Designing the Master Schedule for Demand-Adaptive Transit Systems

Teodor Gabriel Crainic
Département management et technologie
École des Sciences de la Gestion
and
CIRRELT
Université du Québec à Montréal
Canada
theo@crt.umontreal.ca

Fausto Errico
DEI, Politecnico di Milano
Italy
and
CIRRELT
Université du Québec à Montréal
Canada
fausto.errico@polimi.it

Federico Malucelli
DEI, Politecnico di Milano
Italy
malucell@elet.polimi.it

Maddalena Nonato
Facoltà di Ingegneria, Università di Ferrara
Italy
nntmdl@unife.it

June 30, 2008

Abstract

Demand-Adaptive Systems (DAS) display features of both traditional fixed-line bus services and purely on-demand systems such as dial-a-ride, that is they offer demand-responsive services within the framework of traditional scheduled bus transportation. A DAS bus line serves, on one hand, a given set of compulsory stops according to a predefined schedule specifying the time windows associated with each, providing the traditional use of the transit line, without requiring any reservation. On the other hand, passengers may also issue requests for transportation between two desired, optional, stops, which induces detours in the vehicle routes. The design of a DAS line is a complex planning operation that requires to determine not only its design in terms of selecting the compulsory stops, but also its master schedule in terms of the time windows associated with the compulsory stops. Designing a DAS thus combines elements of strategic and tactical planning. In this paper we focus on determining a master-schedule for a single DAS line. We propose a mathematical description and a solution framework based on the estimation of a number of statistical parameters of the demand and the DAS line service. Results of numerical experiments are also given and analyzed.

Keywords: Public transit, demand-responsive systems, demand-adaptive systems, scheduling

1 Introduction

Traditional transit services are particularly suited to handle situations where the demand for transportation is *strong*, i.e., when there is a consistently high demand over the territory and for the time period considered. The high degree of resource sharing by a large number of passengers makes it then possible to provide efficiently and economically high quality, i.e., frequent, services operating generally high-capacity vehicles over fixed routes and schedules. Routes and schedules may and do vary during the day, but, in almost all cases, they are not dynamically adjusted to the fluctuations of demand. In contrast, when the demand for transportation is *weak*, e.g., during out of rush-hour periods or in low-population density zones, operating a good-quality traditional transit system is very costly. In particular the fixed structure of traditional transit services cannot economically and adequately respond to significant variations in demand.

Demand-responsive systems are a family of mass transportation services which, as their name suggests, are *responsive* to the actual demand for transportation in a specific time period. Such services evolve toward a personalization of mass services: itineraries, schedules, and stop locations are variable and determined according to the needs for transportation as they change in time. Demand-responsive systems were introduced under the name of *Dial-a-Ride* (*DAR*) as door-to-door services for users with particular needs or reduced mobility, such as handicapped and elderly people (Ioachim *et al.* 1995 [10], Toth *et al.* 1996 [14]) The *flexibility* of DAR systems to respond to varying individual requests for transportation provides the means to offer more personalized services, while still maintaining a certain degree of resource sharing. This has lead certain transportation or city authorities to extend DAR services to more general transportation settings.

DAR systems display, however, a number of drawbacks, some of which follow from the extreme flexibility inherent in the system definition. Thus, for example, because the supply of transportation service changes according to needs expressed for particular time periods, neither the transit operator nor the users may predict the vehicle itineraries, stop locations, and associated schedules. As a consequence, users are obliged to book the service well in advance of the actual desired time of utilization and the actual pick up time is very much left to the discretion of the operator. For similar reasons, it is extremely difficult to integrate DAR and other traditional transit services.

A new type of demand-responsive systems, denoted *Demand-Adaptive System* (*DAS*) has been introduced to address some of these issues (Malucelli *et al.* 1999 [12], Quadrifoglio *et al.* 2007 [13]). DASs are transit services displaying features of both traditional fixed-line bus service and purely on-demand systems such as DAR. In other words, a DAS attempts to offer demand-responsive services within the framework of traditional scheduled bus transportation. The relevance of this kind of hybrid services in general public transit is also underlined in [8] (Hickman *et al.* 2000) and in [9] (Horn 2002).

A DAS bus line serves, on one hand, a given a set of *compulsory* stops according to a predefined schedule specifying the time windows associated with each, providing the traditional use of the transit line, without in-advance reservations. On the other hand, similarly to DAR services, passengers may issue requests for transportation between two desired, *optional* stops (not necessarily on the same line), which induces detours in the vehicle routes.

Similarly to most transportation systems dedicated to serve several demands with the same vehicle, traditional transit systems involve a complex planning system made up of many interrelated decisions. Schematically, the design of the system in terms of line routes is determined during the so-called strategic planning phase, timetables and vehicle schedules and routes are part of the tactical planning phase, and crew schedules are built during operational planning (Ceder *et al.* 1997 [2]). Comparatively, purely on-demand services such as DAR, need little strategic design, mainly to define service areas and the composition of the fleet (e.g., number and type of vehicles). The most important planning process for DAR is at the operational level when routes and schedules are determined little time before actual operations and are possibly dynamically modified one service has begun.

DAS services combining characteristics of traditional and on-demand systems require both a system-design phase and an operational, time and user request-dependent adjustment of vehicles routes and schedules. The latter has been addressed in [11] (Malucelli *et al.* 2001) and in [6] (Crainic *et al.* 2005). The former forms the topic of this paper. It is, in a certain sense, a more complex planning operation than for traditional transit because it requires not only to determine the design of the line as the selection of the compulsory and optional stops, but also the determination of the time windows associated with the compulsory stops. Designing a DAS thus combines elements of strategic and tactical planning. To emphasize this characteristics, we identify the process as the *master-route* network and the *master-schedule*, respectively.

In this paper we focus on determining a master-schedule for a single DAS line. We propose a mathematical description and a solution framework based on the estimation of a number of statistical parameters of the demand and the DAS line service. A sampling approach is used for the estimations.

The remaining part of the paper is organized as follows. We give a brief description of DAS services in Section 2, while commonalities and differences among scheduling DAS, DAR, and traditional transit services are discussed in Section 3. Section 4 is dedicated to the description of the DAS line master-scheduling problem and the solution framework we propose. We discuss the effectiveness of the method and report computational results in Section 5. We conclude in Section 6.

2 Demand Adaptive Systems

Demand adaptive systems were first introduced in [12] (Malucelli *et al.* 1999) and then treated in a more general context in [6] (Crainic *et al.* 2005) (see also Crainic *et al.* 2000, 2001 [4, 5] and Malucelli *et al.* 2001 [11]). A similar type of service is also described in [13] (Quadrifoglio *et al.* 2007).

A DAS targets low-density/volume demand areas and attempts to conjugate the advantages of traditional transit transportation services and the flexibility of on-demand personalized services. It is based on the observation that even in such areas there are locations where a relatively important part of the overall demand may be consistently found: railway and underground stations, shopping centers, hospitals, etc. This leads to the possibility to economically design a backbone transit service covering these most attractive stops, while allowing vehicles to detour as needed to pick up and drop off passengers at the other stops. The latter capability, combined to an on-request booking system, increases customer satisfaction and the dimension of the potential user group.

In its most general form, a DAS is made up of several lines and is connected to the lines of the traditional transit system. Several vehicles operate on each DAS line providing service among a sequence of *compulsory* stops. Each compulsory stop is served within a predefined *time window*. The collection of time windows corresponding to the compulsory stops, including the start and end of the line, makes up the *master schedule* of the DAS line. This makes up the traditional part of a DAS. Additional service and flexibility is provided by allowing customers to request service from and to *optional* stops, that is, stops which are served only if a request is issued and it is accepted. We identify users who request service at an optional stop as *active*, while users moving only between compulsory stops are identified as *passive*.

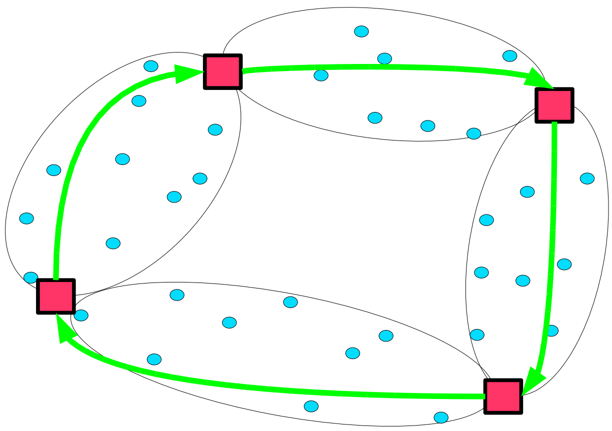

Figure 1: A Basic DAS Line Serving the Compulsory Stops

To serve optional stops, the vehicle must generally deviate from the shortest path joining two successive compulsory stops. The set of optional stops that it is possible to visit between

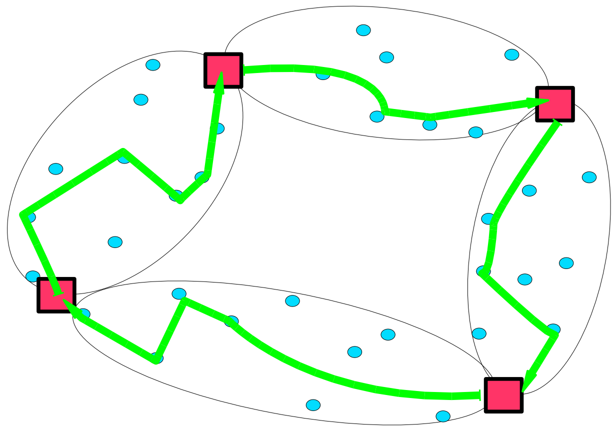

Figure 2: The DAS Line Serving Optional Stops

two consecutive compulsory stops is part of the design of the DAS line and is denoted *segment*. An optional stop cannot belong to more than one segment. Figure 1 depicts the basic DAS service of the compulsory stops, while Figure 2 illustrates the same DAS line when user requests for optional stops are present.

Transfers between DAS lines and between these and regular transit lines take place at compulsory stops. Time windows play an important role in this context because they establish time relations among different DAS and traditional lines which share the same compulsory stops. The time windows in the master schedule also influence the flexibility the service may provide for user requests at optional stops. The wider they are, the more flexibility there is. Yet, it is not possible to increase their width arbitrarily, because the service would slow down excessively, loosing attractiveness. Notice finally, that the time windows and the segment specification provide an *a priori* guarantee relative to the longest time users might have to spend traveling on the line. In any case, the detours associated with optional stops must be consistent with the time windows at compulsory stops.

A DAS service is currently operated in Los Angeles County (Quadrifoglio *et al.* 2007 [13]) as a nighttime service. The system is called by the authors MAST and slightly differs from DAS because, instead of considering a set of optional stops, it defines a *service area* and allows service among any point in such an area. This is possible because the road network is very regular and the line covers a quasi rectangular area with a single vehicle traveling back and forth within this rectangular. The service could be assimilated to a circular DAS line with 5 segments. Another implementation of a DAS service is the one we are currently studying in the city of Brescia, in northern Italy. The line links some of the suburbs, mainly mountainous regions, to the center of the city. The DAS line which we are planning consists of 6 segments and a total of 53 optional stops. At the moment we are studying several possible designs and we are simulating the operations. A discussion on such a topic goes beyond the purpose of this paper and it is the subject of an ongoing work.

3 Scheduling Issues in DAS, DAR and Traditional Transit Systems

Scheduling is a fundamental planning activity for any transportation system, in particular for DASs, DAR and traditional transit systems. The nature of the scheduling process changes significantly, however, according to the type of service at hand.

In traditional transit systems, a schedule indicates the passing times at each stop of each line. Vehicles are supposed to follow these times as strictly as possible, since users of the system base their trip plans on the published schedules. Scheduling problems in traditional transit system belong to the so-called tactical planning level, the line definitions and the service frequencies being usually assumed known. Once the schedule has been established, it remains unchanged for medium-term periods, such as six months or one year. For a more in-depth discussion of scheduling issues in traditional transit systems, the interested readers are referred to [2] (Ceder *et al.* 1997).

The situation is different for DAR systems. Schedules are still indicating vehicle itineraries, stops, and passing times, but these are particular to each vehicle tour according to the actual requests for transportation accepted for the corresponding time period. This corresponds to an operational planning level activity that decides on all schedule components (i.e., line itinerary, stops, and passing times), which are valid only for the duration of the specific service. See [3] (Cordeau *et al.* 2003) for a review of the topic.

The case of DASs is more complex. Because DAS aims to provide demand-responsive services within the framework of traditional scheduled transit transportation, its scheduling combines the two planning processes briefly sketched above. Two schedules are thus built. A master schedule defines the partial line (vehicle) itineraries, the sequence of compulsory stops, and time windows at these stops. This schedule plays the same role for the transit authority and the passive users of the system as the schedule in traditional transit systems. At operation time, the actual schedule is built to incorporate the additional, optional stops corresponding to the accepted active-user requests, while respecting the time windows constraints imposed by the master schedule. The problem of finding a DAS schedule at operational level was addressed in [11] (Malucelli *et al.* 2001), and in [6] (Crainic *et al.* 2005).

Building the master schedule is a tactical planning level activity, where actual service times at compulsory stops are modified according to the season. It is also an important component of the strategic planning process, as the definition of the segments making up the line requires the specification of time windows at compulsory stops. The next section is dedicated to these issues for a single DAS line.

4 The DAS Line Master Scheduling Problem

This section is dedicated to the issue of determining the master schedule of a single DAS line, that is, determining the time windows for the compulsory stops of the line. This so-called *DAS Line Master Scheduling Problem* (*DLMSP*) may be viewed as the last stage of the DAS line design problem which is addressed in more depth in [7] (Errico 2008).

The single-line DAS design problem assumes that the territory to be covered by the DAS line has been determined, the travel times between any pair of potential stops in the territory (these include transfer points to other lines or transportation systems) have been accurately estimated, and that a measure of the transportation demand among the potential stops is available. For a given time horizon where demand is assumed stable (e.g., morning rush hour), the DAS line design problem is made up of several interrelated decisions regarding the selection of compulsory stops among all the potential stops in the territory, their sequencing, the partitioning of the optional stops into segments, and the determination of the master schedule, that is the definition of the time windows vehicles will have to respect at compulsory stops. The first three components make up the so-called topological-design phase of the problem and a number of methodological approaches are proposed in [7] (Errico 2008) to address various problem settings, e.g., objectives to be satisfied (vehicle cost, travel time experienced by users, a combination thereof, etc. - we use a combination), whether all potential stops should be reachable by the designed line, and so on. The last component of the design process constitutes the object of this paper.

A formal model for DLMSP is presented in Section 4.1. In Section 4.2 we focus on the *Single Segment Master Scheduling Problem*, a core subproblem in addressing the DLMSP. Section 4.3 presents the complete solution approach we propose for the DLMSP.

4.1 Problem description and modeling

The problem of building the DAS line master schedule, that is, to fix the time windows at all compulsory stops of the line, assumes two inputs. The first consists in the topological design of the line: the ordered set of compulsory stops and the associated set of optional stops partitioned into segments. The demand for transportation between the stops of the line makes up the second input.

The choice of time windows must be performed taking into account several conflicting goals. First, the master schedule should provide sufficient time between compulsory stops such that, during actual operations, vehicles may serve all requests for service at optional stops. Second, for economical reasons, the total maximum time of the line should be as short as possible. Finally, quality of service criteria also induce conflicting actions: while users already on the bus prefer narrow time windows, to avoid long delays at compulsory stops, and short travel times between consecutive compulsory stops, to avoid being on the bus for long, users at optional stops prefer longer travel times that allow vehicles to detour by their stop.

To illustrate the incompatibility of these goals, consider that simultaneously enforcing small time windows and high probability of being able to serve all potential requests implies a rather long travel time for each segment compared to the corresponding shortest path. But, since the actual number of requests is usually small compared to the total potential number, such a strategy would result in vehicles arriving at compulsory stops well before the earliest departure time, significant dead times at compulsory stops for users, and long ride times for the line. To avoid this, time windows have to be smaller, resulting in a smaller probability of being able to serve the whole set of issued requests. This is indeed general DAS operational policy (Malucelli *et al.* 1999 [12] and Crainic *et al.* 2005 [6]), requests that cannot be accommodated being either lost, served by a later vehicle or by taxi, according to the policy of the transit company.

We thus assume a maximum width for time windows at compulsory stops and aim to select the time displacement among them which minimize the maximum vehicle ride time while guaranteeing to serve the set of requests with a given probability. The maximum width and the service probability are, of course, managerial decisions and thus application dependent.

Demand for transportation is usually described as the number of potential trips that might be requested during the time period considered for each pair of stops. Based on this information, it is straightforward to compute the probability of at least a request being issued for a given pair of stops, as well as the probability of each optional stop of being requested for service either to board or unboard a vehicle. We work with this last set of probabilities in the model we propose. Consequently, the goal of serving the *whole set of requests* with a given probability becomes serving the *whole set of requested stops* with a given probability. This makes it easier to address the problem.

To formally write the model, consider a sequence of compulsory stops $H = \{f_0, f_1, \ldots, f_n\}$. Sets of optional stops F_h, with $h = 1, \ldots, n$ are associated with each pair of consecutive compulsory stops $\langle f_{h-1}, f_h \rangle$. The sets F_h are mutually disjoint. We define a directed graph $G_h = (N_h, A_h)$ for any pair $\langle f_{h-1}, f_h \rangle$ such that $N_h = F_h \cup \{f_{h-1}, f_h\}$ and $A_h = N_h \times N_h$. We call G_h a *segment* and $G = \cup G_h$. A traversing (travel) time c_{ij} is associated with each arc $(i, j) \in A$. A probability p_i of being requested for service is associated with each optional stop $s_i \in \cup_h F_h$.

The DLMSP consists in associating to each compulsory stop f_h a time window $[a_h, b_h]$ such that the vehicle must not leave the compulsory stop f_h before a_h, nor after b_h; it is allowed, however, to arrive to f_h before a_h. Our goal is to find the best sequence $\{\langle a_1, b_1 \rangle, \langle a_2, b_2 \rangle, \ldots, \langle a_n, b_n \rangle\}$ which, with a given probability P_ϵ, guarantees service for *every* optional stop which can be requested for service. We define the best sequence as the one displaying the smallest value of b_n.

For the sake of simplicity of exposition, we consider the case where all time windows are of equal width $b_h - a_h = \delta$, which reduces the problem to that of finding the best sequence $\{b_0, b_1, \ldots, b_n\}$. The procedure we propose extends straightforwardly, however, to the cases when 1) the time windows are fixed but different for the compulsory stops, and 2) time windows are bounded by a maximum width value but are allowed to be smaller. The latter case could also consider finding, for a given compulsory stop f_h, the maximum a_h which guarantees with a given probability no vehicle dead time at f_h.

We now focus on a core subproblem that estimates the travel time, and thus the time window at the destination compulsory stop, of a single segment. The full algorithm will then bring together the sequence of segments making up the route.

4.2 The Single Segment Subproblem

How long does it take to travel a segment? The answer obviously depends on what optional stops have to be serviced and this is usually different each time a vehicle travels the line and the segment, because service at optional stops follows particular user requests that have been accepted. Consequently, such operational information must be estimated for the tactical-planning purpose of building a master schedule. We propose a statistical estimation of the travel time of a given segment based on an efficient sampling procedure.

Consider the generic segment $G_h = (N_h, A_h)$, and recall that $N_h = F_h \cup \{f_{h-1}, f_h\}$, f_{h-1} and f_h are the initial and terminal compulsory stops of the segment, respectively, F_h is a set of optional stops, and $A_h = N_h \times N_h$. Let $\bar{L}_{h-1}$ represent the departure time from compulsory stop f_{h-1}.

As indicated previously, each optional stop $s_i \in F_h$ has a positive known probability p_i of being *active*, that is, of being requested for service during a particular vehicle run. We assume these probabilities to be mutually independent. The set of optional stops that are simultaneously active during a vehicle run, $S_h \subseteq N_h$, is denoted the *active set* (with every $s_i \in S_h$ being active, while every $s_i \in F_h \setminus S_h$ not being active). The probability of any set S_h to be active, p_{S_h}, is positive and may be easily derived from the probabilities p_i of its active optional stops.

The time required by a vehicle to travel from the initial to the terminal compulsory stop of segment h serving a **given** set S_h of active optional stops in S_h is denoted H_h and is called the *service time* of set S_h. Assuming an efficient operation of the line, the service time H_h may be approximated at planning level as the duration of the shortest path starting in f_{h-1}, ending in f_h, and passing by all the stops of the active set S_h. We thus used a Minimum Hamiltonian Path solver to compute H_h for our experimentations. (More sophisticate procedures may be implemented to take advantage of particular application attributes, but this does not change the general behavior of the proposed methodology.)

The service time associated with segment h is of course a random variable at planning level and we denote it $H_h(\omega)$. The goal of the *Single Segment Meta Schedule Problem* (*SSMST*) is to determine the lowest value b_h which guarantees, with probability $1-\epsilon$, that the vehicle has sufficient time to serve an active set. That is, fix $b_h = H_h^{1-\epsilon} + \bar{L}_{h-1}$, where $H_h^{1-\epsilon}$ is such that $\mathcal{P}\{H_h(w) \leq H_h^{1-\epsilon}\} \geq 1-\epsilon$.

The computation of $H_h^{1-\epsilon}$ requires the knowledge of the Cumulative Distribution Function (CDF) and thus of the Probability Mass Function (PMF) of the random variable $H_h(\omega)$. Since the latter requires $2^{|N_h|}$ Minimum Hamiltonian Path computations, this approach is not computationally affordable in most cases. Consequently, we estimate the PMF and sampling appears as the method of choice.

It is difficult to estimate how large a sample that represents adequately the population of active sets should be, but we suppose it could become quite large. Then, for computing efficiency reasons, we propose instead the very simple following algorithm:

- Take a number $\{r_1, r_2, \ldots, r_l\}$ of random samples of relatively small cardinality;
- For each sample r_k, compute its PMF_k and CDF_k, as well as the value of b_h^k;
- Compute the mean value and standard deviation of b_h^k; If the standard deviation is close to the mean value, that is if the solution is *precise*, stop;
- Otherwise, increment the cardinality of the samples and iterate the previous steps.

The undeniable advantage of this algorithm is its simplicity. On the other side, one cannot guarantee an unbiased solution, nor that the dimension of the samples will stay within computationally reasonable dimensions. A number of parameters (e.g., the number of samples) must also be calibrated. Yet, as illustrated by the results of Section 5, the method is very effective and adverse effects are not noticeable.

4.3 Solution Approach to DLMSP

We now present the complete solution method for the DAS line master schedule problem, where we need to sew segments together.

In the previous subsection, we decoupled segments by assuming that the vehicle departure time from its initial compulsory stop f_{h-1} was known for each segment G_h. Actually, this is true for the first segment only, the departure time from the first compulsory stop f_0 being here arbitrarily denoted $t = 0$, which also translates into $\mathcal{P}\{L_0(\omega) = 0\} = 1$. For all subsequent segments, the departure time depends upon the arrival time of the previous segment, which depends in turn on its departure and service times and so on. Since service times are random variables, segment departure times for all segments but the first are also random variables.

We therefore introduce the random variable $L_h(\omega)$ representing the vehicle departure time from compulsory stop f_h, where $a_h \le L_h(\omega) \le b_h$. We introduce also the random variable $T_h(\omega)$:

$$T_h(\omega) = L_{h-1}(\omega) + H_h(\omega) \tag{1}$$

not constrained to belong to $[a_h, b_h]$. The vehicle departure times at two consecutive compulsory stops and the service time for the segment to which they belong to are then related as follows:

$$L_h(\omega) = \begin{cases} T_h(\omega) & \text{if } \omega \mid a_h \le T_h(\omega) \le b_h; \\ a_h & \text{if } \omega \mid T_h(\omega) < a_h; \\ b_h & \text{if } \omega \mid T_h(\omega) > b_h. \end{cases} \tag{2}$$

We assume in the following, without loss of generality but with a simplified notation, that as long as a vehicle arrives at a compulsory stop f_h during the interval $[a_h, b_h]$, the arrival and departure times coincide.

Recall that the value b_h for segment G_h has to be such that it is possible to serve all possible active sets with a given probability. We must therefore compute the PMF (and consequently the CDF) of $T_h(\omega)$, that is, select $b_h = T_h^{1-\epsilon}$, where $T_h^{1-\epsilon}$ is such that $\mathcal{P}\{T_h(\omega) \le T_h^{1-\epsilon}\} > 1 - \epsilon$.

Notice that, by hypothesis, $H_h(\omega)$ and $L_{h-1}(\omega)$ are independent. Consequently, the PMF of their summation, $T_h(\omega)$, can be computed through the simple convolution of the PMFs of $H_h(\omega)$ and $L_{h-1}(\omega)$. The problem of finding b_h then reduces to the problem of estimating the PMF of $H_h(\omega)$ and $L_{h-1}(\omega)$. We showed in the previous subsection how to compute the CDF of $H_h(\omega)$. The CDF of $L_{h-1}(\omega)$ may be easily obtained from that of $T_{h-1}(\omega)$.

We can now present the scheme of the algorithm we propose for the DLMSP. The algorithm accepts as input the sequence of segments $G = \cup_{1,2,\ldots,n} G_h$ and the service probability $\mathcal{P}_\epsilon = (1 - \epsilon)^n$, and proceeds as follows:

1. For every segment $G_h,\ h \in \{1, 2, \ldots, n\}$
   (a) Compute PMF and CDF of $L_h(\omega)$
   (b) Compute PMF and CDF of $H_h(\omega)$
   (c) Compute PMF and CDF of $T_h(\omega)$ as the convolution of the PMFs of L_{h-1} and H_h
   (d) Compute $T_h^{1-\epsilon}$ and set $b_h = T_h^{1-\epsilon}$.

The algorithm returns the best sequence $\{b_1, b_2, \ldots, b_n\}$ of latest departure times for the segments, such that any randomly requested optional stop is served with probability $\mathcal{P}_\epsilon =$

$(1-\epsilon)^n$. Notice that the expression for $\mathcal{P}_\epsilon$ is linked to expression of service probability for a single segment $1-\epsilon$ by the hypothesis that single segment service times are mutually independent.

5 Results

This section is dedicated to a discussion of computational results relative to the estimation of the PMF of random variables $H_h(w)$ when considering decoupled segments, as well as of the corresponding values $H_h^{1-\epsilon}$. This is in fact the core point of the solution methodology proposed. Experimental results support, in particular, the claim that precise and unbiased values of $H_h^{1-\epsilon}$ are obtained even for relatively small sample dimensions.

Name	Longest 1-Path	Hamiltonian Path	Sum Prob	$> 50\%$
A20	1924.8	4113	9.03	9
B20	1925.43	4032	9.31	9
C20	1634.38	3695	9.94	8
A30	1924.8	4612	14.75	13
B30	1925.43	4464	14.84	13
C30	1667.65	4356	13.26	15
A40	1924.8	4972	22.5	14
B40	1925.43	5353	18.83	16
C40	1907.8	4947	22.64	14
A50	1924.8	5748	27.39	21
B50	1925.43	6462	25.13	20
C50	1907.8	5422	24.65	25

Table 1: Features of Test Problem Instances

We tested our algorithm over square-shaped segments with initial and terminal compulsory stops located at the extremities of one of the diagonals. Optional stops are uniformly distributed on the square. Distances between optional stops are Euclidean and traveling times are proportional to distances with proportionality constant 1. We generated instances with a number of stops, including the two compulsory ones, varying from 20 to 50 by steps of 10. With each optional stop is associated a probability included in the open interval $(0,1)$. For each problem dimension we randomly generated three instances, different both in the possible locations of optional stops and the probabilities associated with them.

Table 1 displays the main features of the test problem instances: name; time length of the longest path with only one optional stop and time length of the Hamiltonian path taken over all optional stops which represent respectively a lower and upper bound on the value of the latest departure time at the second compulsory stop; the sum of the probabilities associated with optional stops, and the number of optional stop with probability grater than 50%. Hamiltonian paths are computed with a modified version of the Asymmetric TSP code available in [1] (Carpaneto *et al.* 1995).

Name&Dim.	nSamples	dimSamples	1-*epsilon*	$H^{1-\epsilon}$	Dev-Strd	Time (sec.)
A20	10	50	0.95	3470.21	68.0074	3.46
A20	10	100	0.95	3472.21	60.9672	6.92
A20	10	150	0.95	3508.21	51.4296	10.38
A20	10	200	0.95	3504.21	39.2556	13.89
A20	10	250	0.95	3496.21	38.4318	17.34
A20	20	50	0.95	3476.21	81.2219	6.93
A20	20	100	0.95	3481.21	71.3583	13.87
A20	20	150	0.95	3497.21	55.9553	20.81
A20	20	200	0.95	3501.21	45.7384	27.72
A20	20	250	0.95	3491.21	36.4829	34.68
A20	30	50	0.95	3492.88	98.818	11.32
A20	30	100	0.95	3478.21	67.6535	22.62
A20	30	150	0.95	3490.88	56.1783	33.86
A20	30	200	0.95	3500.21	42.0714	45.21
A20	30	250	0.95	3492.88	34.3074	56.65
B20	10	50	0.95	3836.21	48.5283	3.63
B20	10	100	0.95	3840.21	38	7.22
B20	10	150	0.95	3822.21	31.241	10.81
B20	10	200	0.95	3830.21	30.7246	14.45
B20	10	250	0.95	3820.21	32.2955	18.03
B20	20	50	0.95	3827.21	48.2701	7.24
B20	20	100	0.95	3837.21	41.1461	14.45
B20	20	150	0.95	3822.21	31.8904	21.59
B20	20	200	0.95	3836.21	34.6121	28.82
B20	20	250	0.95	3830.21	36.1386	36.01
B20	30	50	0.95	3824.88	49.7896	11.78
B20	30	100	0.95	3836.88	43.2088	23.53
B20	30	150	0.95	3833.55	36.4417	35.42
B20	30	200	0.95	3836.88	37.6298	47.15
B20	30	250	0.95	3824.88	32.6803	58.82
C20	10	50	0.95	3124.21	42.2019	3.49
C20	10	100	0.95	3128.21	33.5708	6.99
C20	10	150	0.95	3112.21	10.8628	10.46
C20	10	200	0.95	3114.21	9.05539	13.94
C20	10	250	0.95	3116.21	10.8628	17.4
C20	20	50	0.95	3116.21	48.9183	6.99
C20	20	100	0.95	3129.21	34.5254	13.94
C20	20	150	0.95	3119.21	18.735	20.97
C20	20	200	0.95	3117.21	15.7162	27.93
C20	20	250	0.95	3112.21	13.8924	34.99
C20	30	50	0.95	3112.88	48.7237	11.46
C20	30	100	0.95	3134.88	32.4962	22.87
C20	30	150	0.95	3116.88	18.4662	34.26
C20	30	200	0.95	3114.21	16.3401	45.79
C20	30	250	0.95	3115.55	15.4596	57.28

Table 2: Results for Segments with 20 Nodes

Name&Dim.	nSamples	dimSamples	1-*epsilon*	$H^{1-\epsilon}$	Dev-Strd	Time (sec.)
A30	10	50	0.95	3926.21	86.3423	4.11
A30	10	100	0.95	3924.21	58.8303	8.27
A30	10	150	0.95	3932.21	53.2635	12.37
A30	10	200	0.95	3936.21	42.3674	16.49
A30	10	250	0.95	3950.21	31.9844	20.6
A30	20	50	0.95	3938.21	76.2889	8.26
A30	20	100	0.95	3926.21	63.6946	16.45
A30	20	150	0.95	3938.21	49.6286	24.74
A30	20	200	0.95	3934.21	36.8511	32.92
A30	20	250	0.95	3937.21	31.1609	41.17
A30	30	50	0.95	3933.55	81.4187	12.4
A30	30	100	0.95	3926.21	57.7062	24.75
A30	30	150	0.95	3930.88	47.1381	37.09
A30	30	200	0.95	3929.55	40.6448	49.31
A30	30	250	0.95	3936.88	36.1525	61.61
B30	10	50	0.95	4062.21	50.7642	5.54
B30	10	100	0.95	4088.21	38.0132	11.13
B30	10	150	0.95	4096.21	38.4187	16.7
B30	10	200	0.95	4088.21	32.2645	22
B30	10	250	0.95	4102.21	25.5734	27.6
B30	20	50	0.95	4075.21	52.7257	11.15
B30	20	100	0.95	4102.21	41.7133	21.99
B30	20	150	0.95	4104.21	34.9571	33.37
B30	20	150	0.95	4104.21	34.9571	33.37
B30	20	200	0.95	4089.21	40.4475	44.03
B30	20	250	0.95	4095.21	32.5883	54.95
B30	30	50	0.95	4086.21	59.0931	16.73
B30	30	100	0.95	4109.55	43.8634	33.32
B30	30	150	0.95	4096.88	37.2827	49.67
B30	30	200	0.95	4084.21	40.3733	65.4
B30	30	250	0.95	4098.21	28.7054	81.89
C30	10	50	0.95	3362.21	20.4695	3.8
C30	10	100	0.95	3390.21	33.2716	7.51
C30	10	150	0.95	3372.21	27.55	11.22
C30	10	200	0.95	3358.21	28.0535	14.99
C30	10	250	0.95	3366.21	24.0208	18.7
C30	20	50	0.95	3375.21	41.1825	7.56
C30	20	100	0.95	3370.21	32.573	14.98
C30	20	150	0.95	3367.21	29.7825	22.41
C30	20	200	0.95	3356.21	26.7208	29.8
C30	20	250	0.95	3362.21	23.9374	37.22
C30	30	50	0.95	3368.21	42.0357	11.26
C30	30	100	0.95	3373.55	32.1092	22.37
C30	30	150	0.95	3363.55	32.5883	33.54
C30	30	200	0.95	3354.21	27.8747	44.71
C30	30	250	0.95	3363.55	22.4277	55.78

Table 3: Results for Segments with 30 Nodes

Name&Dim.	nSamples	dimSamples	1-*epsilon*	$H^{1-\epsilon}$	Dev-Strd	Time (sec.)
A40	10	50	0.95	4552.21	52.4595	7.46
A40	10	100	0.95	4568.21	30.9516	14.67
A40	10	150	0.95	4562.21	33.6452	21.88
A40	10	200	0.95	4564.21	22.2711	29.4
A40	10	250	0.95	4562.21	22.1811	36.67
A40	20	50	0.95	4554.21	46.9042	14.68
A40	20	100	0.95	4567.21	29.189	29.4
A40	20	150	0.95	4557.21	27.0555	44.17
A40	20	200	0.95	4564.21	24.0832	58.71
A40	20	250	0.95	4560.21	20.1494	73.81
A40	30	50	0.95	4557.55	51.9808	21.86
A40	30	100	0.95	4564.21	29.2575	44.16
A40	30	150	0.95	4558.88	27.7849	66.37
A40	30	200	0.95	4562.21	22.4944	88.1
A40	30	250	0.95	4557.55	20.9284	109.74
B40	10	50	0.95	4362.21	23.9165	34.8
B40	10	100	0.95	4366.21	23.9165	64.79
B40	10	150	0.95	4378.21	11.8322	98.96
B40	10	200	0.95	4376.21	24.3311	131.69
B40	10	250	0.95	4382.21	20.2485	166.35
B40	20	50	0.95	4373.21	34.322	64.82
B40	20	100	0.95	4369.21	31.5595	131.72
B40	20	150	0.95	4382.21	21.4009	197.53
B40	20	200	0.95	4373.21	20.4939	261.99
B40	20	250	0.95	4375.21	24.0832	322.11
B40	30	50	0.95	4375.55	38.3406	98.98
B40	30	100	0.95	4370.21	31.3369	197.57
B40	30	150	0.95	4373.55	22.9783	287.2
B40	30	200	0.95	4370.21	22.4054	388.42
C40	10	50	0.95	4258.21	77.3434	10.67
C40	10	100	0.95	4256.21	53.963	20.65
C40	10	150	0.95	4284.21	51.5364	30.84
C40	10	200	0.95	4314.21	46.4327	42.43
C40	10	250	0.95	4314.21	49.7594	52.17
C40	20	50	0.95	4256.21	81.4739	20.63
C40	20	100	0.95	4293.21	62.1289	42.4
C40	20	150	0.95	4312.21	59.3127	62.74
C40	20	200	0.95	4323.21	37.6829	83.53
C40	20	250	0.95	4323.21	44.4747	104.5
C40	30	50	0.95	4273.55	86.406	30.82
C40	30	100	0.95	4294.88	61.2862	62.75
C40	30	150	0.95	4316.21	60.1498	95.12
C40	30	200	0.95	4324.88	43.2666	123.76
C40	30	250	0.95	4318.88	45.5851	153.86

Table 4: Results for Segments with 40 Nodes

Name&Dim.	nSamples	dimSamples	1-*epsilon*	$H^{1-\epsilon}$	Dev-Strd	time (sec.)
A50	10	50	0.95	4960.21	43.795	33.38
A50	10	100	0.95	4954.21	26.7582	63.34
A50	10	150	0.95	4964.21	29.9333	97.89
A50	10	200	0.95	4970.21	34.3802	128.29
A50	10	250	0.95	4968.21	19.9499	162.8
A50	20	50	0.95	4962.21	39.7744	63.34
A50	20	100	0.95	4962.21	40.2492	128.26
A50	20	150	0.95	4969.21	30.9839	191.93
A50	20	200	0.95	4963.21	32.619	252.65
A50	20	250	0.95	4953.21	32.0312	308.64
A50	30	50	0.95	4970.21	47.7703	97.9
A50	30	100	0.95	4954.21	39.3954	191.93
A50	30	150	0.95	4958.88	37.2022	278.73
A50	30	200	0.95	4958.88	31.3688	365.49
A50	30	250	0.95	4950.21	30.0333	453.32
B50	10	50	0.95	4854.21	33.3766	41.39
B50	10	100	0.95	4880.21	45.607	83.79
B50	10	150	0.95	4874.21	21.7256	125.1
B50	10	200	0.95	4874.21	23.5372	174.17
B50	10	250	0.95	4884.21	22.2261	214.45
B50	20	50	0.95	4871.21	47.5184	83.8
B50	20	100	0.95	4887.21	40.2492	174.18
B50	20	150	0.95	4881.21	23.1517	255.68
B50	20	200	0.95	4888.21	32.3419	340.3
B50	20	250	0.95	4888.21	29.0861	430.86
B50	30	50	0.95	4870.88	45.9565	125.09
B50	30	100	0.95	4890.21	37.1483	255.78
B50	30	150	0.95	4888.21	28.9482	387.66
B50	30	200	0.95	4886.88	28.4956	513.3
B50	30	250	0.95	4888.88	28.2489	638.01
C50	10	50	0.95	4496.21	43.2435	88.62
C50	10	100	0.95	4518.21	51.1859	184.92
C50	10	150	0.95	4506.21	41.1339	273.92
C50	10	200	0.95	4494.21	21.8174	363.81
C50	10	250	0.95	4500.21	23.622	436.01
C50	20	50	0.95	4511.21	55.9464	184.86
C50	20	100	0.95	4515.21	45.3652	364.03
C50	20	100	0.95	4515.21	45.3652	364.03
C50	20	150	0.95	4497.21	41.6173	521.95
C50	20	200	0.95	4487.21	23.0651	686.62
C50	20	250	0.95	4495.21	30.5941	851.22
C50	30	50	0.95	4508.21	50.9706	273.88
C50	30	100	0.95	4514.21	53.7215	521.82
C50	30	150	0.95	4491.55	38.9615	766.99
C50	30	200	0.95	4486.88	20.3961	1016.13
C50	30	250	0.95	4490.21	28.5657	1265.88

Table 5: Results for Segments with 50 Nodes

Tables 2, 3, 4, and 5 report the computational results over the four sets of problem instances. The columns display, respectively, the name of the instance and its dimension, the number of samples created, their dimensions, the value $1 - \epsilon$, the average of $H^{1-\epsilon}$ over the number of samples, the standard deviation, and the computing time in CPU seconds.

The experimental results indicate that increasing the dimension of the sample yields more precise solutions. The number of samples is relatively less important. The results also indicate that, in general, the standard deviations are very good, being average estimated values - standard deviation ratios smaller than 2% even in the worst case.

Regarding possible biases, we compared the values obtained by using our algorithm to those obtained by computing $H^{1-\epsilon}$ using a single sample of cardinality 100000. As supported by the figures in Table 6, the values we found are very good.

Name&Dim.	nSamples	dimSamples	1-*epsilon*	$H^{1-\epsilon}$	Dev-Strd	Time sec.
A20	1	100000	0.95	3514.21	0	714.33
A30	1	100000	0.95	3954.21	0	848.39
A40	1	100000	0.95	4554.21	0	1472.54
A50	1	100000	0.95	4954.21	0	5983.55

Table 6: Values of $H^{1-\epsilon}$ Computed over 100000-large Samples

To conclude, notice that the values of $H^{1-\epsilon}$ are remarkably smaller than those of the Hamiltonian path characterizing segments. They should actually be even better in practice. This is because in our experimentation we considered probabilities associated with optional stops uniformly distributed in the interval $(0, 1)$. Yet, in the real word, stops with a value close to 1 would most likely be chosen as compulsory stops. In other words, in an actual implementation of a DAS service, optional stops are not requested frequently and consequently we expect better $H^{1-\epsilon}$ - Hamiltonian path-length ratios.

6 Conclusions

In this paper, we examined from a scheduling point of view a new type of semi-flexible transit service, the Demand-Adaptive Service. Comparing it to traditional transit and dial-a-ride services, we introduced the "new" scheduling requirements of DAS, which we identified as the construction of a *master* schedule. We formalized the master scheduling problem for a single DAS line and proposed a solution framework based on decoupling the origin-destination demand and using a particular sampling technique. Computational results show that the method we propose is efficient and produces high-quality results.

Acknowledgments

Partial funding for this research was provided by Natural Sciences and Engineering Research Council of Canada (NSERC) through its Industrial Research Chair and Discovery grant programs. This research was also partially funded by Regione Lombardia, Italy, through its INGENIO grants.

Fausto Errico wishes to thank Professor Stein W. Wallace for an enlightening conversation held in Montreal in December 2006.

While working on this project, Dr. Teodor Gabriel Crainic was Adjunct Professor at the Department of Computer Science and Operations Research of the Université de Montréal, Canada, and at Molde University College, Norway.

References

[1] Carpaneto, G., Dell'Amico, M., and Toth, P. Exact Solution of Large-scale, Asymmetric Traveling Salesman Problems. *ACM Trans. Math. Softw.*, 21(4):394–409, 1995.

[2] Ceder, A. and Wilson, H.M. Public Transport Operations Planning. In *Design and Operation of Civil and Environmental Engineering systems*, pages 395–434. John Wiley & Sons, Inc., New York, 1997.

[3] Cordeau, J.-F. and Laporte, G. The Dial-a-Ride Problem (DARP): Variants, Modeling Issues and Algorithms. *4OR*, 1(2):89–101, 2003.

[4] Crainic, T.G., Malucelli, F, and Nonato, M. A Demand Responsive Feeder Bus System. In *CD-ROM of 7th World Congress on Intelligent Transport Systems*. 7WC-ITS, Torino, Italia, 2000.

[5] Crainic, T.G., Malucelli, F, and Nonato, M. Flexible Many-to-few + Few-to-many = An Almost Personalized Transit System. In *Preprints TRISTAN IV - Triennial Symposium on Transportation Analysis*, volume 2, pages 435–440. Faculdade de Ciências da Universidade de Lisboa and Universidade dos Açores, São Miguel, Açores, Portugal, 2001.

[6] Crainic, T.G., Malucelli, F, Nonato, M., and Guertin, F. Meta-Heuristics for a Class of Demand-Responsive Transit Systems. *INFORMS Journal on Computing*, 17(1):10–24, 2005.

[7] Errico, F. *The design of flexible transit systems: models and solution methods.* PhD thesis, Politecnico di Milano, Italy, 2008.

[8] Hickman, M. and Blume, K. A Method for Scheduling Integrated Transit Service. In *Proceedings of the 8th International Conference on Computer-Aided Scheduling of Public Transport (CASPT)*, 2000.

[9] Horn, M.E.T. Multi-modal and demand-responsive passenger transport systems: a modelling framework with embedded control systems. *Transportation Research A*, 36:167–188, 2002.

[10] Ioachim, I., Desrosiers, J., Dumas, Y., and Solomon, M.M. A Request Clustering for Door to Door Handicapped Transportation. *Transportation Science*, 29(1):63–68, 1995.

[11] Malucelli, F, Nonato, M., Crainic, T.G., and Guertin, F. Adaptive Memory Programming for a Class of Demand-Responsive Transit Systems. In Voß, S. and Daduna, J.R., editors, *Computer-Aided Scheduling of Public Transport*, volume 505 of *Lecture Notes in Economics and Mathematical Systems*, pages 253–273. Springer, Berlin, 2001.

[12] Malucelli, F, Nonato, M., and Pallottino, S. Some Proposals on Flexible Transit. In Ciriani, T.A., Johnson, E.L., and Tadei, R., editors, *Operations Research in Industry*, pages 157–182. McMillian, 1999.

[13] Quadrifoglio, L., Dessouky, M.M., and Palmer, K. An Insertion Heuristic for Scheduling Mobility Allowance Shuttle Transit (MAST) Services. *Journal of Scheduling*, 10(1):25–40, 2007.

[14] Toth, P. and Vigo, D. Fast Local Search Algorithms for the Handicapped Persons Transportation Problem. In Osman, I.H. and Kelly, J.P., editors, *Meta-Heuristics: Theory & Applications*, pages 677–690. Kluwer Academic Publishers, Norwell, MA, 1996.