Time Scheduling of Transit Systems with Transfer Considerations Using Genetic Algorithms

Kalyanmoy Deb
Department of Mechanical Engineering
Indian Institute of Technology
Kanpur, Pin 208 016, India
E-mail: deb@iitk.ernet.in

Partha Chakroborty
Department of Civil Engineering
Indian Institute of Technology
Kanpur, Pin 208 016, India
E-mail: partha@iitk.ernet.in

Abstract

Scheduling of a bus transit system must be formulated as an optimization problem, if the level of service to passengers is to be maximized within the available resources. In this paper, we present a formulation of a transit system scheduling problem with an objective of minimizing the overall waiting time of transferring and non-transferring passengers while satisfying a number of resource and service related constraints. It is observed that the number of variables and constraints for even a simple transit system (a single bus station with three routes) is too large to tackle using classical mixed-integer optimization techniques. The paper shows that genetic algorithms (GAs) are ideal for these problems, mainly because of certain aspects of GAs: (i) GAs naturally handle binary variables, thereby taking care of transfer decision variables which amounts to the majority of the decision variables in the transit scheduling problem, and (ii) GAs allow procedure-based declarations, thereby allowing complex algorithmic approaches (involving if-then-else conditions) to be handled easily. The paper also shows how easily the same GA procedure with minimal modifications can handle a number of other more pragmatic extensions to the simple transit scheduling problem: Buses with limited capacity, buses which do not arrive exactly as per scheduled times, and a multiple-station transit system having common routes among bus stations. Simulation results show success of GAs in all these problems and suggest the application of GAs in more complex scheduling problems.

Keywords: Genetic algorithms, Mixed-integer programming, Reliability, Time scheduling, Transfer time, Transit system

1 Introduction

In order to provide better level of service to passengers, public transport systems must be efficient. A major factor which increases the efficiency of a public transport system is proper scheduling of its transit vehicles (namely, buses). Besides other factors, a good schedule should minimize both the waiting time for passengers as well as the transfer time of passengers from one route to another.

When formulated mathematically, the time scheduling problem becomes a mixed-integer nonlinear programming problem (MINLP) having a large number of resource and service related constraints. Although attempts have been made in the past to find an optimal schedule of a simplified model using classical optimization techniques (Bookbinder and Désilets, 1992; Kikuchi and Parameswaran, 1993), it is observed that this is an extremely difficult task even for a small transit network. The difficulty arises mainly because of the large number of variables and constraints, discrete nature of variables, and nonlinearities involved in the objective function and the constraints. In this paper, we use genetic algorithms (GAs) to solve the time scheduling problem.

GAs are particularly suited for the scheduling problem, because it allows an efficient reformulation of the problem, which reduces the difficulties mentioned above. Although GAs have been used in other kinds of scheduling problems related to transportation engineering (Martinelli and Teng, 1995; Wren and Wren, 1995), none of them are related to the time scheduling problem. A number of different types of realistic transit scheduling problems are also formulated here: (i) a transit system where buses have limited available bus capacity, (ii) a transit system where buses do not arrive at the station exactly at their scheduled times, and (iii) a transit system having multiple transfer stations. It is found that the same GA procedure with minimal modification can be successfully used to solve all of the above cases. The results show the efficacy of using GAs as the solution tool for the development of optimal transit schedules.

In the remainder of the paper, the time scheduling problem is described. Thereafter, a mathematical formulation of a generic transit system scheduling problem is presented. The characteristic of the search space is outlined in the context of solving such problems using classical optimization techniques. Thereafter, a discussion on why GAs are naturally suitable to solve such complex problems is provided. Finally, four different types of transit scheduling problems are described and solved using GAs to show the efficacy of the proposed method.

2 Time Scheduling of Transit Systems

A typical transit system consists of buses plying on various routes which intersect at a number of transfer stations. The purpose of a transit system is to transport passengers from their station of origin to their station of destination. However, direct routes between all pairs of origins and destinations do not exist. Passengers with such pairs of origins and destinations, therefore, have to use more than one route to reach their destination. These passengers generally come to a transfer station on some route and wait to transfer to another route which will eventually take them to their destinations. Thus, at any transfer station, there are two types of passengers (customers)—transferring passengers and non-transferring passengers (passengers whose station of origin is the transfer station itself).

A typical transit system operating on a network of streets (light lines) is shown in figure 1. The

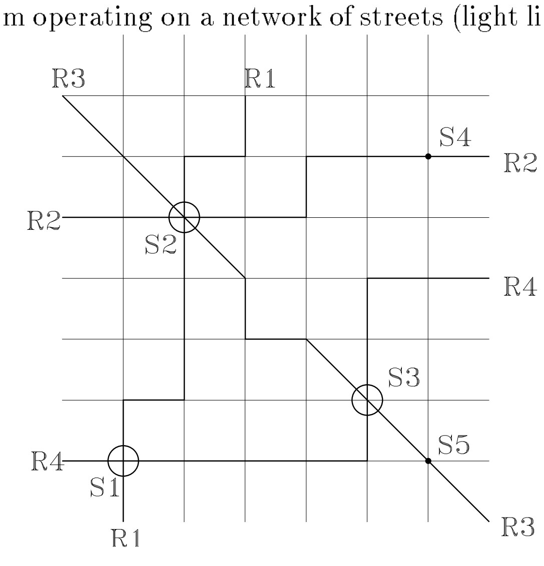

Figure 1: A typical transit system network.

bold lines represent the routes and circles represent the transfer stations. Consider the transfer station S1. At this station, three routes intersect. Two types of passengers exist: (i) transferring passengers; for example, the passengers who want to go from S1 to S4, may arrive at station S2 on route R1 and wait for a bus on route R2, (ii) non-transferring passengers; for example, the passengers who want to go from S2 to S5.

One of the objectives of a transit system design is to provide a good level of service to its users. A cost-effective way to achieve this goal is to optimally schedule the buses within the available resources. An optimal schedule coordinates the arrival and departure times of the buses so as to minimize the waiting time of passengers. Thus, the time scheduling of transit system design problem is an optimization problem having an objective and a number of constraints, which we describe in the next section.

As there are two types of passengers, there are two types of waiting times—initial waiting time (IWT), the waiting time of non-transferring passengers, and transfer time (TT), the waiting time of transferring passengers. Both IWT and TT depends on the arrival and departure time of buses in all routes. The objective function, total waiting time (TWT), is the sum of IWT and TT for all passengers.

The available resource constraints such as fleet size (the number of buses available for each route) and available bus capacity (the number of persons who can board the bus at the station) are assumed to be known. In addition, a number of service related constraints such as the minimum and maximum stopping time of buses at stations, the maximum headway (the time between two consecutive buses of the same route), and others are considered.

3 Mathematical Formulation

In this section, we formulate the optimization problem of a transit network scheduling as a mathematical program (MP). First, we outline the notations used in the formulation and then present the formulation.

a_{ij}^l : Arrival time of the l-th bus of j-th route at the i-th station.

d_{ik}^m : Departure time of the m-th bus of the k-th route at the i-th station.

h_{ij} : Policy headway of the j-th route at the i-th station; this is the maximum bound on the difference in the arrival times of two successive buses.

M : An arbitrary large number used in the formulation.

$s_{ij}^{\min}$: Minimum stopping time of a bus in j-th route at the i-th station.

$s_{ij}^{\max}$: Maximum stopping time of a bus in j-th route at the i-th station.

$t_{(i-1)ij}^l$: Travel time of the l-th bus of the j-th route from the $(i-1)$-th station to the i-th station.

T : Maximum transfer time.

$v_{ij}^l(t)$: Arrival pattern of passengers for the l-th bus on the j-th route at the i-th station.

w_{ijk}^l : Transfer volume from the l-th bus of j-th route to the k-route at the i-th station.

δ_{ijk}^{lm} : A binary variable which is one if a transfer from the l-th bus of the j-th route to the m-th bus of the k-th route at the i-th station is possible and optimal; zero otherwise (also refer to section 3.2).

It may be noted that the following assumptions are made in the formulation given below.

1. It is assumed that buses will arrive and depart exactly as per the schedule. This is referred to as exact (strict) schedule adherence. When arrival time of buses deviate from the scheduled arrival time, it is referred to as stochastic schedule adherence.

2. It is also assumed that the available capacity of a bus arriving at the station is enough to accommodate all passengers who are waiting to board this bus. This is referred to as unlimited bus capacity in the rest of the paper. If this assumption does not hold, we refer to that condition as limited bus capacity condition.

The following is the MP formulation (Chakroborty, Deb, and Srinivas, in press). The constraints and the objective function are explained later.

$$
\begin{array}{lll}
\text{Minimize} & \displaystyle\sum_i \sum_j \sum_{\substack{k \\ k\neq j}} \sum_l \sum_m \delta_{ijk}^{lm}(d_{ik}^m - a_{ij}^l) w_{ijk}^l + \sum_i \sum_j \sum_l \int_0^{d_{ij}^l - d_{ij}^{l-1}} v_{ij}^l(t)(d_{ij}^l - d_{ij}^{l-1} - t)dt & \\
\text{Subject to} & & \\
& G1 \equiv d_{ij}^l - a_{ij}^l \leq s_{ij}^{\max} \qquad \forall i,j,l & \\
& G2 \equiv d_{ij}^l - a_{ij}^l \geq s_{ij}^{\min} \qquad \forall i,j,l & \\
& G3 \equiv a_{ij}^l - a_{ij}^{l-1} \leq h_{ij} \qquad \forall i,j,l & (1)\\
& G4 \equiv (d_{ik}^m - a_{ij}^l)\delta_{ijk}^{lm} \leq T \qquad \forall i,j,k,l,m & \\
& G5 \equiv d_{ik}^m - a_{ij}^l + M(1-\delta_{ijk}^{lm}) \geq 0 \qquad \forall i,j,k,l,m & \\
& G6 \equiv \sum_m \delta_{ijk}^{lm} = 1 \qquad \forall i,j,k,l & \\
& G7 \equiv a_{ij}^l - d_{(i-1)j}^l = t_{(i-1)ij}^l \qquad \forall j,l,i \geq 2 &
\end{array}
$$

The decision variables in the formulation given in Equation 1 are the arrival times a_{ij}^l, the departure times d_{ij}^l and δ_{ijk}^{lm} values. The variable δ_{ijk}^{lm} is a zero-one integer variable. A value of zero means that the transfer from the l-th bus of the j-th route to the m-th bus of the k-th route at the i-th transfer station is either not possible (that is, the l-th bus of the j-th route arrives after the m-th bus of the k-th route has departed) or non-optimal to the passengers (that is, there is at least one bus on the k-th route which departs after the arrival of the l-th bus of the j-th route and before the departure of the m-th bus of the k-th route). A value of one means otherwise.

3.1 Objective function

The objective function consists of two terms; the first term represents the total transfer time for all the transferring passengers at all transfer stations, and the second term is the total initial waiting time for all the passengers at their station of origin.

The transfer time for each of the w_{ijk}^l passengers who want to transfer from l-th bus of the j-th route to the k-th route at the i-th station is

$$(d_{ik}^m - a_{ij}^l)\delta_{ijk}^{lm},$$

where d_{ij}^m is the departure time of the m-th bus of the k-th route from the i-th station and a_{ij}^l is the arrival time of the l-th bus of the j-th route at the i-th station. The overall transfer time is calculated by multiplying the above term by w_{ijk}^l and summing over all the buses, routes and stations.

The second term is arrived at by assuming that passengers boarding the l-th bus of the j-th route at the i-th station arrive according to some function $v_{ij}^l(t)$ (where t is measured from d_{ij}^{l-1}) between d_{ij}^{l-1} and d_{ij}^l. The integral in the second term is therefore the total initial waiting time for all passengers boarding the l-th bus of the j-th route at the i-th station. This term when summed over all buses of all routes at all stations gives the total IWT.

3.2 Constraints

There are seven types of constraints, as shown in the formulation. Constraints G1 through G4 are service related constraints. Constraints G1 state that the stopping time $(d_{ij}^l - a_{ij}^l)$ for the l-th bus on the j-th route at the i-th station should be less than or equal to the maximum stopping time, $s_{ij}^{\max}$, and Constraints G2 state that the stopping time $(d_{ij}^l - a_{ij}^l)$ should be greater than or equal to the minimum stopping time, $s_{ij}^{\min}$. Constraints G3 state that the headway (time difference between arrivals of two consecutive buses) should not be larger than a stipulated maximum headway, h_{ij}. Constraints G4 restrict the transfer time for any transfer to be less than or equal to the maximum transfer time T.

Constraints G5 through G7 are logic constraints which define the feasibility of transfers and dependency arrival times. They are explained below. Constraints G5 assure that δ_{ijk}^{lm} is zero whenever a transfer from the l-th bus of j-th route to the m-th bus of the k-th route at the i-th station is not possible. Constraints G6 along with Constraints G5 assure that transfer from a particular bus of a particular route is made to only one of the buses of another route (among many buses to which a transfer was possible). This fact, in the context defined by the objective function, ensures that transfer is made only to the next available transit vehicle. Constraints G7 incorporate the dependency of the arrival time of a bus at a particular station on the departure time of the same bus from the previous station.

The resource related constraint of fleet size is implicit in the formulation. That is, we assume that we know the number of buses in each route. However, the formulation presented above assumes that the available bus capacity at any station is enough to accommodate all passengers waiting for the bus. On the other hand, if the bus capacity is limited one needs to maintain a queue for the passengers who arrive at the station. The description of queues within the MP formulation framework is a very difficult task and hence we postpone the discussion on limited bus capacity till section 6.2.

In the above formulation, it is assumed that the buses will arrive and depart strictly as per the schedule. In general, this a not true and the actual arrival and departure times of the buses are stochastic in nature. Such stochasticity in arrival and departure times is difficult to incorporate in the MP formulation because of the dependencies among the decision variables. We shall deal with this case later in section 6.3.

4 Characteristics of the MP Formulation

In this section, we discuss the characteristics which make the above problem difficult to solve using classical techniques.

1. Discrete search space
2. Nonlinear search space
3. Dimension of search space

In the above formulation, there are three types of decision variables: arrival times, a_{ij}^{l}, departure times, d_{ij}^{l}, and transfer variables, δ_{ijk}^{lm}. Among them, the δ variables are binary, taking a `1` or a `0`. This makes the search space discrete. Further, since the other two types of variables represent scheduled arrival and departure times of buses, it is desirable that they are in minutes rather than in fractions of a minute. For example, a schedule arrival time of 9:35 AM is better than 9:34:22.5 AM. Hence, although not absolutely necessary, it is better (and pragmatic) that these variables are represented as discrete quantities. Thus, the above problem is a discrete programming problem, which necessitates the use of mixed-integer programming techniques such as the Branch-and-Bound method. Such methods are highly iterative in nature and not very efficient. Moreover, if the standard Branch-and-Bound technique is used to handle δ variables, the algorithm may require computation of TT term for non-binary values of δ variables. This may result in intermediate solutions which are not meaningful.

The nonlinearity in the above formulation comes from the objective function and from Constraints G4. A common method of handling such nonlinearities is to use the Frank-Wolfe technique (Deb, 1995). This technique uses successive linear approximations of the nonlinear functions to obtain solutions. Further, this technique introduces additional linear constraints in order to handle nonlinear constraints. These fix-ups make this technique slow and often cause convergence problems.

It can be seen from the formulation that there are $O(br^2n^2)$ number of variables for a network with b transfer stations, an average of r routes through each transfer station, and an average of n buses on each route. For example, for a small network with one transfer station ($b = 1$), 3 routes through the transfer station ($r = 3$), and 10 buses on each route ($n = 10$), there are a total of 660 variables. Of these, 600 are δ variables. Thus, even for a small network, together with the nonlinearity and discreteness of the search space, the dimensionality of the problem makes the problem hard to solve using classical methods. Not only is the dimensionality of the space is large, the number of constraints is also large. A little computation will reveal that there are also $O(br^2n^2)$ constraints associated with the problem. For the above single-station network, there are a total of 1,350 linear and nonlinear constraints.

These characteristics of the problem motivate us to use a different optimization technique which may reduce some of the above difficulties and help us obtain optimal schedules.

5 Motivation for Using Genetic Algorithms

On examining the MP formulation it becomes clear that (i) a large number of constraints arise from specifying bounds on stopping times and headways (Constraints G1, G2, and G3), (ii) constraints G5, G6 and the delta variables contribute largely to the complexity of the problem; yet these variables as well as the constraints are present only to state that passengers transfer to the next available bus on the route to which they intend to transfer, and (iii) Constraints G7 only state that arrival time at a subsequent station is related to the departure time at a former station.

Since, genetic algorithms (GAs) naturally work in an environment where variables are always bounded, the bound constraints can be eliminated by using GAs. Further, GAs allow external procedure-based declarations, that is, GAs can use information obtained from procedures which are external to the optimization algorithm. This feature of GAs allow us to eliminate Constraints G5, G6 and G7 by using small procedures (these are discussed later). This feature also eliminate δ variables (which are binary) from the set of decision variables, as discussed in Section 6.1.1. As a result, the complexity of the GA formulation is far less than that of the MP formulation discussed earlier.

In section 4 it was stated that it is desirable that the arrival and departure times are represented as discrete quantities. Although, such representation in the MP formulation would have increased the complexity manifold, doing so in the GA formulation has no effect on the complexity. This is so, because GAs with binary string coding inherently work with discrete search spaces.

Hence, we use a binary-coded GA (with reproduction, crossover, and mutation) to solve various cases of the scheduling problem. These cases are described in the next section. It is important to note that even though the cases are widely different (involving stochasticity in arrival times, limitations on bus capacity, and others), the same GA-based algorithm can be used as the solution tool in all the problems.

6 Case Studies

In this section, we apply genetic algorithms to four different cases of transit scheduling. Each case considers a different combination of network characteristics, schedule adherence, and available bus capacity.

Case 1: The network consists of only one transfer station; buses are assumed to arrive and depart exactly as per the schedule; available bus capacity is unlimited.

Case 2: The network consists of one transfer station; buses are assumed to arrive and depart exactly as per the schedule; available bus capacity is limited.

Case 3: The network consists of one transfer station; arrival time buses are assumed to vary stochastically around the scheduled arrival time; available bus capacity is unlimited.

Case 4: The network consists of three transfer stations; buses are assumed to arrive and depart exactly as per the schedule; available bus capacity is unlimited.

We discuss the GA formulation of each of the above cases and present the simulation results in the following subsections.

6.1 Case 1: Single station, exact adherence, unlimited capacity

This is a special case of the problem described earlier (Section 3). Here, the number of transfer stations, b is equal to one. The mathematical formulation given in equation 1 remains the same except that constraints G7 are no longer necessary. This is because in this case there is only one transfer station. Even in this problem, the complexity is $O(r^2n^2)$.

In earlier study (Chakroborty, Deb, and Subrahmanyam, 1995), we have attempted to solve the resulting MINLP problem (equation 1) for this case using the Branch-and-Bound technique of NAG software. On a Convex C220 vector machine, the algorithm repeatedly failed to converge to any solution. We now present the GA-based formulation for this problem.

6.1.1 GA formulation

We discuss how the formulation given in equation 1 (the subscript i is dropped for a single transfer station) is revised in order to use GA. Specifically, we present the binary string representation of variables, procedure-based declarations to take care of some of the constraints, and procedure of computing the objective function.

Recall that the scheduling problem has three types of variables—arrival times a_j^l, departure times d_j^l, and transfer variables δ_{jk}^{lm}. Realizing that the constraints involve headway and stopping time bounding constraints (Constraints G1 through G3), we use the following two types of variables in the GA formulation:

$x_j^l = (a_j^l - a_j^{l-1})$	The headway between l-th and $(l-1)$-th bus of the j-th route
$dx_j^l = (d_j^l - a_j^l)$	Stopping time of l-th bus of the j-th route

The arrival and departure times (a_j^l and d_j^l) can be computed from the above two types of variables with an initial condition $a_j^0 = 0$ and using the following recursive equations:

$$\begin{aligned} a_j^l &= a_j^{l-1} + x_j^l, \\ d_j^l &= a_j^l + dx_j^l. \end{aligned}$$

We show later how the transfer variables δ_{jk}^{lm} can be derived from these arrival and departure times and how some of the constraints can be eliminated. Now, we show a typical GA string representing a complete transit schedule.

In each route, there are as many arrival and departure times (or, x_j^l and dx_j^l) as there are buses. However, to simplify the matter, we assume that all buses in a particular route has the same stopping time. Thus, $d_j^l = d_j$. Thus, for n_j buses in the j-th route at a station, there are a total of $n_j + 1$ variables. However, in order to restrict the schedule for a particular time window (say $\mathcal{T}$), we always fix the arrival time of the last bus at the end of the scheduling window. Thus, we vary the arrival and departure times of $n_j - 1$ buses, instead of all n_j buses. Thus, there are a total of n_j variables for each route in each station in the GA formulation. We represent each variable with binary substrings. Finally, we concatenate all n_j variables for other routes together to get the complete string. The following string shows the sequence of variables used in a GA string to represent all variables needed to fully represent a schedule for a single transfer station having n_R routes:

$$((dx_1\; x_1^1\; x_1^2\; \ldots x_1^{n_1})\; \ldots (dx_R\; x_R^1\; x_R^2\; \ldots x_R^{n_R}))$$

Coding stopping time and headways of all buses for a particular route together helps propagate good partial schedules through GA operators, a matter which is important from the schema processing point of view (Goldberg, 1989; Holland, 1975; Radcliffe, 1991). We now discuss how the constraints are handled in the GA formulation.

Constraints G1 through G3 (shown below) are variable bounds and can be handled easily by limiting the lower and upper bounds in the decoding of binary substrings corresponding to the headway and stopping time variables:

$$\begin{array}{lll} G1 \equiv & x_j^l \leq h_j & \forall j, l, \\ G2 \equiv & dx_j^l \leq s_j^{\max} & \forall j, l, \\ G3 \equiv & dx_j^l \geq s_j^{\min} & \forall j, l. \end{array}$$

The lower bound on headway x_j^l is kept at a suitable lower limit.

Constraints G5 and G6 can be eliminated by using the following procedure to calculate δ_{jk}^{lm}:

```
flag=0
for all combinations of routes (j,k) and buses (l,m)
        if d_j^l < a_k^m or flag==1 then
                                δ_jk^lm = 0
        else
                                δ_jk^lm = 1
                                flag = 1;
```

Thus, the revised formulation of the scheduling problem reduces to minimizing the TWT term subject to only one type of constraints (Constraints G4). These constraints are handled in GA using a bracket-operator penalty term Ω (Deb, 1995; Reklaitis, Ravindran, and Ragsdell, 1983):

$$\Omega(g) = \begin{cases} Rg^2, & \text{if } g < 0; \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where g is the left-hand side function value of the constraint represented as $G \geq 0$. A constant penalty parameter R of 10^3 is used. The number of problem variables and constraints in the above GA formulation are $2rn$ and $r(r-1)n^2$, respectively. The reductions in the number of variables (from quadratic to linear in r and n) and in the number of constraints (from $3r(r-1)n^2 + 3rn$ to $r(r-1)n^2$) are the primary advantages of using GAs in the transit scheduling problem. In the following, we present the simulation results of GAs.

6.1.2 Simulation results

To demonstrate the proof-of-principle results, we assume that there a total of 30 buses (10 in each route) available to ply in three routes. We choose the scheduling time window ($\mathcal{T}$) to be from 7 AM to 11 AM (240 minutes). Following parameters are chosen for the scheduling problem:

- Minimum headway, $h_j^{\min} = 14$ minutes, except for the first bus where $h_j^{\min} = 0$ minutes.
- Maximum headway, $h_j^{\max} = 45$ minutes, except for the first bus where $h_j^{\max} = 31$ minutes.
- Minimum stopping time $s_j^{\min} = 2$ minutes.
- Maximum stopping time $s_j^{\max} = 5$ minutes.
- Maximum transfer time $T = 30$ minutes.

Using the above variable bounds, let us compute the total string length to represent a complete schedule. For j-th route, headways (x_j^l, $l = 1, 2, \ldots, 9$) for nine buses and one stopping time (dx_j) are variables. Allowing only integer values of the variables, the chosen variable bounds suggest that each headway requires 5 bits (having 32 alternatives) and each stopping time requires 2 bits (having 4 alternatives). Thus, the total string length for a complete schedule becomes $[3(9\times 5+2)]$ or 141. The following GA parameters are chosen (using suggestions taken from the GA literature and performing some trail-and-error experiments):

- Population size is 350.
- Binary tournament selection is used.
- Single-point crossover on the complete string is used.
- Crossover probability is $p_c = 0.95$.
- Bit-wise mutation with probability $p_m = 0.005$ is used.
- GA is terminated when 200 generations are exceeded or the difference in population minimum and average is less than 10^{-7}.

First, we consider an objective of minimizing the transfer time (TT) only. In this case, the optimal schedule will be the one where buses of different routes arrive at and depart from the transfer station approximately at the same time. Figure 2 shows the best schedule obtained by GA. The arrows pointing to the horizontal lines represent arrival and arrows emanating from the horizontal lines represent departure. This schedule requires only 160 minutes of total TT, whereas the best schedule in the initial population required a TT of 1,170 minutes. It can be seen from the figure that, as expected, the buses of the three routes arrive and depart more or less at the same time.

We now investigate the effect of initial waiting time (IWT) alone on the optimal schedule. We choose the arrival pattern of passengers ($v_j^l(t)$) to be triangular with a height L_j of 1.244, as shown in Figure 3. In this case, we expect the optimal schedule to have each headway of a route to be equal, because this problem is similar to a multi-variable (x_i) optimization problem of minimizing $\sum_i x_i^2$ subject to $\sum_i x_i =$ constant (Note that IWT term is quadratic in terms of headways and the sum of all headways on any route must be 240 minutes). Figure 4 shows the optimal schedule—all headways are more or less equal.

To investigate the effect of both TT and IWT on the optimal schedule, next we consider the objective function with TT ($w_{jk}^l = 1$) and IWT terms. The optimized schedule is shown in Figure 5. It is interesting to note that the optimal schedule for only IWT case is also an optimal solution for the only TT case. Thus, when TWT is minimized, the optimal solution will be the same as that in the only IWT case. Both Figures 4 and 5 show that IWT value in the optimal solution for only IWT case and TWT case are the same and equal to 4,477 minutes.

Above simulation results show how easily GAs can be used to find optimized schedules. Similar performance of GAs are also observed in many other test cases including unequal buses in each

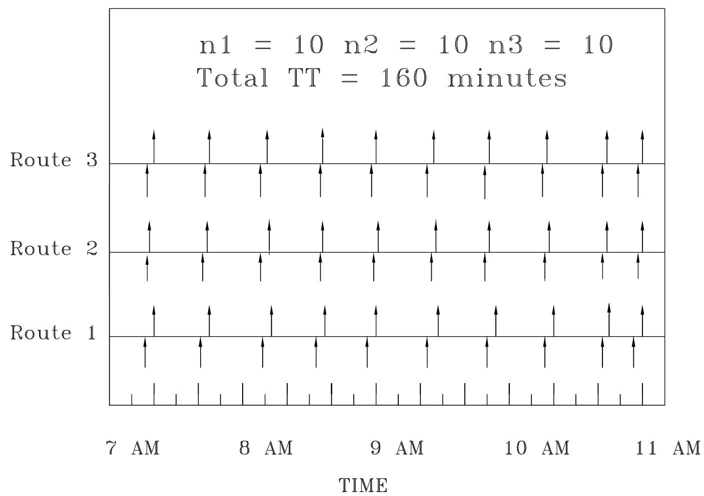

Figure 2: Optimized schedule for TT obtained by GAs.

route and non-uniform arrival patterns of passengers. Based on the efficiency and reliability of the above approach, we tackle more complicated and realistic transit scheduling problems in the following subsections.

6.2 Case 2: Single station, exact adherence, limited capacity

In the above case, the available bus capacity was assumed to be unlimited. Hence, it was not necessary to keep track of the arrival time of individual passengers. However, when the available bus capacity is limited, all passengers (both transferring as well as non-transferring passengers) who arrive to board a particular bus may not able to do so. Therefore, it becomes imperative that we maintain a queue of arriving passengers for each route. Non-transferring passengers join the queue of their interest as they arrive at the station. Passengers transferring to a particular route join that queue as and when they are brought to the station by a bus of another route. The queue for a route keeps growing in this manner till the arrival of a bus on that route. If the available capacity of this bus (say, l-th bus of j-th route) is C_j^l, then, at this time, the first C_j^l passengers from the queue board the bus. The rest forms the initial queue for the next bus on the

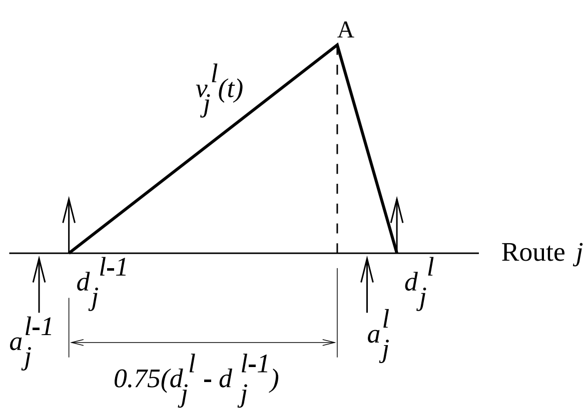

Figure 3: The triangular arrival pattern of passengers.

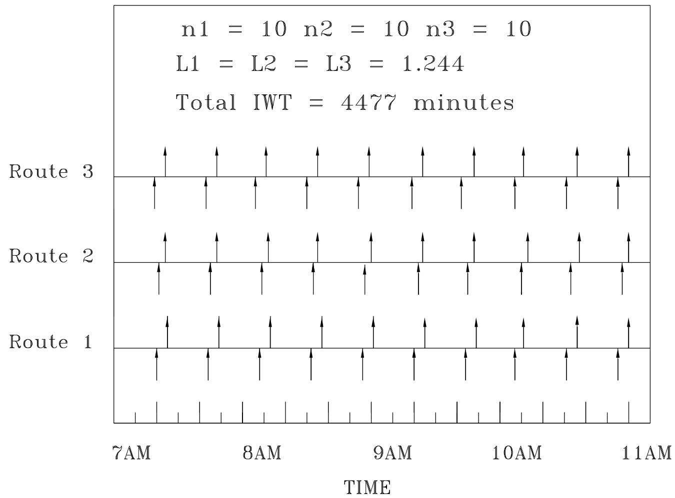

Figure 4: Optimized schedule for IWT obtained by GAs.

same route. This process continues. Notice that the IWT and TT will have to be computed by summing the times spent by individual non-transferring and transferring passengers in the queue, respectively.

Although the above description seems simple, anyone familiar with modeling transient behavior of queues would know the difficulties in implementing the above process in terms of constraints and objective function. Given that the classical optimization techniques failed to obtain optimal solution to the simpler problem stated in Case 1, we do not attempt to solve this problem using classical techniques.

6.2.1 GA formulation

In this GA formulation, the string representation procedure remains the same as in Case 1. However, the evaluation of a string varies considerably. As earlier, a string still decodes to arrival and departure time of each bus on each route. Given the arrival pattern of non-transferring passengers ($v_j^l(t)$) and transferring passengers (ω_{jk}^l), and arrival and departure time of each bus, the queuing process described above is implemented through a procedure. Using this procedure, IWT and TT terms are computed. Notice that the procedure for determining δ_{jk}^{lm} is no longer necessary, because the δ_{jk}^{lm} variables as defined earlier are no longer meaningful.

In the GA formulation described in Case 1, we penalized a schedule which gives rise to a transfer time more than T. In addition to this, we penalize a schedule if at the end of the scheduling time window there are some passengers for any route j left unserved, when the total available bus capacity for the j-th route ($\sum_l C_j^l$) is more than the total number of passengers arriving for this route.

6.2.2 Simulation results

In order to illustrate the sensitivity of optimized schedule to the available bus capacity, different cases are studied. Here, we provide two such cases.

In the first case, the objective function consists of only IWT term and arrival pattern of passengers is such that more passengers arrive in the latter half than in the former half of the scheduling time window[1]. It is also assumed that the arrival pattern of passengers on all the

[1]The locus of the vertices of the inter-bus arrival patterns of passengers (for example, A in Figure 3) is assumed

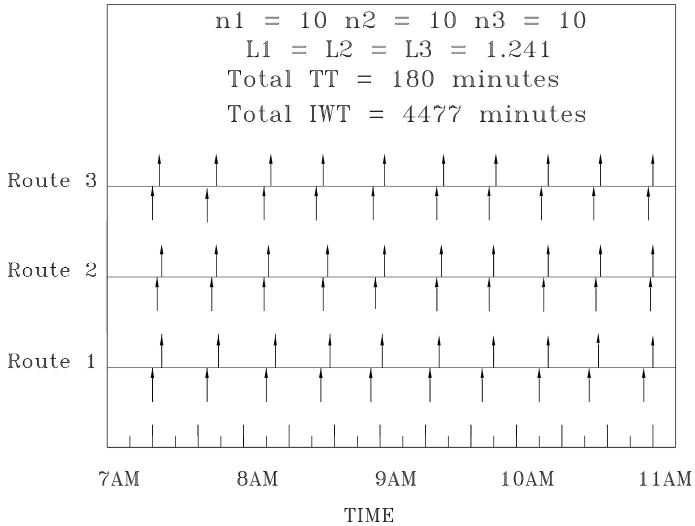

Figure 5: Optimized schedule for TWT obtained by GAs.

routes are the same. GA parameters are kept the same as before except that a population size of 600 and a maximum generation of 1,200 are used. Figure 6 shows the optimized schedule for the case described above (with $\mu = 0.2$ in all routes, thereby exhibiting the peak of locus of maximum passenger arrival at $(1 - 0.2) \times 240$ or 192 minutes after 7AM[2]) and available bus capacity of 35. As can be seen, more buses arrive between 120 and 240 minutes than between 0 and 120

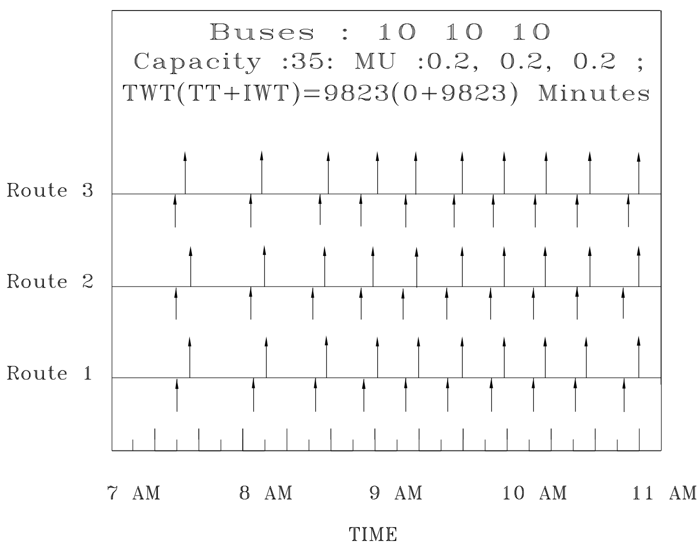

Figure 6: Optimized schedule for IWT with more passengers arriving in the latter half of scheduling window.

minutes. The total number of passengers arriving for a route is approximately 320, whereas the

to follow the following function: $3.7\left(\frac{\tau}{1-\mu}\right)^{1-\mu}\left(\frac{1-\tau}{\mu}\right)^{\mu}$, where $\tau \in [0,1]$ is the non-dimensionalized time across the scheduling time window and μ ($\in [0,1]$) is a parameter which fixes the maximum of the above function at $\tau = 1-\mu$. In the figures, the parameter μ is denoted as MU.

[2]However, it is important to note that for two consecutive buses in a route with departure times d_j^{l-1} and d_j^l, the peak of passenger arrival remains always at $0.75(d_j^l - d_j^{l-1})$ from d_j^{l-1} (refer to Figure 3).

total available bus capacity in each route is 35×10 or 350.

For the above problem description and an assumption of unlimited bus capacity, there would exist an optimal schedule. This schedule will provide the least IWT per passenger for the given problem. However, when the bus capacity becomes limited, this optimal schedule may not be achievable. One can only expect that as the available bus capacity increases, the optimized schedule should tend towards the optimal schedule corresponding to unlimited bus capacity. Alternatively, one should expect as bus capacity increases IWT per passenger for an optimized schedule should decrease. Figure 7 shows this fact.

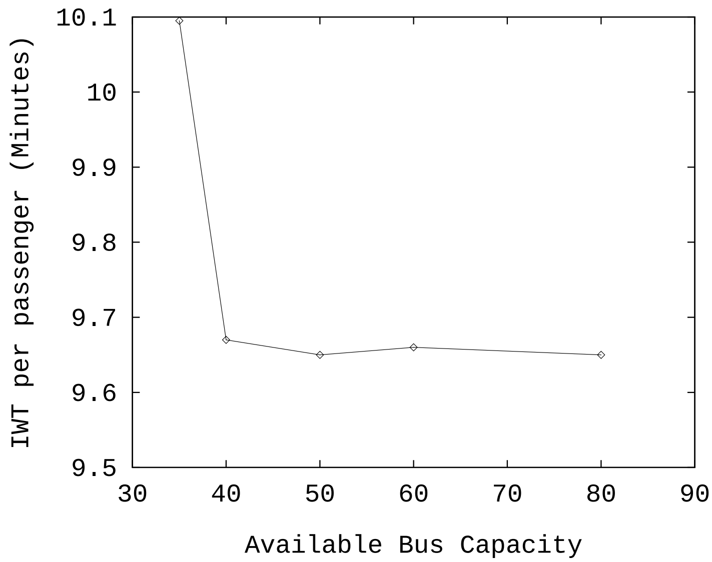

Figure 7: IWT per passenger for optimized schedules versus available bus capacity.

In the other case, the TWT is considered as the objective function. The arrival pattern of passengers for the first route is such that more passengers arrive in the latter half (we choose $\mu = 0.2$); for the second route is such that more passengers arrive during the middle of the scheduling period (we choose $\mu = 0.5$); for the third route is such that more passengers arrive in the former half (we choose $\mu = 0.8$). The total number of non-transferring passengers is approximately the same as before and the total number number of transferring passengers is approximately 100. Figure 8 shows the optimized schedule. As expected, the buses are distributed as per the arrival pattern of passengers. In order to see the effect of TT on the above optimized schedule, we put more weightage to TT in the objective function (10 times more than IWT) and rerun GAs. Figure 9 shows that average TT per transferring passenger (total TT divided by the total number of transferring passengers)has reduced from 9.40 minutes to 6.56 minutes. The figure also indicates that there are more transfers which require minimal transfer times (that is, buses on the different routes arrive more or less at the same time) than in the previous case (Figure 8).

6.3 Case 3: Single station, stochastic adherence, unlimited capacity

It is unrealistic to assume that buses will arrive at and depart from a transfer station exactly as per schedule. Thus, although the level of service to passengers can be increased by using an optimal schedule, if buses do not adhere to this schedule, the realized level of service may not be as expected. However, if, in such situations, the optimal schedule is determined based on the assumption that buses may arrive stochastically around the announced scheduled times, the realized level of service may be better than that with an optimal schedule obtained using exact

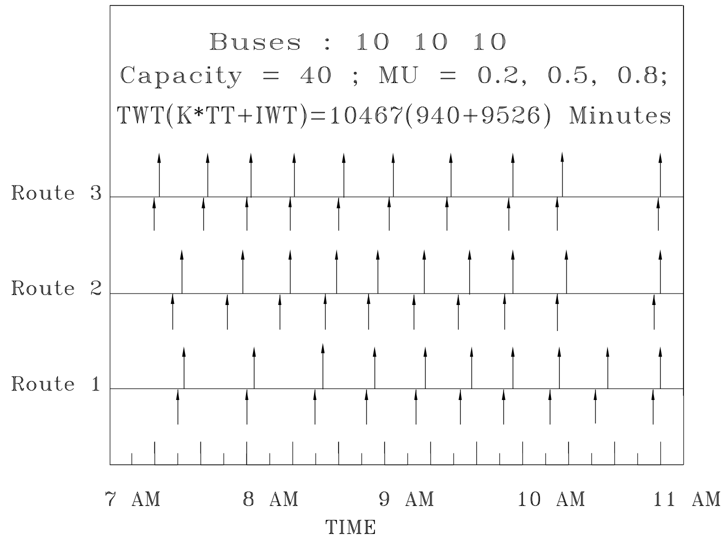

Figure 8: Optimized schedule with TWT minimization for different arrival patterns of passengers.

adherence of schedules. In this case study, we consider that buses arrive at a transfer station stochastically with a predefined distribution function and formulate a stochastic programming problem. We first formulate the optimization problem and then present simulation results of GAs.

6.3.1 Stochastic considerations

All scheduling parameters such as arrival time, departure time, transfer time, initial waiting time and total waiting time are stochastic. However, stochasticity in all these parameters arise due to the stochasticities in the arrival time only. It may be noted that in the subsequent discussion, a variable represented using bold characters refer to the scheduled value for the variable, the corresponding non-bold character refers to the stochastic quantity for the same variable. This notation is used only for this case study.

Arrival time: Considering that the scheduled arrival time is $\mathbf{a}_i^k$, we assume that the buses arrive at the transfer station with a probability density function $f_{a(j,l)}$. Any reasonable continuous density function like normal, exponential, gamma distribution may be used for $f_{a(j,l)}$.

Departure time: The departure time depends on the arrival time. If a bus arrives on or before the scheduled arrival time (that is, $a_j^l \le \mathbf{a}_j^l$), then the bus has to wait till its scheduled departure time $\mathbf{d}_j^l$. On the other hand, if a bus arrives after the scheduled arrival time, then the bus has to wait for a time equal to the scheduled stopping time and hence it cannot depart at the scheduled departure time. This dependency of the departure time on the arrival time makes the former also stochastic:

$$f_{d(j,l)} = \begin{cases} f_{a(j,l)}, & \text{if } d_j^l \ (= a_j^l + \mathbf{s}_j^l) > \mathbf{d}_j^l; \\ \delta(d_j^l - \mathbf{d}_j^l) \int_{a_j^l \le \mathbf{d}_j^l - \mathbf{s}_j^l} f_{a(j,l)} da_j^l, & \text{otherwise} \end{cases} \quad (3)$$

where δ is the Dirac Delta function. Thus, the probability, $P(d_j^l = \mathbf{d}_j^l) = \int_{a_j^l \le \mathbf{d}_j^l - \mathbf{s}_j^l} f_{a(j,l)} da_j^l$ and $P(d_j^l > \mathbf{d}_j^l) = \int_{a_j^l > \mathbf{d}_j^l - \mathbf{s}_j^l} f_{a(j,l)} da_j^l$.

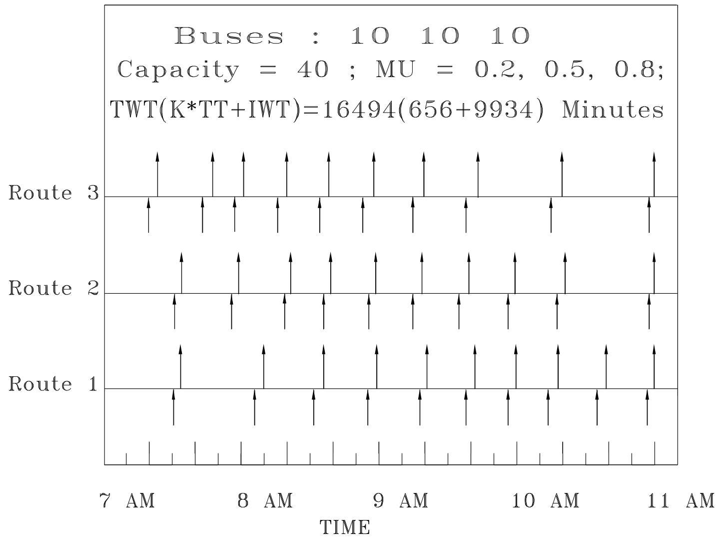

Figure 9: Optimized schedule with TWT minimization for different arrival patterns of passengers and large weightage to TT.

Initial waiting time (IWT): The initial waiting time is dependent on the departure times of two consecutive buses on a route. Hence, it is also stochastic. The probability distribution function of IWT, $f_{\varsigma(j,l)}$, between $(l-1)$-th and l-th buses of route j can be written as follows:

$$f_{\varsigma(j,l)} = \int_{d_j^l} f_{d(j,l)}(d_j^l) f_{d(j,l-1)}(d_j^l - h_j^l) dd_j^l, \tag{4}$$

where h_j^l (the headway between $(l-1)$-th and l-th bus of the j-th route) and ς_j^l (the actual IWT for the l-th bus of the j-th route) are related as follows:

$$\varsigma_j^l = \int_0^{h_j^l} v_{j,l}(t)(h_j^l - t)dt. \tag{5}$$

In equation 4, $v_{j,l}(t)$ is the arrival pattern of non-transferring passengers for the l-th bus on the j-th route and t denotes the time from the departure time of $(l-1)$-th bus.

Note that ς_j^l and ς_j^{l+1} are not independent, as both of them are dependent on the random variable d_j^l. Thus, one cannot obtain the distribution for total IWT ($\varsigma = \sum_j \sum_l \varsigma_j^l$), f_ς, through simple extension of the above procedure.

Transfer time (TT): Since departure and arrival times are stochastic, the transfer time is also stochastic. The probability distribution, $f_{\zeta(j,k,l,m)}$, for transfer time (ζ_{jk}^{lm}) from the l-th bus of j-th route to m-th bus of k-th route is as follows:

$$f_{\zeta(j,k,l,m)} = \mathrm{Prob}(\delta_{jk}^{lm} = 1) \int_{a_j^l} f_{a(j,l)} f_{d(k,m)}(a_j^l + \zeta_{jk}^{lm}) da_j^l. \tag{6}$$

This expression is the product of the probability that a transfer takes place from the l-th bus of j-th route to the m-th bus of k-th route and the probability that the difference between the departure time of the latter bus and the arrival time of the former bus is ζ_{jk}^{lm}. The definition of the binary variable δ_{jk}^{lm} makes it difficult to obtain the first term in the above expression. As earlier, we cannot simply extend the above procedure to obtain the distribution f_ζ for the total transfer time, ζ, because there exists a dependency between the δ_{jk}^{lm} terms.

Total Waiting Time (TWT): It is clear from the above that it is difficult to obtain the distribution for ζ and ς. The probability distribution, f_Γ, of the total waiting time (Γ), which is a function of ζ and ς is further difficult. The difficulty arises not only due to the difficulties in obtaining f_ζ and f_ς, but also due to the fact that ζ and ς are not independent variables.

6.3.2 Optimality criterion

The TWT cannot be used as an optimality criterion in a transit system where strict adherence to a given schedule is not possible, because the TWT computed based on a schedule is meaningful only if the buses arrive at and depart from the station as per the schedule. Hence, we have to redefine the optimality criterion realizing that the TWT in this case is a stochastic quantity. In the following, we describe two different optimality criteria.

Minimize Mean TWT: As discussed earlier, for a given schedule we have a probability distribution for TWT, f_Γ. Since mean is a measure of the central tendency of a distribution, we can claim that a schedule which offers a lower mean TWT is, on an average, better than a schedule which offers a higher mean TWT. Therefore, one of the optimality criterion would be to minimize the mean TWT (given by, $\int_\Gamma \Gamma f_\Gamma d\Gamma$). Other possibilities include minimizing variance of TWT or mean^2 TWT + variance of TWT.

Maximize Reliability: Another good measure of system performance in the case of stochastic systems is its reliability. We define the reliability, R, of a schedule as the probability that the TWT, Γ, is less than or equal to a permissible limit, Γ_ℓ:

$$R = \int_{\Gamma \leq \Gamma_\ell} f_\Gamma d\Gamma. \tag{7}$$

6.3.3 Classical solution techniques and their difficulties

From the above discussions, it is clear that obtaining a functional description of the objective function as well as constraints in this stochastic case are extremely difficult, if not impossible. In any case, the above problem is a stochastic, nonlinear, mixed-integer programming problem (S-NLMIP). Classical techniques for solving such problems involve making unnecessary assumptions about the problem like linearization, knowledge of probability distribution arising in the problem, and others (Rao, 1984; Taha, 1989). Given the difficulties experienced while trying to solve the NLMIP problem (arising in Case 1) using classical techniques, we believe that an attempt at solving the present problem using classical methods will be futile. Again, GA's ability to use procedure-based declarations comes useful in making the problem tractable. Here, we present how GAs are used to handle the S-NLMIP problem.

6.3.4 GA formulation

The purpose of obtaining the optimal schedule under stochastic arrival time conditions is to obtain a schedule which will be the "best" even when the schedule is not adhered to precisely (since a_i^k's are random). Thus, in practice, if one has to determine the best amongst many feasible schedules, the following procedure can be adopted:

1. Create many realistic situations (referred to as "instances") for each feasible schedule by perturbing the arrival times in the schedules (and therefore the departure times) using random numbers (that is, a realistic situation for a feasible schedule consists of the perturbed arrival and the consequent departure times, which may be thought of as the actual arrival and departure times of the buses on a given day),

2. Calculate the TWT for each of these situations for each schedule, and

3. Compare the TWT (or their means or any other measure defined on TWT's).

One could then claim that the schedule which outperformed all other schedules is the "best" schedule. This procedure is akin to making decisions about a stochastic system by performing statistical experiments on it through stochastic simulation. It is interesting to note that the above procedure eliminates the need for obtaining analytical descriptions of all the probability distributions (except that of a_j^l) discussed above.

Since genetic algorithms allow external procedure based declarations during the optimization process one could simulate the above process (using a procedure as shown below) and use the information on the comparisons in the optimization process. Notice that a GA string represents the scheduled headway h_j^l and stopping time s_j^l, which are deterministic variables.

```
Procedure Objective(a,d):
   for a feasible schedule (known s_j^l and h_j^l)
        obtain a_j^l and d_j^l
        make m copies of the schedule
        for each copy (u=1 to m) of the schedule
            generate a set of random numbers r_j^l using a given distribution
            calculate a_j^l = a_j^l + r_j^l
            calculate d_j^l:
                      if a_j^l + s_j^l <= d_j^l then d_j^l = d_j^l
                      else d_j^l = a_j^l + s_j^l
            calculate δ_{i,j}^{k,l} using Procedure Delta(a, d)
            calculate TWT (Γ_u) for u-th copy of the schedule
        calculate combined fitness of the schedule using f(Γ_u) (u = 1 to m)
```

In the case of mean TWT, $f(\Gamma_u)$ is $\sum_{u=1}^{m} \Gamma_u$ and for reliability objective $f(\Gamma_u)$ is as follows:

$$f(\Gamma_u) = 1 - \frac{\sum_{u=1}^{m} H(\Gamma_u - \Gamma_\ell)}{m},$$

where $H(\cdot)$ is the Heaviside function:

$$H(x) = \begin{cases} 1, & \text{if } x > 0; \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

6.3.5 Simulation results

The scheduling problem parameters and the GA parameters are same as in Case 1, except that a population size of 450 is used here.

We first present the simulation results for the **mean TWT** objective. The number of copies, m, considered for each schedule is 55. The following subcases which differ in the assumptions on the distributions of r_j^l are considered:

1. $r_j^l \sim N(0,2)$ (that is, r_j^l follows a normal distribution with mean zero and variance 2)
2. $r_j^l \sim N(0,4)$ 3. $r_j^l \sim N(1,2)$ 4. $r_j^l \sim N(2,2)$
5. $r_j^l \sim E(2)$ (that is, r_j^l follows a negative exponential distribution with mean 2)

Instead of presenting the schedules, here we present comparisons of the performance of the optimized schedules obtained in this case (Approach S) with those obtained in Case 1 (Approach D).

Table 1: Comparison of TWTs from Approaches S and D

SC	AP	Ins1	Ins2	Ins3	Ins4	Ins5	Ins6	Ins7	Ins8	Ins9	Ins10
1	S	5124	4921	5422	5102	5154	5034	5095	5316	5076	5109
1	D	5172	4860	5464	5216	5204	4965	5047	5328	5128	5219
2	S	5608	5241	5935	5624	5598	5351	5534	6098	5568	5792
2	D	5640	5210	6008	5733	5750	5616	5808	5983	5796	5901
3	S	5409	5054	5642	5304	5374	5223	5313	5542	5264	5396
3	D	5480	5048	5752	5395	5372	5243	5441	5581	5445	5562
4	S	5624	5313	5873	5611	5560	5339	5606	5758	5514	5615
4	D	5861	5451	6157	5847	5797	5620	5748	6041	5740	6033
5	S	5160	5184	4970	5184	5084	5192	5237	5013	5063	5022
5	D	5138	5214	4992	5223	5089	5210	5253	4993	5092	5040

SC: Subcase, AP: Approach, Ins$i$: Instance i

In order to compare the two approaches, the following procedure is used. For each subcase and for each of the two schedules (one obtained using Approach S and the other using Approach D), ten different instances (of which five were not in the set of 55 instances used during optimization) are simulated using the corresponding r_j^l distributions. The resulting TWTs for each of the two schedules are then compared in Table 1. It can be observed from the table that in most instances the TWT obtained from Approach S is lesser than that obtained from Approach D.

Next, the **reliability** of a schedule is used as the objective function. Here, $m = 100$ is used. Although various r_j^l distributions were tested, we only present the results for $r_j^l \sim N(0, 2)$. Since the choice of threshold TWT (Γ_ℓ) affects the reliability of a schedule, we have considered various threshold values while comparing the schedules. The results are presented in Figure 10.

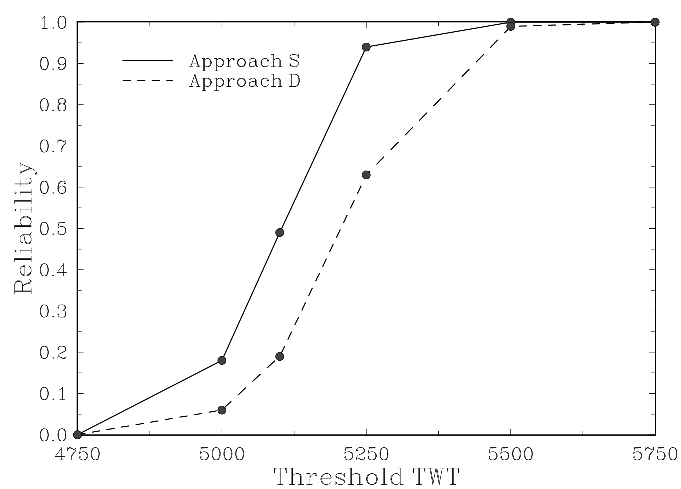

Figure 10: Comparison of reliability of Approaches S and D

It can be seen from the figure that the reliabilities of schedules from Approach S are always better than those obtained from Approach D. Obviously for very low and high threshold values of TWT, the reliabilities of schedules from both the approaches are zero and one, respectively.

The difference in the performance of the schedules becomes prominent for intermediate threshold values.

6.4 Case 4: Multiple stations, exact adherence, unlimited capacity

So far, we have considered only one transfer station. Thus, in the above cases, constraints G7 were inconsequential. However, in this case, these constraints also have to be incorporated while determining the scheduled arrival and departure times. Further, as will be discussed later, obtaining TT requires some special consideration. Since all procedures other than (i) obtaining arrival and departure times, and (ii) obtaining TT, are the same as in Case 1, we only describe these two procedures.

6.4.1 Obtaining arrival and departure times

Once the headways (h^l_{ij}, the l-th headway for j-th route at i-th station) and the stopping times (s^l_{ij}, the l-th stopping time for j-th route at i-th station) are computed for a string, the corresponding arrival and departure times can be calculated using a procedure described later. First, the following should be noted. In a network, at each station i, there are two types of routes: (i) independent routes (I_i), which go through only one transfer station, and (ii) common routes (C_i), which go through at least two transfer stations. To identify routes which belong to sets I_i and C_i at each station i, the following procedure is used. First, a station A is selected at random. All routes going through the station are included in set I_A. Thus, $C_A = \emptyset$. Next, another station $A + j$ is selected. For each route going through this station, we check whether the route is included in any of the sets I_A through I_{A+j-1}. If so, that route is placed in C_{A+j}, else it is placed in I_{A+j}. Once the sets I and C are determined for each station, the arrival and departure times are calculated as follows:

```
for all stations (i = 1 to b) compute
        for all routes (j = 1 to r_i) compute
            if j ∈ I_i; /* independent route */ a^0_ij = 0;
               for all transit buses (l = 1 to n_ij) calculate
                       a^l_ij = a^(l-1)_ij + h^l_ij;  d^l_ij = a^l_ij + s^l_ij;
            else /* common route, j ∈ C_i */
               for all transit buses (l = 1 to n_ij) calculate
                       a^l_ij = a^l_(i-1)j + α;  d^l_ij = a^l_ij + s^l_ij;
```

The parameter α mirrors the dependency of arrival times of buses on the common routes at two different transfer stations. As is clear, this procedure eliminates Constraints G7.

6.4.2 Obtaining TT

The value of total transfer time for a particular schedule is obtained by evaluating the first term of the objective function in equation 1. However, although not apparent, there exists a difficulty in obtaining TT for all buses. We discuss this matter and its remedy by using a procedure called Dependency of Arrival Time Algorithm (DATA) in the following.

The purpose of DATA is to overcome the problem in trying to obtain the TT when the arrival pattern of buses on one or more routes at a station gets fixed once the arrival times of the buses on the same routes are chosen at another station. Thus, if the scheduling time window at one station is $[p, q]$, the travel time between this and the next transfer station is c and the stopping time of the buses on the common route is d, then the time window over which buses of the common route arrive at the next station will be $[p + (c + d), q + (c + d)]$. If at the next station the scheduling time window is again $[p, q]$, obviously the buses of the common route will not be able to meet it. A little thought into the above fact reveals that this means undefined transfer times for certain buses as well as artificially magnified transfer times for other buses:

1. **Artificially magnified transfer time:** Passengers from the first few buses of an isolated route (for example, Route R2 in Figure 1) at a station will have to wait very long just because the first bus of the connecting route (Route R3, for example) cannot arrive before $p + (c + d)$ minutes.

2. **Undefined transfer time:** Passengers from the last few buses of the connecting route cannot be allowed to transfer to the other routes (Route R2, for example), because the latter have no more buses.

These problems are artificial because in reality, there are buses plying on the routes even outside the scheduling time window being considered. DATA is a procedure-based declaration which utilizes this fact to solve the problem of undefined and artificially magnified transfer times.

The basic assumption DATA makes is that the scheduling time window is fixed such that the demand pattern on all scheduling time windows are the same. That is, DATA requires that the demand pattern in scheduling time windows $[p^{n-1}, q^{n-1}]$ and $[p^{n+1}, q^{n+1}]$ be the same as that in the window $[p^n, q^n]$, where $[p^n, q^n]$ is the time window for which scheduling is being done and $n - 1$ and $n + 1$ in the superscript refer to the earlier and latter time windows, respectively. This assumption is justifiable since there is no restriction on the extent of the time window; it could be 1 hour long or it could even be 24 hours long. The assumption then leads to the fact that a schedule which is optimal for $[p^n, q^n]$ is optimal for all other scheduling time windows (such as $[p^{n-1}, q^{n-1}]$). Therefore, the schedules obtained for one period could be repeated for other periods. This then will represent arrivals of buses at a station much more realistically (as opposed to buses arriving only during one scheduling time period) and thereby solve the problems of undefined as well as unrealistically large transfer times. Figure 11 illustrates this fact. The

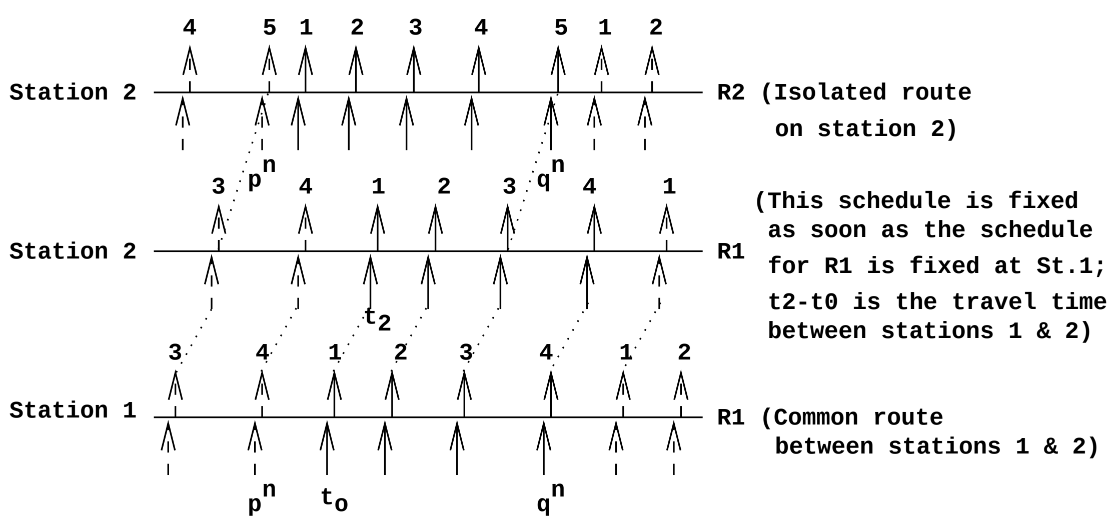

Figure 11: Illustration of DATA.

solid arrows are the arrival and departure times of the buses which appear in the set of decision variables or equivalently buses which appear within $[p^n, q^n]$. The vertical dashed arrows refer to actual arrivals and departures of buses on all other scheduling time windows. In the figure, the dotted line between departure time of a bus in Route 1 (R1) on Station 1 to arrival time of the bus in Route 1 on Station 2 signifies the travel of the same bus on Route 1 between Stations 1 and 2.

The above figure illustrates how the strategy adopted in DATA overcomes the problems associated with transfer times. For example, the first bus of the isolated route (R2) at Station 2 can transfer to the 4-th bus of the common route (R1) and vice versa. This avoids the following: (i) *Undefined transfer time* for passengers transferring from the 4-th bus of R1 to R2 at Station 2, and (ii) *Artificially magnified transfer time* for passengers transferring from the 1-st bus of R2 to R1 at Station 2.

It should be noted that while presenting the results, the meaning of the solid and dashed vertical arrows and the dotted lines are the same as those mentioned here.

6.4.3 Simulation results

The network considered here consists of three transfer stations: S1, S2, and S3. Routes R1, R2, and R3 go through station S1; routes R1, R4, and R5 go through station S2; and routes R5, R6, and R2 go through station S3; The travel time from station S1 to S2, station S1 to S3, and station S2 to S3 are 30, 40, and 50 minutes, respectively. The GA parameters used are same as before except the following: (i) Population size = 1000, (ii) Mutation probability = 0.002, (iii) String length = 288.

Figure 12 shows the best schedule obtained using the GA-based procedure. The objective function is TWT and the number of buses on each route is 10. The figure shows that the headways are uniform for each route at all the stations, as expected.

7 Conclusions

In this paper, we have formulated a transit system scheduling problem (determining optimal arrival and departure times of buses) into a mixed-integer nonlinear programming (MINLP) problem. The MINLP problem involves minimizing the total waiting time (TWT) of all passengers which is a sum of the initial waiting time (IWT) of non-transferring passengers and transfer time (TT) of transferring passengers. The MINLP problem also involves a number of resource and service related constraints such as fleet size, minimum and maximum stopping time and headway, and others.

Genetic algorithms (GAs) are particularly chosen to solve the transit scheduling problem because the classical optimization techniques had difficulties in solving the problem. Difficulties arise because of discrete and complex search space having nonlinear constraints, and a large number of integer and real decision variables. Most of these difficulties can be avoided by using simple procedure-based declarations. GAs provide a framework in which such procedure-based declarations can be easily handled. Further, the binary string coding mechanism allowed in GAs eliminates a number of constraints and provides a natural way to handle binary decision variables.

The efficacy of GA-based approach is shown by applying the proposed procedure to different types of transit scheduling problems—limited versus unlimited bus capacity, deterministic versus stochastic arrival time, and single versus multiple transfer stations. It may be noted that the MP formulation may not be possible to write in most of the transit scheduling problems studied here.

The results presented here are for equal number of buses on each route. This is done to show that the *same* GA-based approach is able to find optimal/near-optimal schedules in *all* these cases where the optimal solutions are reasonably known a priori. Obviously, the above procedure can also be used for unequal number of buses on each route (Agrawal, 1997; Reddy, 1996; Srinivas, 1995; Subrahmanyam, 1995). These results are not provided here for brevity.

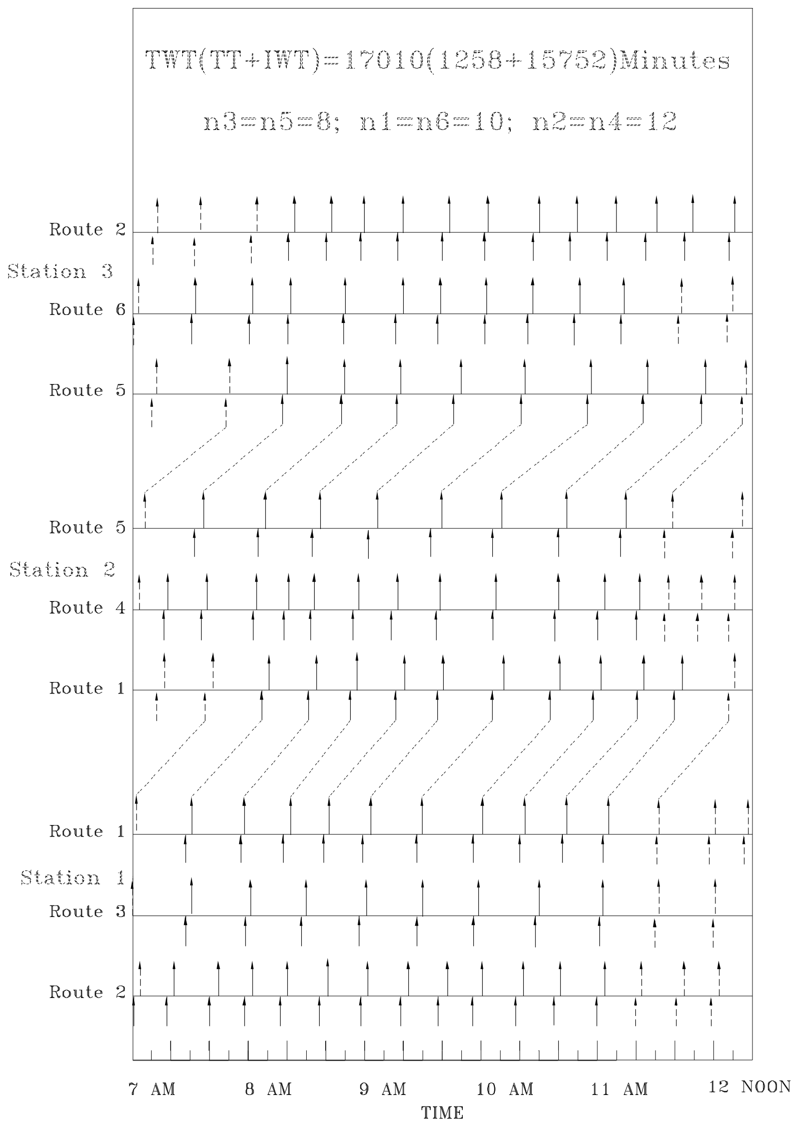

Figure 12: Optimized schedule obtained using GA-based approach for multiple transfer stations.

Acknowledgments

The authors acknowledge the help of a number of their students: Pulugurtha Subrahmanyam, Bandaru Srinivas, Y. S. Reddy, Ravi Agrawal, and Rajeev Mehrotra.

References

Agrawal, R. (1997). Transit system scheduling with limited vehicle capacity. (Master's thesis). Kanpur: Department of Civil Engineering, Indian Institute of Technology.

Bookbinder J.H. and Désilets A. (1992). Transfer optimization in a transit network, *Transportation science. 26*(2), 106–118.

Chakroborty, P., Deb, K., and Srinivas, B. (in press). Network-wide scheduling of bus transit systems using genetic algorithms. *Microcomputers in Civil Engineering.*

Chakroborty P, Deb K. and Subrahmanyam P.S. (1995). Optimal Scheduling of Urban Transit Systems using Genetic Algorithms, *ASCE Journal of Transportation Engineering*, 121(6), 544-553.

Deb, K. (1995) *Optimization for Engineering Design: Algorithms and Examples*, Prentice-Hall, New Delhi.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning.* Reading, MA: Addison-Wesley.

Holland, J. H. (1975). *Adaptation in natural and artificial systems.* Ann Arbor: University of Michigan Press.

Kikuchi, S., and Parameswaran, J. (1993). Solving a schedule coordination problem using a fuzzy control technique. Presented at *Intelligent Scheduling Systems Symposium* at the ORSA-TIMS conference. San Francisco, CA.

Martinelli, D. and Teng, H. (1995). A genetic algorithm approach for solving the train formulation problem. *Transportation Research Record, 1497.* 62–69.

Radcliffe, N. J. (1991). Formal analysis and random respectful recombination. In R. K. Belew and L. B. Booker (Eds.), *Proceedings of the Fourth International Conference on Genetic Algorithms* (pp. 222–229).

Rao, S. S. (1984). *Optimization theory and applications.* New Delhi: Wiley Eastern.

Reddy, Y. S. (1996). Transit system scheduling under stochastic arrival times (Master's thesis). Kanpur: Department of Civil Engineering, Indian Institute of Technology.

Reklaitis, G. V., Ravindran, A., and Ragsdell, K. M. (1983). *Engineering optimization—Methods and applications.* New York: Wiley.

Srinivas, B. (1995). Network-wide transit system scheduling using genetic algorithms. *Masters Thesis.* Kanpur: Department of Civil Engineering, Indian Institute of Technology.

Subrahmanyam, P. S. (1995). Transit system scheduling: Use of genetic algorithms. *Masters Thesis.* Kanpur: Department of Civil Engineering, Indian Institute of Technology.

Taha, H. (1989). *Operations Research.* New York: Macmillan.

Wren, A and Wren, D. O. (1995). A genetic algorithm for public transport driver scheduling. *Computers and Operations Research, 22*, 101–110.