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
deb@iitk.ernet.in

Partha Chakroborty
Department of Civil Engineering
Indian Institute of Technology
Kanpur, Pin 208 016, India
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 the objective of minimizing the overall waiting time of transferring and nontransferring 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 they (i) naturally handle binary variables, thereby taking care of transfer decision variables, which constitute the majority of the decision variables in the transit scheduling problem; and (ii) 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 that do not arrive exactly as per scheduled times, and a multiple-station transit system having common routes among bus stations. Simulation results show the 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 levels of service to passengers, public transport systems must be efficient. A major factor that 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 of 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 & Désilets, 1992; Kikuchi & 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 they allow 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 & Teng, 1995; Wren & 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 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 for solving 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 various routes that 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 that will eventually take them to their destinations. Thus, at any transfer station, there are two types of passengers (customers): transferring passengers and nontransferring passengers (those whose station of origin is the transfer station itself).

A typical transit system operating on a network of streets is shown in Figure 1. Consider the transfer station S2. 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; and (ii) nontransferring 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 the transit system design problem is an optimization problem having an objective and a number of constraints that we describe in the next section.

Because there are two types of passengers, there are two types of waiting times—initial waiting time (IWT) (the waiting time of nontransferring passengers) and transfer time (TT) (the waiting time of transferring passengers). Both IWT and TT depend on the arrival and departure times of buses on all routes. The objective function, total waiting time (TWT), is the sum of IWT and TT for all passengers.
Figure 1. A typical transit system network. Light lines represent the streets; bold lines represent the routes; and circles represent the transfer stations.

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 on 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:

- $a_{ij}^r$: Arrival time of the $r$th bus on the $j$th route at the $i$th station.
- $d_{ik}^m$: Departure time of the $m$th bus on 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.
- $q_{ij}^{\min}$: Minimum stopping time of a bus on the $j$th route at the $i$th station.
- $q_{ij}^{\max}$: Maximum stopping time of a bus on the $j$th route at the $i$th station.
- $t_{(i-1)j}^r$: Travel time of the $r$th bus on the $j$th route from the $(i-1)$th station to the $i$th station.
- $T$: Maximum transfer time.
- $v_{ij}(t)$: Arrival pattern of passengers for the $r$th bus on the $j$th route at the $i$th station.
Transfer volume from the $i$th bus on the $j$th route to the $k$th route at the $i$th station.

$\delta_{ijk}^m$: A binary variable that is 1 if a transfer from the $i$th bus on the $j$th route to the $m$th bus on the $k$th route at the $i$th station is possible and optimal; zero otherwise (see also 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 on schedule. This is referred to as exact (strict) schedule adherence. When the arrival time of buses deviates from the scheduled arrival time, this is referred to as stochastic schedule adherence.

2. It is also assumed that the available capacity of a bus arriving at a station is sufficient 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 the limited bus capacity condition.

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

Minimize

\[
\sum_{i} \sum_{j} \sum_{k} \sum_{m} \delta_{ijk}^m (d_i^m - d_j^m) \delta_{ijk}^l + \sum_{i} \sum_{j} \sum_{k} \int_0^{d_i^m - d_j^m} \psi_i(t)(d_j^m - d_j^{m-1} - t) \, dt
\]

Subject to

\begin{align*}
G1 &\equiv d_i^j - d_j^j \leq \epsilon_{ijn}, & \forall i, j, n \\
G2 &\equiv d_i^j - d_j^j \geq \epsilon_{ijn}, & \forall i, j, n \\
G3 &\equiv d_i^j - d_j^{j-1} \leq h_{ij}, & \forall i, j, l \\
G4 &\equiv (d_i^m - d_j^m) \delta_{ijk}^m \leq T, & \forall i, j, k, l, m \\
G5 &\equiv d_i^m - d_j^m + M(1 - \delta_{ijk}^m) \geq 0, & \forall i, j, k, l, m \\
G6 &\equiv \sum_{k} \delta_{ijk}^m = 1, & \forall i, j, k, l, \\
G7 &\equiv d_i^j - d_{i-1}^j = t_{i-1}^{j-1}, & \forall j, i, i \geq 2
\end{align*}

The decision variables in the formulation given in Equation 1 are the arrival times $d_i^j$, the departure times $d_j^j$, and $\delta_{ijk}^m$ values. The variable $\delta_{ijk}^m$ is a zero-one integer variable. A value of zero means that the transfer from the $i$th bus on the $j$th route to the $m$th bus on the $k$th route at the $i$th transfer station is either not possible (i.e., the $i$th bus on the $j$th route arrives after the $m$th bus on the $k$th route has departed) or nonoptimal to the passengers (i.e., there is at least one bus on the $k$th route that departs after the arrival of the $i$th bus on the $j$th route and before the departure of the $m$th bus on 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 (TT) for all the transferring passengers at all transfer stations and the second term the total initial waiting time (IWT) for all the passengers at their station of origin.
The transfer time for each of the $w_{ijk}$ passengers who want to transfer from the $l$th bus on the $j$th route to the $k$th route at the $i$th station is

$$(a_{jk}^{lm} - d_{ij})v_{ijk}$$

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

The second term is arrived at by assuming that passengers boarding the $l$th bus on the $j$th route at the $i$th station arrive according to some function $v_{ij}(t)$ (where $t$ is measured from $d_{ij}$) between $d_{ij}$ and $d_{ij}^{-1}$. The integral in the second term is therefore the total IWT for all passengers boarding the $l$th bus on the $j$th route at the $i$th station. This term when summed over all buses on 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 $(d_{ij} - a_{ij})$ 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 $s_{ij}^{\text{max}}$, and constraints G2 state that the stopping time $(d_{ij}^{-1} - a_{ij})$ should be greater than or equal to the minimum stopping time $s_{ij}^{\text{min}}$. 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 TT for any transfer to be less than or equal to the maximum TT $T$.

Constraints G5 through G7 are logic constraints that define the feasibility of transfers and dependency arrival times. They are explained below. Constraints G5 ensure that $\delta_{ijk}^{lm}$ is zero whenever a transfer from the $l$th bus on the $j$th route to the $m$th bus on the $k$th route at the $i$th station is not possible. Constraints G6 along with constraints G5 ensure that transfer from a particular bus on a particular route is made to only one of the buses on 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 on 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 until Section 6.2.

In the above formulation, it is assumed that the buses will arrive and depart strictly on schedule. In general, this is 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.
Characteristics of the MP Formulation

In this section, we discuss the characteristics that 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 \(a_i\), departure times \(d_j\); and transfer variables \(\delta_{ij}\). Among them, the \(\delta\) variables are binary, taking a 1 or a 0. This makes the search space discrete. Furthermore, since the other two types of variables represent scheduled arrival and departure times of buses, it is desirable that they be in minutes rather than in fractions of a minute. For example, a scheduled 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 be 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 \(\delta\) variables, the algorithm may require computation of the \(TT\) term for nonbinary values of \(\delta\) variables. This may result in intermediate solutions that 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. Furthermore, 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 \((b = 1)\), three 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 \(\delta\) 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 large, the number of constraints is also large. A little computation will reveal that there are also \(O(br^3n^2)\) constraints associated with the problem. For the above single-station network, there are a total of 1350 linear and nonlinear constraints.

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

Motivation for Using GAs

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 \(G5\) and \(G6\) 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 GAs naturally work in an environment where variables are always bounded, the bound constraints can be eliminated by using GAs. Furthermore, GAs allow external procedure-based declarations; that is, GAs can use information obtained from procedures that are external to the optimization algorithm. This feature of GAs allows us to eliminate constraints G5, G6, and G7 by using small procedures (these are discussed later). This feature also eliminates 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 be 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 GAs 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 on schedule; available bus capacity is unlimited.

Case 2: The network consists of one transfer station; buses are assumed to arrive and depart exactly on 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 on 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 1. 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 \(O(n^2)\).
In an earlier study (Chakroborty, Deb, & Subrahmanyam, 1995), we 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. In the following subsection, we 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 the GA. Specifically, we present the binary string representation of variables, the procedure-based declarations to take care of some of the constraints, and the procedure for computing the objective function.

Recall that the scheduling problem has three types of variables—arrival times \( a_j \), departure times \( d_j \), and transfer variables \( b_{ij} \). 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:

\[
\begin{align*}
x'_j &= (a'_j - a^{j-1}_j): & \text{the headway between the } l \text{th and } (l-1) \text{th bus on the } j \text{th route} \\
dx'_j &= (a'_j - d'_j): & \text{the stopping time of the } l \text{th bus on the } j \text{th route}
\end{align*}
\]

The arrival and departure times \((a'_j \text{ and } d'_j)\) can be computed from the above two types of variables with an initial condition \(d'_0 = 0\) and using the following recursive equations:

\[
d'_j = a^{j-1}_j + x'_j, \quad d'_j = a'_j + dx'_j
\]

We show later how the transfer variables \(b_{ij}\) 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.

On each route, there are as many arrival and departure times (or \(x'_j\) and \(dx'_j\)) as there are buses. However, to simplify the matter, we assume that all buses on a particular route have the same stopping time. Thus, \(d'_j = d_j\). Thus, for \(n_i\) buses on the \( j \)th route at a station, there are a total of \(n_i + 1\) variables. However, in order to restrict the schedule for a particular time window (say \( T \)), 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 \(n_i - 1\) buses, instead of all \(n_i\) buses. Thus, there are a total of \(n_i\) variables for each route at each station in the GA formulation. We represent each variable with binary substrings. Finally, we concatenate all \(n_i\) 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:

\[
((dx_1 x'_1 x^{n_1}_1 \ldots x^{n_1}_r) \ldots (dx_R x'_R x^{n_R}_1 \ldots x^{n_R}_r))
\]

Coding stopping time and headways of all buses for a particular route together help to propagate good partial schedules through GA operators, a matter 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:

\[ G1 \equiv x_j^l \leq b_j, \quad \forall j, l \]
\[ G2 \equiv d_j^l \leq \gamma_j^{\max}, \quad \forall j, l \]
\[ G3 \equiv d_j^l \geq \gamma_j^{\min}, \quad \forall j, l \]

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 \( \delta_{jk} \):

\[
\text{flag} = \text{false} \\
\text{for all combinations of routes } (j, k) \text{ and buses } (l, m) \\
\text{if } d_j^l < \delta_{jk} \text{ or flag } = \text{true} \text{ then} \\
\delta_{jk} = 0 \\
\text{else} \\
\delta_{jk} = 1 \\
\text{flag } = \text{true}
\]

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

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

where \( g \) is the left-hand side function value of the constraint represented as \( G \geq 0 \). A constant penalty parameter \( R \) of \( 10^4 \) is used. The number of problem variables and constraints in the above GA formulation are \( 2m \) and \( Y(T - 1)n^2 \), respectively. The reductions in the number of variables (from quadratic to linear in \( r \) and \( n \)) and in the number of constraints \([3r(r - 1)n^2 + 3m \text{ to } r(r - 1)m^2] \) 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 is a total of 30 buses (10 on each route) available to ply in three routes. We choose the scheduling time window \( (T) \) to be from 7 to 11 AM (240 minutes). The following parameters are chosen for the scheduling problem:

- Minimum headway, \( b_j^{\min} = 14 \text{ minutes} \), except for the first bus, where \( b_j^{\min} = 0 \text{ minutes} \).
- Maximum headway, \( b_j^{\max} = 45 \text{ minutes} \), except for the first bus, where \( b_j^{\max} = 3 \text{ 1 minutes} \).
- Minimum stopping time \( \gamma_j^{\min} = 2 \text{ minutes} \).
- Maximum stopping time \( \gamma_j^{\max} = 5 \text{ minutes} \).
- Maximum TT \( T = 30 \text{ minutes} \).

Using the above variable bounds, let us compute the total string length to represent a complete schedule. For the \( j \)th route, headways \( (x_j^l, l = 1, 2, \ldots, 9) \) for nine buses and one stopping time \( (d_j^l) \) 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 four 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 trial-and-error experiments):

- Population size is 350.
- Binary tournament selection is used.
- Single-point crossover on the complete string is used.
- Crossover probability is $p_c = 0.95$.
- Bitwise 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 the objective of minimizing the transfer time (TT) only. In this case, the optimal schedule will be the one where buses on different routes arrive at and depart from the transfer station at approximately the same time. Figure 2 shows the best schedule obtained by the 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 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.

Next, we investigate the effect of initial waiting time (IWT) alone on the optimal schedule. We choose the arrival pattern of passengers ($c_i(t)$) to be triangular, with a height $I_i$ of 1.244, as shown in Figure 3. In this case, we expect the optimal schedule to have each headway of a route be equal, because this problem is similar to a multivariable optimization problem of minimizing $\sum_i x_i^2$ subject to $\sum_i x_i = \text{constant}$ (note that the 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.
GAs in Transit System Scheduling

Figure 3. The triangular arrival pattern of passengers.

Figure 4. Optimized schedule for IWT obtained by GAs. The arrival pattern of passengers is triangular with height $L_1 = L_2 = L_3$.

To investigate the effect of both TT and IWT on the optimal schedule, we consider next the objective function with the TT ($w_T = 1$) and IWT terms. The optimized schedule is shown in Figure 5. It is interesting to note that the optimal schedule for the 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 values, in the optimal solution for only IWT case and TWT case, are the same and equal to 4,477 minutes. Hence in this case the optimal schedule should also have equal headways on each route.

The above simulation results show how easily GAs can be used to find optimized schedules. Similar performances of GAs are also observed in many other test cases, including unequal buses on each route and nonuniform 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 nontransferring passengers) who arrive to board a particular bus may not be able to do so. Therefore, it becomes imperative that we maintain a queue of arriving passengers for each route. Nontransferring 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 until the arrival of a bus on that route. If the available capacity of this bus (say, the \( k \)th bus on the \( j \)th route) is \( C_j^k \), then, at this time, the first \( C_j^k \) passengers from the queue board the bus. The rest form the initial queue for the next bus on the same route. This process continues. Note that the IWT and TT will have to be computed by summing the times spent by individual nontransferring 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 solutions 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 the arrival and departure time of each bus on each route. Given the arrival pattern of nontransferring passengers \((\alpha'_j(t))\) and transferring passengers \((\omega'_j(t))\) and the arrival and departure times of each bus, the queuing process described above is implemented through a procedure. Using this procedure, the IWT and TT terms are computed. Note that the procedure for determining \( \delta_{jk}^{\text{lim}} \) is no longer necessary, because the \( \delta_{jk}^{\text{lim}} \) variables as defined earlier are no longer meaningful.

In the GA formulation described in Case 1, we penalized a schedule that gives rise to
6.2.2 Simulation Results  To illustrate the sensitivity of an 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 the IWT term, and the arrival pattern of the passengers is such that more passengers arrive in the latter half than in the former half of the scheduling time window. It is also assumed that the arrival pattern of passengers on all the routes is the same. GA parameters are kept the same as before, except that a population size of 600 and a maximum generation of 1200 are used. Figure 6 shows the optimized schedule for the case described above [with \( \mu = 0.2 \) ("mu" in Figure 6) for all routes, thereby exhibiting a peak locus of maximum passenger arrival at \((1 - 0.2) \times 240\), or 192 minutes after 7AM\(^2\) and an available bus capacity of 35. As can be seen, more buses arrive between 120 and 240 minutes than between 0 and 120 minutes. The total number of passengers arriving for a route is approximately 320, whereas the total available bus capacity on each route is \(35 \times 10\), 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 toward the optimal schedule corresponding to unlimited

---

1. The locus of the vertices of the inter-bus arrival patterns of passengers (e.g., \( \Lambda \) in Fig. 3) is assumed to follow the following function: \( 3.7(\tau/(1 - \mu))^{1 - \mu}(1 - \tau)/\mu^\mu \), where \( \tau \in [0, 1] \) is the nondimensionalized time across the scheduling time window, and the maximum of the above function occurs at \( \tau = 1 - \mu \).

2. However, it is important to note that for two consecutive buses on a route with departure times \( d^j_{n-1} \) and \( d^j_n \), the peak of passenger arrival always remains at \( 0.75(d^j_{n-1} - (d^j_{n-1}) \) from \( d^j_{n-1} \) (refer to Fig. 3).
bus capacity. Alternatively, one should expect that as bus capacity increases, the IWT per passenger for an optimized schedule should decrease. Figure 7 shows this fact.

In the other case, the TWT is considered 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, the arrival pattern is such that more passengers arrive in the former half (we choose $\mu = 0.8$). The total number of nontransferring passengers is approximately the same as before, and the total number of transferring passengers is approximately 100. Figure 8 shows the optimized schedule. As expected, the buses are distributed according to the arrival pattern of passengers. In order to see the effect of TT on the above optimized schedule, we weight TT more heavily in the objective function ($K = 10$ times more than IWT compared to $K = 1$ used in Fig. 8) and rerun the GAs. Figure 9 shows that the average TT per transferring passenger (total TT divided by the total number of transferring passengers) has decreased from 9.40 (Fig. 8) to 6.36 minutes (Fig. 9). The figure also indicates that there are more transfers that require minimal transfer times (i.e., buses on the different routes arrive more or less at the same time) than in the previous case (Fig. 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 on 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 adherence to 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.
GAs in Transit System Scheduling

Figure 8. Optimized schedule with TWT minimization for different arrival patterns of passengers. MU = μ in this figure.

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

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 arises due to the stochasticities in the arrival time only. It may be seen that in the subsequent discussion, a variable shown in boldface refers to the scheduled value for the variable, and the corresponding non-boldface variable 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 $a_i$, we assume that the buses arrive at the transfer station with a probability density function $f_{a_i(j,p)}$. Any reasonable continuous density function like normal, exponential, and gamma distribution may be used for $f_{a_i(j,p)}$. 

Evolutionary Computation Volume 6, Number 1
**Departure time:** The departure time depends on the arrival time. If a bus arrives on or before the scheduled arrival time (i.e., $d_j^i \leq d_j^i$), then the bus has to wait until its scheduled departure time, $d_j^i$. 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:

$$f_{a,j,l} = \begin{cases} f_{a,j,l}(d_j^i - d_j^i) & \text{if } d_j^i (= d_j^i + r_j) > d_j^i \\ \delta(d_j^i - d_j^i) \int_{d_j^i - d_j^i} f_{a,j,l} \, dd_j^i & \text{otherwise} \end{cases}$$

where $\delta$ is the Dirac delta function. Thus, the probability

$$P(d_j^i = d_j^i) = \int_{d_j^i - d_j^i} f_{a,j,l} \, dd_j^i$$

and

$$P(d_j^i > d_j^i) = \int_{d_j^i > d_j^i} f_{a,j,l} \, dd_j^i$$

**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, $f_{i,j,l}$, between the $(l-1)$th and $l$th buses on route $j$ can be written as follows:

$$f_{i,j,l} = \int_{d_j^i} f_{a,j,l}(d_j^i - b_j^{l-1})(d_j^i - b_j^l) \, dd_j^i$$

where $b_j^l$ [the headway between the $(l-1)$th and $l$th buses on the $j$th route] and $\zeta_j^l$ (the actual IWT for the $l$th bus on the $j$th route) are related as follows:

$$\zeta_j^l = \int_0^{b_j^l} v_{j,l}(t)(b_j - t) \, dt$$

In Equation 5, $v_{j,l}(t)$ is the arrival pattern of nontransferring passengers for the $l$th bus on the $j$th route, and $t$ denotes the time from the departure time of the $(l-1)$th bus.

Note that $\zeta_j^l$ and $\zeta_j^{l+1}$ are not independent because both are dependent on the random variable $d_j^i$. Thus, one cannot obtain the distribution for total IWT ($\zeta_j^m = \sum_l \sum_i \zeta_j^l$), $f_{i,j,l}$, 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, $f_{t,j,k,l,m}$ for TT $(\zeta_{jk}^m)$ from the $l$th bus on the $j$th route to the $m$th bus on the $k$th route is as follows:

$$f_{t,j,k,l,m} = \text{Prob}(\zeta_{jk}^m = 1) \int_{d_j^i} f_{a,j,k}(d_j^i - \zeta_{jk}^m) \, dd_j^i$$
This expression is the product of the probability that a transfer takes place from the \( l \)th bus on the \( j \)th route to the \( m \)th bus on the \( k \)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 \( \delta_{jk} \). The definition of the binary variable \( \delta_{jk} \) 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 \( f_z \) for the total TT \( \zeta \) because there exists a dependency between the \( \delta_{jk} \) terms.

**Total Waiting Time (TWT):** It is clear from the above that it is difficult to obtain the distribution for \( \zeta \) and \( \zeta \). The probability distribution \( f_z \) of the TWT \( \Gamma \), which is a function of \( \zeta \) and \( \zeta \), is even more difficult. The difficulty arises not only due to the difficulties in obtaining \( f_z \) and \( f_z \), but also to the fact that \( \zeta \) and \( \zeta \) 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 on 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 \( f_z \) for TWT. Since the mean is a measure of the central tendency of a distribution, we can claim that a schedule that offers a lower mean TWT is, on average, better than a schedule that offers a higher mean TWT. Therefore, one optimality criterion would be to minimize the mean TWT (given by \( \int \Gamma f_z d\Gamma \)). Other possibilities include minimizing the 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 \( \Gamma \) is less than or equal to a permissible limit \( \Gamma_r \):

\[
R = \int_{\Gamma < \Gamma_r} f_z d\Gamma
\]  

### 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, such as linearization and knowledge of probability distribution arising in the problem, among 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, the GAs ability to use procedure-based declarations becomes 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 that will be the "best" even when the schedule
is not adhered to precisely (since $a_i^j$'s are random). Thus, in practice, if one has to determine
the best among 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 (i.e., 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 that 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 $a_i^j$) discussed above.

Since genetic algorithms allow external procedure-based declarations during the opti-
mization process, one could simulate the above process (using a procedure as shown below)
and use the information on the comparisons in the optimization process. Note that a GA
string represents the scheduled headway $h_i^j$ and stopping time $s_i^j$, which are deterministic
variables.

Procedure Objective($a_i^j$, $d_i^j$):

for a feasible schedule (known $s_i^j$ and $h_i^j$)

obtain $a_i^j$ and $d_i^j$ 

make $m$ copies of the schedule

for each copy ($u=1$ to $m$) of the schedule

generate a set of random numbers $r_i^j$ using a given

distribution

calculate $a_i^{j'} = a_i^j + r_i^j$

calculate $d_i^{j'}$:

if $a_i^{j'} + s_i^j \leq d_i^j$ then $d_i^{j'} = d_i^j$
else $d_i^{j'} = a_i^{j'} + s_i^j$

calculate $\delta_i^{j',j}$ using Procedure Delta($a_i$, $d_i$)

calculate TWT ($\Gamma_u$) for $u$th copy of the schedule

calculate combined fitness of the schedule using $f(\Gamma_u)$

($u = 1$ 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:

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

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

$$H(x) = \begin{cases} 
1 & \text{if } x > 0 \\
0 & \text{otherwise} 
\end{cases}$$
Table 1. Comparison of TWTs from Approaches S and D.

<table>
<thead>
<tr>
<th>SC</th>
<th>AP</th>
<th>Ins1</th>
<th>Ins2</th>
<th>Ins3</th>
<th>Ins4</th>
<th>Ins5</th>
<th>Ins6</th>
<th>Ins7</th>
<th>Ins8</th>
<th>Ins9</th>
<th>Ins10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>5124</td>
<td>4921</td>
<td>5422</td>
<td>5102</td>
<td>5154</td>
<td>5034</td>
<td>5095</td>
<td>5316</td>
<td>5076</td>
<td>5109</td>
</tr>
<tr>
<td>1</td>
<td>D</td>
<td>5172</td>
<td>4860</td>
<td>5464</td>
<td>5216</td>
<td>5204</td>
<td>4965</td>
<td>5047</td>
<td>5328</td>
<td>5128</td>
<td>5219</td>
</tr>
<tr>
<td>2</td>
<td>S</td>
<td>5608</td>
<td>5241</td>
<td>5935</td>
<td>5624</td>
<td>5598</td>
<td>5351</td>
<td>5534</td>
<td>6098</td>
<td>5568</td>
<td>5792</td>
</tr>
<tr>
<td>2</td>
<td>D</td>
<td>5640</td>
<td>5210</td>
<td>6008</td>
<td>5733</td>
<td>5750</td>
<td>5616</td>
<td>5808</td>
<td>5983</td>
<td>5796</td>
<td>5901</td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>5409</td>
<td>5054</td>
<td>5642</td>
<td>5304</td>
<td>5374</td>
<td>5223</td>
<td>5313</td>
<td>5542</td>
<td>5264</td>
<td>5396</td>
</tr>
<tr>
<td>3</td>
<td>D</td>
<td>5480</td>
<td>5048</td>
<td>5752</td>
<td>5395</td>
<td>5372</td>
<td>5243</td>
<td>5441</td>
<td>5581</td>
<td>5445</td>
<td>5622</td>
</tr>
<tr>
<td>4</td>
<td>S</td>
<td>5624</td>
<td>5313</td>
<td>5873</td>
<td>5611</td>
<td>5560</td>
<td>5339</td>
<td>5606</td>
<td>5758</td>
<td>5514</td>
<td>5615</td>
</tr>
<tr>
<td>4</td>
<td>D</td>
<td>5861</td>
<td>5451</td>
<td>6157</td>
<td>5847</td>
<td>5797</td>
<td>5620</td>
<td>5748</td>
<td>6041</td>
<td>5740</td>
<td>6033</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>5160</td>
<td>5184</td>
<td>4970</td>
<td>5184</td>
<td>5084</td>
<td>5192</td>
<td>5237</td>
<td>5013</td>
<td>5063</td>
<td>5022</td>
</tr>
<tr>
<td>5</td>
<td>D</td>
<td>5138</td>
<td>5214</td>
<td>4992</td>
<td>5223</td>
<td>5089</td>
<td>5210</td>
<td>5253</td>
<td>4993</td>
<td>5092</td>
<td>5040</td>
</tr>
</tbody>
</table>

SC: subcase; AP: approach; Ins: instance $i$.  

6.3.5 Simulation Results  The scheduling problem parameters and the GA parameters are the 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 that differ in the assumptions on the distributions of $r_f$ are considered:

1. $r_f \sim N(0,2)$ (i.e., $r_f$ follows a normal distribution with mean zero and variance 2)
2. $r_f \sim N(0,4)$
3. $r_f \sim N(1,2)$
4. $r_f \sim N(2,2)$
5. $r_f \sim E(2)$ (i.e., $r_f$ 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). 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), 10 different instances (of which 5 were not in the set of 55 instances used during optimization) are simulated using the corresponding $r_f$ 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 less 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 $r_f$ distributions were tested, we only present the results for $r_f \sim N(0,2)$. Since the choice of threshold TWT ($\Gamma_r$) affects the reliability of a schedule, we have considered various threshold values while comparing the schedules. The results are presented in Figure 10.

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 thresh-
old 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. Furthermore, 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 \( h_{ij} \), the \( l \)th headway for the \( j \)th route at the \( i \)th station \) and the stopping times \( s_{ij} \), the \( l \)th stopping time for the \( j \)th route at the \( i \)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 that 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 \( I_i \) 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:

\[
\text{for all stations } (i=1 \text{ to } b) \text{ compute } \notag \\
\text{for all routes } (j=1 \text{ to } n_i) \text{ compute } \notag \\
\quad \text{if } j \in I_i; \text{ /* independent route */ } a_{ij}^0 = 0; \notag \\
\quad \text{for all transit buses } (l=1 \text{ to } n_y) \text{ calculate } \notag 
\]
The parameter $\alpha$ 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 TT 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 the 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
Kalyanmoy Deb and Partha Chakroborty

Figure 11. Illustration of DATA.

Figure 11 illustrates this fact. The solid vertical arrows are the arrival and departure times of the buses that appear in the set of decision variables or, equivalently, buses that appear within \([p^*, q^*]\). The dashed vertical arrows refer to actual arrivals and departures of buses on all other scheduling time windows. In the figure, the dotted line between the departure time of a bus on Route 1 (R1) at Station 1 and the arrival time of the bus on Route 1 at Station 2 signifies the travel time 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 fourth bus of the common route (R1), and vice versa. This avoids the following: (i) undefined \(TT\) for passengers transferring from the fourth bus of R1 to R2 at Station 2, and (ii) artificially magnified \(TT\) for passengers transferring from the first 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 times 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 the same as before, except for the following: (i) population size = 1000; (ii) mutation probability = 0.002; and (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 the sum of the initial waiting time (IWT) of nontransferring passengers...
and the 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, among others.

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. Furthermore, 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 the 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 an 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 an unequal number of buses on each route (Agrawal, 1997; Reddy, 1996; Srinivas, 1995; Subrahmanyam, 1995). These results are not provided here for brevity.

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


