antennas, we know that the uniform antenna deployment is optimal, which leads to the conclusion that \( d(a_0, a_{n-1}) = z \cdot (1 - \frac{1}{n}) \).

The above antenna deployment may, of course, be impractical when the train length is less than \( z \cdot (1 - \frac{1}{n}) \). In cases where the train length is shorter than this distance, according to Lemma 11 we benefit even just by moving \( a_0 \) closer to the mid-point between its two
neighboring antennas. Since for any uniform antenna deployment of $a_0, ..., a_{n-1}$, the train length imposes

$$d(a_{i-1}, a_0) > \frac{z}{n} > d(a_0, a_1),$$

the optimal location for $a_0$ and $a_{n-1}$ is at the edges of the train, where $a_0$ can be closest to this mid-point. A similar argument can show that $a_{n-1}$ should be at the opposite edge of the train. This yields the following theorem:

**Theorem 7** (1) In the symmetric model, the optimal antenna deployment over a train of length $l_n \geq z \cdot (1 - \frac{1}{n})$ is a uniform deployment s.t. $d(a_i, a_{i+1}) = \frac{z}{n}$.

(2) In the symmetric model, the optimal antenna deployment over a train of length $l_n < z \cdot (1 - \frac{1}{n})$ is a uniform deployment over the whole body of the train.

## 5 Buffer-and-Burst and MASS combined

Though Buffer-and-Burst is optimal regardless of the specific delay constraints required, for short delay constraints the demands it would impose on the system would still cause large disruptions in the QoS experienced by the users in the cell, making its applicability somewhat limited. We therefore strive to exploit the positive characteristics of the proposed MASS architecture, to further expand the range of delay constraints and incoming data rates which could be supported by Buffer-and-Burst. Intuitively, the increased capacity which is available to the train as a result of employing MASS could support higher volumes of incoming data, and a reasonable deployment of these antennas would do much to eradicate the times in which the capacity is extremely low.

Obviously, the performance of any given slot allocation does not worsen as a result of this combination, as it only increases the available capacity at any given slot. Any optimal slot allocation for the single-antenna model is therefore a legal slot allocation for any deployment of MASS antennas. However, there might be hidden costs to the combination of this architecture and a given transmission policy. In the case of the Buffer-and-Burst policy, the MASS architecture with $n$ antennas will cause $B&B$ to sub-divide each cycle into $n$ sub-cycles over which the policy is performed. The possible deviation from the optimal allocation
in this new deployment rises therefore to 4n additional slots per cycle (4 per sub-cycle). This implies that for extremely large n there may be a point in which adding antennas no longer improves the performance of Buffer-and-Burst, and might indeed increase the size of the slot allocation.

Even when dealing with a smaller number of antennas, for which the capacity gain greatly improves the resulting allocation, there are numerous configurations in which to deploy them, and applying Buffer-and-Burst to each may yield a different cost. This section is devoted to devising the $(1 + n \cdot \epsilon)$-competitive (relative to the optimal) deployment of these antennas, and presenting the approach which we use to prove this optimality. A detailed proof is left for future research.

Our claim, which is presented formally below, is that the uniform antenna deployment presented in section 4 yields allocations with a quasi-optimal cost per sub-cycle. To prove this claim, we compare the resulting slot allocation given some arbitrary deployment to the slot allocation over the uniform deployment, each of which must transmit the same $P$ packets overall. For each deployment and for some constant $k$ we partition $P$ into $k$ disjoint sets of consecutive packets, such that a pairing exists between the two partitions of equal-sized sets. Formally,

$$P = \biguplus_{i=1}^{k} P_i = \biguplus_{i=1}^{k} P'_i$$

$$\forall \ 1 \leq i \leq k \ |P_i| = |P'_i|$$

where $\{P_i\}_{1 \leq i \leq k}$ is the partition for the uniform deployment. Denote by $\text{slots}_D(P)$ the number of slots required to transmit packets $P$ using deployment $D$. This partitioning and pairing will be constructed in such a way that for every $i$, the number of slots allocated in the uniform deployment to transmit $P_i$ exceeds the number required by the competing deployment to transmit $P'_i$ by no more than a single slot. Since $k$ is a constant, it is possible to deduce that for the entire cycle and some constant $c \leq k$,

$$\sum_{i=1}^{k} \text{slots}_{\text{uni}}(P_i) \leq \sum_{i=1}^{k} \text{slots}_{\text{rnd}}(P'_i) + c \cdot n$$

**Theorem 8** When using $B&b$ over a monotonic placement of the end antennas, the uniform deployment yields an allocation which is sub-optimal only by an additive factor of $O(n)$
slots per cycle, where \( n \) is the number of antennas.

5.1 Preliminaries

We begin by sketching out the family of deployments over which we attempt to prove optimality.

- As our model is discrete, we concern ourselves here with discrete distances of \( x_{\text{slot}} \)-granularity between antennas. A trivial implication of this is that for each two antennas \( i, j \), there exists a \( k \in \mathbb{Z} \) s.t. \( C^i(t) = C^j(t + k) \).

- Next, continuing the approach used in the previous section, we consider here only deployments which ensure monotonic-complete handoff between base-stations. Under this assumption, the analysis of any given deployment can be limited to the single base-station view, and results can be concatenated. Without this assumption the problem becomes very difficult, and the optimal deployment will be very sensitive to changes in the base-station configuration.

- In order to prove the optimality of a deployment, one must consider all possible delay constraints and incoming data rates (\( In^{Tr} \)). One scenario which can be dealt with easily is that where \( In^{Tr} \) is very low and/or \( w_{\text{max}} \) is very high, to the degree that \( B&B \) will require transmission only during segments \( \gamma \) for the uniform deployments. Such an allocation can be shown to include only the strongest slots globally, and is thus obviously optimal, with possible additive cost of \( n \) due to under-utilization of a single slot in each sub-cycle.

Our interest in the forthcoming discussion is therefore in other, more complex scenarios, in which transmission will take place in other segments as well in the uniform antenna deployment. We further assume that any competing, arbitrary deployment is such that for every sub-cycle \( i \),

\[
\gamma^i = M_{\text{max}}^i
\]  

(14)

and more specifically that the buffer is full when reaching this segment. This does not limit the generality of the Theorem, as we demonstrate in the Appendix (Lemma 17).
We define a slightly modified behavior of $B&B$. These modifications are done solely for accounting purposes in the analysis which is to follow. The modifications are two:

- As we recall, in segment $\delta$ there is an internal policy switch, from which point and on all the slots are allocated until the end of the segment. We slightly modify the manner in which packets are transmitted after this switch: the first of these slots does not transmit in full. Rather, it transmits only enough packets to ensure that all the remaining slots in the segment will fully utilize their capacity.

- In segment $\beta$, transmission takes place only to avoid delay constraint violation. In the last such allocated slot, we transmit only enough packets so that by the following segment $\gamma$ the buffer is completely full.

It can easily be verified that after these changes do not affect the quasi-optimality of the allocation. This variation of the policy, however, has two useful properties which we rely on:

1. In each segment, at most one allocated slot is under-utilized.

2. If $|\beta^i| > 0$, we arrive at $\gamma^i$ with a full buffer.

Finally, in order to simplify the notation of comparing allocations over different deployments, we define the process of index alignment as a shift in the slot indices of one deployment (allowing for negative indices in the process). The natural shift in packet indices follows such a shift. An antenna-based index-alignment is an index alignment after which the peak capacity slot of some given antenna $i$ has the same index in both deployments. Given a pair of $x_{slot}$-granularity deployments, denoted $D1, D2$, such an index-alignment ensures that for every slot, the capacity of antenna $i$ is the same in both antenna deployments for every slot:

$$\forall t \ C^{iD1}(t) = C^{iD2}(t).$$

We discuss the method of proving the Theorem first for the case of 3 antennas, where the two end antennas are set in place and only the middle one is moved between deployments. The manner in which these proofs can be expanded to the general case of $n$ antennas is then discussed.
5.2 3 antenna deployment

Each antenna is assigned a sub-cycle, such that sub-cycle $i = 0, 1, 2$ is the period in which antenna $a_i$ reaches its peak capacity w.r.t the single base-station. We call the interim in the train capacity function between two consecutive peak capacities, w.r.t. a given base-station, a capacity valley, denoted CV. (Figure 8). In the case of three antennas deployed over the train, there are two such capacity valleys.

Let $D_1$ and $D_2$ be two deployments which differ only in the position of the intermediate antenna. When compared to each other, the segmentation (i.e. division into segments for $B&B$) of the resulting sub-cycles differ, in that each segment may be shorter in one deployment than in the other (see Figure 9). Specifically, a segment $\chi = \alpha, \beta, \delta$ in some deployment (wlog $D_1$) will be called short ($\chi_{\text{short}}$) if it resides within a capacity valley which is shorter than its counterpart in the other deployment (wlog $D_2$). In the same way we label...
segments in a longer CV as long ($\chi_{\text{long}}$). $P_{\chi_{\text{long}}}$, $P_{\chi_{\text{short}}}$ are the packets transmitted during these segments. It can easily be shown that all segments in a short CV are not longer than their counterpart in a long CV. Also note that this classification is not defined to segments of type $\gamma$ since it is not necessary for the forthcoming analysis. Specifically, as long as Eq. 14 is maintained over all deployments, all segments of type $\gamma$ are of maximal size and do not vary in size between antenna deployments.

We want to compare the allocation efficiency during each segment in both deployments. This comparison is performed in the following manner. Let $x$ be the number of packets transmitted during some short segment. We look at the $x$ packets transmitted most efficiently in its long counterpart, and calculate the number of slots this transmission requires. We show that for any two deployments, there is only a slight difference in the number of allocated slots between deployments. The remaining packets, those transmitted in the long segment in relatively less efficient manner, are separately shown to be transmitted by the uniform deployment with another small constant cost.

This is formally done here via the packet partitioning as laid out in the introduction to this section. Each segment $\chi \in \{\alpha, \beta, \delta\}$ is assigned an index $i = 1, 2, 3$ respectively, and for each such index we define the following division:

1. $P_i = P_{\chi_D^{\text{short}}}$.  
2. $P'_{i+3} = \text{a set of } |P_i| \text{ packets transmitted most efficiently in } \chi_{D_2}^{\text{long}}$.  
3. $P'_{i+6} = P_{\chi_D^{\text{long}}} \setminus P'_{i+3}$.  

The packets included in $P_{i+3}$ are easily characterized per segment. When the capacity is increasing, as in segment $\beta$ ($i = 2$), these are the last slots of the segment, and the first slots in segment $\delta$ ($i = 3$) where capacity decreases. For segment $\alpha$ ($i = 1$) these are the slots at the beginning and end of the segment. See figures 10-12.

**Lemma 12** When dealing with $x_{\text{slot}}$-granularity deployments, $C_{\text{Tr}}(\alpha_{\text{long}}) \supseteq C_{\text{Tr}}(\alpha_{\text{short}})$.

**Proof:** $x_{\text{slot}}$-granularity deployments ensure each antenna experiences the same capacity values regardless of its location. Since segment $\alpha$ begins for both deployments with the same
capacity (by definition of the segment), shifting antennas can only change the set of capacities in segment $\alpha$ by causing the antenna switch to occur sooner. Since the antenna switch in the long segment occurs later, the lemma is proven.

![Figure 10: Slot division for segment $\alpha$](image)

**Lemma 13** (*Comparing segments $\alpha$*)

$$\text{slots}_{D_1}(P_4) = \text{slots}_{D_2}(P'_1).$$

**Proof:** From Lemma 12 we know $C^{Tr}(\chi_{long}) \supseteq C^{Tr}(\chi_{short})$ for all segments. Since $\alpha$-policy is to allocate all required slots in decreasing order of capacity, the lemma is proven.

Moving on to segment $\beta$, we say that two allocations $A_0, A_1$ are *interleaved* over some slot interval if, between every two slots allocated in $A_i$, $t'_1 < t'_2$, there is an allocated slot in $A_{1-i}$, $t'_1 \leq t^{1-i} < t'_2$. In any two interleaved allocations, it is easy to see the number of slots allocated in one is at most 1 slot more than those allocated in the other.

**Lemma 14** (*Comparing segments $\beta$*) Let $A_{D_1}, A_{D_2}$ be the respective allocations which transmit $P_5$ and $P'_2$ and which, as we recall, are of equal size. Then these two allocations are interleaved.

**Proof:** First we apply an antenna-based index-alignment on the antenna to which these allocations relate, which aligns the packet indices as well. Note that after the index alignment in both allocations the first packet transmitted by $\gamma$ is identical (Lemma 17), so the last packet transmitted by both allocations is identical as well. Denote $t_1 \in A_{D_1}, t'_1 \in A_{D_2}$
to be the first allocated slots in each allocation. Wlog let $t_1 \leq t_1'$. There exists some packet $p$ which is transmitted by both allocations in their first allocated slot, and since $\beta$-policy requires transmission only in order to avoid illegal packet delay, this implies

$$\text{first}_{A_{D1}}(t_2) \geq \text{first}_{A_{D2}}(t_1')$$

and thus $t_2 \geq t_1'$.

This line of reasoning is applied in an iterative fashion, so the two allocations continue to be interleaved as long as there are packets to be transmitted. Finally, if the last packet is transmitted by $t_k \in A_{D1}$, the following slot in $A_{D2}$ transmits the last packet as well, since its capacity is no less than than $t_k$. This concludes the proof, as both allocations transmit the same packets.

**Lemma 15 (Comparing segments $\delta$)** Let $A_{D1}, A_{D2}$ be the respective allocations which transmit $P_6$ and $P_3'$ and which, as we recall, are of equal size. Then $A_{D1}$ transmits the packets transmitted by $A_{D2}$ using at most $|A_{D2}| + 1$ slots.

**Proof:** First we apply an antenna-based index-alignment on the antenna to which these allocations relate, which aligns the packet indices as well. Note that in both allocation the last packet transmitted by $\gamma$ is identical (Lemma 17), so the first packet transmitted by both allocations is identical as well.

Let $A'_{D1}$ be the allocation for $P_6 \cup P_9$, and define a new allocation $A^*$ to transmit packets $P_6 \cup P_9$ in the following way: packets $P_6(=P_3')$ are transmitted using $A_{D2}$, and the remaining packets are transmitted in the same way they are transmitted in $A'_{D1}$.
Figure 12: Slot division for segment $\delta$

Since $A^*_D$ is locally optimal, $|A^*| \geq |A^*_D|$. Since the suffix of both allocations is identical, and based on the same logical principle used in Theorem 2, the number of slots allocated in $A^*_D$ for transmission of $P'_3$ is at most one more than those required in $A_D$, due to possible slot overlap with the suffix of $A^*_D$. 

The final component in proving our Theorem relates to the strength of the slots allocated for transmitting the remaining packets, namely those of type $P'_{i+6}$ (See Figure 13):

**Lemma 16** For every $i = 1, 2, 3$ $P'_{i+6}$ are transmitted by slots of capacity no greater than the capacity of any slot allocated for transmitting $P_{i+6}$.

A sketch of the proof can be found in the appendix.

**Proof of Theorem 8:**

- From Lemma 13:

  \[ slots_{D1}(P_1) = slots_{D2}(P'_4) \]

- From Lemma 14:

  \[ slots_{D2}(P'_2) - 1 \leq slots_{D1}(P_3) \leq slots_{D2}(P'_2) + 1 \]

- From Lemma 15:

  \[ slots_{D2}(P'_3) - 1 \leq slots_{D1}(P_6) \leq slots_{D2}(P'_3) + 1 \]
The same can be shown when switching between $P'_i$ and $P_i$ for each $i$, which brings us to

$$\sum_{i=1}^{6} slots_{D_1}(P_i) \leq \sum_{i=1}^{6} slots_{D_2}(P'_i) + 4$$

What remains is therefore to assess the cost of transmitting packets $P_{rem} = P_7 \cup P_8 \cup P_9$, and their counterparts $P'_{rem}$. It is here that we finally make use of the fact that we are comparing a random deployment (wlog $D_2$) to a uniform deployment (wlog $D_1$).

![Figure 13: Residue-packets transmission comparison](image)

The uniform deployment results in a periodical allocation, which means as well the the weakest allocated slot in both CVs of the uniform deployment are of identical capacity, denoted $c_{min}$. Packets in $P'_{rem}$, on the other hand, are all transmitted by slots of capacity $\leq c_{min}$. This can be deduced from Lemma 16 and from the observation that in $B&B$ the weakest allocated slot in both segments $\beta^i$ and $\delta^i$ is of the same capacity. Thus the transmission in the uniform deployment is not worse off. Since all but 3 slots are under-utilized (since this packet set is combined from 3 sets), this ensures that

$$slots_{uni}(P_{rem}) \leq slots_{rnd}(P'_{rem}) + 2$$

so for $c = 7$ the Theorem is proven. ■
5.3 $n > 3$ antennas

In order to show that the 3-antenna case analysis is applicable to the general case of $n > 3$ antennas, we need to devise a system in which the 3-antenna proof can be expanded to the general case. A simplistic approach for utilizing the 3-antenna case would be to prove the quasi-optimality of the uniform deployment through a sequence of ”virtual steps”, each one consisting of moving a single antenna to the mid-point between its neighbors (as was done in section 4). We compare the deployments before and after the virtual step, and using the 3-antenna proof conclude the virtual step has cost only $O(1)$. We proceed to move additional antennas, each time comparing to the deployment reached by the previous virtual step. Since we deal here in a discrete model, the number of steps required until the uniform deployment is reached is bounded.

The problem with such an approach is that the number of such virtual steps which would be required might be very large. The bounded cost of moving to the uniform deployment would then be very high, making the claim for quasi-optimality void. A different approach is required, therefore, if the case of $n = 3$ is to be put to use.

We illustrate here such a system, which uses tools and structure similar to that taken previously in this paper and is thus presented in high-level only. Our goal is to reach the uniform deployment in $O(n)$ deployment comparisons (“virtual steps”), thus resulting in a deviation from the optimal allocation of at most $O(n)$. To do so, we want to move not one but several antennas during each virtual step, yet treat it as if only one antenna has moved for analysis purposes. It can be shown that moving several adjacent antennas in the same direction while maintaining the relative distances between each of these antennas does not affect the allocation during the capacity-valleys in between antennas of this group when Equation 14 holds. We use this to define the following, ”virtual” progression. In each virtual step, we look for one of the following situations, one of which must occur in every non-uniform deployment:

- $i$ is the minimal index for which $a_i$ is to the left of its position in the uniform deployment, $j$ is the maximal index for which all $a_i \leq a_h \leq a_j$ are to the left of their position in the uniform deployment.
• $i$ is the minimal index for which $a_i$ is to the right of its position in the uniform deployment, $j$ is the maximal index for which all $a_i \leq a_h \leq a_j$ are to the right of their position in the uniform deployment.

If both situations occur one such couple can be chosen arbitrarily. Given such $i, j$, we move all antennas $a_i...a_j$ as a group until either $a_i$ or $a_j$ reach their position in the uniform deployment, and the resulting deployment is the next virtual step. It can be shown that each such step has the following properties:

1. In each virtual step, one antenna reaches its position in the uniform deployment.

2. Future virtual steps will not shift the position of the antenna which reached its destined position, limiting the number of such steps to $O(n)$.

3. Since all the antennas $a_i...a_j$ were moved as a group, the only changes in the allocation cost are in the capacity valleys immediately to the left of $a_i$ and immediately to the right of $a_j$. At the same time, wlog $a_i$ is left of its position in the uniform deployment, then $a_i$ and $a_j$ were chosen in such a manner that we know $d(a_{i-1}, a_i) < d(a_j, a_{j+1})$. Thus, the same analysis presented previously for the case of 3 antennas can therefore be applied in slightly modified manner to this virtual step, where the capacity valleys of interest are those between peak capacities of $a_{i-1}$ and $a_i$, and $a_j$ and $a_{j+1}$.

6 System Evaluation

6.1 The effects of MASS

In order to assess the possible gain to be made by employing the MASS architecture, Figures 14 and 15 present the gain w.r.t. the total-capacity measure. Figure 14 represents the factor of capacity increase experienced by the train when using two antennas in a symmetric model, as a function of the $z/d$ ratio, where $z$ is the distance between bas-stations and $d$ is the minimal distance between each base-station and the train tracks. Figure 15 does the same for the hypothetical situation of an infinite number of antennas, a hypothetical scenario in which every point along the train has an antenna mounted upon it. Each Figure shows
three curves for different values of the parameter $\eta$, which determines the manner in which distance affects the available capacity. We simulate here over values $-2, -3, -4$, represented by the bottom, middle and top lines, respectively. For each value of $\eta$, the possible gain for $3 \leq n < \infty$ antennas is bounded between the two related curves.

These results demonstrate how, as the train extends its stay in the cell due to large $z/d$ ratio, the relative capacity gain increases. This is due to the fact that MASS not only increases overall available capacity, but specifically does so by dramatically increasing the available capacity around the handoff slots, where capacity is at its lowest point. As the $z/d$ ratio grows, the so does the ratio between the highest and lowest available capacity in the cell, which makes the possible gain by adding antennas increasingly higher.

![Figure 14: Two antennas](image1)

![Figure 15: Infinite antennas](image2)
6.2 B&B and B&B+MASS performance

In order to evaluate the performance of B&B, we compare it to the performance of a family of "channel-unaware" algorithms, dedicated as well to ensuring transmission of packets without exceeding some predefined delay per packet. Each algorithm in this family monitors the number of packets in the buffer at all times, and transmits when there is some packet which has exceeded $\sigma \cdot w_{\text{max}}$ delay, for some algorithm-specific constant $0 < \sigma < 1$. High $\sigma$ values take advantage of the delay constraints but run the risk of violating them as well, while low values of $\sigma$ tend to be wasteful in the number of slots allocated by them. In the cases we simulated, the optimal algorithm in this family which did not violate the delay constraint used $\sigma = 0.7$ (70%). We present here the results of comparing the performance of this algorithm with that of B&B.

Figure 16 (dotted line) demonstrates the relative sizes of slot allocations from B&B and the selected competing, channel-unaware algorithm. As can be seen here, the quasi-optimal policy of B&B cuts the slot requirements of the train dramatically, sometimes to one third the number required by the competing algorithm. Specifically, as the delay constraints are relaxed, we can see a great decrease in required slots. This is due to the fact that longer delays allow B&B more freedom in selecting the slots at which to transmit, while the competing algorithms are blind to the capacity fluctuations and thus are hardly affected by such changes.

Moving on to the combination of MASS and B&B, we simulate and compare the performance increase achieved by using 2 antennas instead of one. The relative slot allocation size between these two is shown in Figure 16 (solid line). Here we notice the opposite trend than the one seen on the dotted line, and as the delay constraints relax so does the advantage of using multiple antennas lessen. The cause of this trend is the same as before, however, since an increase in antennas is less handy when the delay constraint is such that allows great flexibility in slot allocation.

The results presented here were simulated over symmetric networks. Additional testing has been conducted over more realistic scenarios, where the distances of the train from the base-stations vary. These showed that the performance increase displayed here for the
Figure 16: Relative Gains of MASS+B&B and B&B

symmetric scenario holds also for these scenarios.

References


Lemma 17 Assume that in the uniform deployment, B&B transmits in segments other than $\gamma$. Then for every arbitrary deployment $D$ there is a deployment $D'$ such that for every antenna $i$ $\gamma^i = M_{\text{max}}^i$ and the allocation for $D'$ exceeds that for $D$ by at most $n$ slots.

Sketch of proof: Eq. 14 does not apply for some antenna when the interval between $\text{cov}^i$ and $\text{cov}^{i+1}$ is too short, which is caused by increased proximity of one or both neighboring antennas. This state can therefore be changed by spacing out the antennas. Since the uniform deployment requires transmission during segments other than $\gamma$, there exists some antenna $a_i$ in deployment $D$ which can be moved towards one neighbor to increase the size of its segment $\gamma^i$.

Such a move will change the allocation so that more packets are transmitted during the strongest slots in the cycle than previously, which can only improve the allocation size. What remains to be verified is that the rest of the allocation does not worsen as a result. This can be shown by noting that in each segment, starting from the strongest slots and moving down, the same allocation used in deployment $D$ can be used also after the antenna moved (under an index alignment). Thus, the only cost of such a move may stem from the fact that one slot in segment $\gamma^i$ may not be fully utilized. ■

Sketch for proof of Lemma 16: In the uniform deployment, both the capacity curves and the allocations in both capacity valleys are identical, after antenna-based index alignment. We therefore know that for every $i = 1, 2, 3$, the allocation for $P_i$ is the same as the allocation for $P_{i+3} \cup P_{i+6}$, after index alignment (For the case of $i = 2$, see figure 17). In order to prove the lemma it is enough to prove that for every $i = 1, 2, 3$ $P_{i+6}$ are transmitted by slots of capacity no greater than the capacity of any slot allocated for transmitting $P_i$. 55
We prove this by reviewing each value of $i$. In segment $\alpha$ ($i = 1$), we’ve seen that the same top slots are allocated in both deployments, so $P'_7$ is transmitted by the remaining, weaker slots.

In segment $\beta$ ($i = 2$) we proved the allocations for $P_2$ and $P'_5$ are interleaved, so the slots allocated for the remaining packets contain slots no stronger than weakest slot used for transmission of $P_2$ or $P'_5$.

Finally, in segment $\delta$ ($i = 3$) we note that the internal policy shift in $P_3$ is no later than that in transmitting $P'_6$, since the segment is shorter. Therefore, the allocation for $P_3$ never uses slots weaker than the weakest slot allocated for $P'_6$, which once again ensures the lemma to be true for all slots allocated in $P'_9$. ■

Figure 17: Deployment Comparison: Segment $\beta$