Model for simultaneous routing and scheduling using genetic alghoritm

P.J. Gundaliya

Lecturer (Civil Engineering)
L.D. College of Engineering, Ahmadabad, Gujrat, India

Prabhat Shrivastava

Assistant Professor (Civil Engineering)
Sardar Patel College of Engineering
Andheri (W), Mumbai, India

P.L. Dhingra

Professor (Transportation System Engineering)
Civil Engineering Department, Indian Institute of Technology,
Mumbai, India

The design of bus routes and setting frequencies on these routes are two basic decision elements that critically determine public transport system performance. Various attempts are made to solve this type of combinatorial optimisation problem involving non- linearity, non- convexity with multiple objective functions but in most of the approaches design of routes and schedules are dealt separately though they are complementary to each other. In this study a model is developed for simultaneous routing and scheduling using a robust optimization technique namely Genetic Algorithm (GA). For this model objective function is minimisation of the sum of user and operator costs. User cost is taken as combination of in-vehicle travel time, waiting time and transfer time where as operator cost is vehicle operating cost of buses. Constraints are related to load factor, fleet size and overloading of links. The model is tested for Mandl's Swiss Transit Network and Demand Matrix. It is found that the developed model gives the better-optimised values over other existing results for the same network and demand matrix.

1. Introduction

In the metropolitan cities of developing countries like India due to heavy growth of vehicular population specially private and intermediate transport services and limited carriageway capacity has affected mobility of individuals. As a result traffic congestion, reduced travel speed, poor level of service and environmental pollution are prevailing. Under such worse conditions it is required to increase the efficiency of public transport system so that use of intermediate transport be reduced to a lesser degree and private transport is discouraged. Therefore it is required to plan and design urban public transit in the most efficient manner with existing constraints.

Usage of mass public transportation in four major metropolitan cities of India is above 70% in which the share of bus transport is dominant. Thus public transportation has become an integral and essential feature of cities especially in developing countries. In populous and developing countries like India, where urbanisation trend is on the increase, mass transit systems like commuter rails and bus transportation are mostly unavoidable.

Among various means of public transportation, the bus transport has dominated because of its door to door accessibility and flexibility in operation. The efficiency of bus transport system depends on routes and schedules. In past various attempts are made to design routes and frequencies. In most of the approaches development of routes and selection of frequencies of buses is done separately to avoid complications and computational burden. In actual practice both go hand to hand. If routing and scheduling go together the routes generated will support the defined schedule. Hasselstorm (1981) and Marwah et al (1984) used a complex two level optimization model, which first reduces the network by eliminating links that are seldom or never used by passengers. A large set of possible routes is then generated from the remaining links. Finally the network routes are selected by assigning frequencies using a linear programming model which maximizes the number of transfers saved by changing from a link network to a public transport network. In the proposed study single optimization model is developed, which finds optimal routes and schedules together and optimizes the objective function, which is the minimisation of user and operator's costs with the real life constraints. Thus from literature review reveals that limited attempts are made for simultaneous design of routes and schedules. In other words the developed routes should support a set of schedules.

2. Model for simultaneous routing and scheduling

In this research work a model is developed for simultaneous routing and scheduling of public buses. The model determines optimal fleet size for optimally designed routes and corresponding set of frequencies of public buses. User cost is taken as total travel time, which includes in-vehicle travel time, waiting time and the transfer time. The operator cost is taken as running time cost of the buses. The objective function is minimisation of summation of both costs. Constraint for load factor is taken for better level of service and to ensure a minimum number of passengers for economical operation. Fleet size constraint is adopted to keep the number of buses under certain limit and link-overloading constraint ensures that there can not be buses over certain specified limit on any link. The mathematical formulation of the model is as follows.

Objective Function

$$\text{Minimize}\left\{ C_1\times\left[\sum_{j=1}\sum_{i=1} d_{ij}\times t_{invttij}\right]+C_2\times\left[\sum_{j=1}\sum_{i=1} d_{ij}\times t_{wt.ij}\right]+C_3\times\left[\sum_{j=1}\sum_{i=1} d_{ij}\times t_{tt.ij}\right]+C_4\times\left[\sum_{all k\in SR} f_K\times T_K\right]\right\}$$

Subject to
Load factor constraints:

$$LF_K=\frac{(Q_K)_{\max}}{f_K\times CAP}\leq LF_{max} \qquad \text{for all } k\in SR$$

$$LF_K=\frac{(Q_K)_{\max}}{f_K\times CAP}\leq LF_{min} \qquad \text{for all } k\in SR$$

Fleet size constraint:

$$\sum_{all\, K\in SR} N_k=\left[\sum_{all\, K\in SR} f_k \times T_k\right]\leq W$$

Link overloading constraint:

$$fij\leq F\max$$

Where,
dij = demand between nodes i and j
CAP = Seating Capacity of buses operating on the network's routes
SR = Set of transit routes
C1 = In - vehicle time cost per minute,
C2 = Waiting time cost per minute
C3 = Transfer time cost per minute
C4 = Bus operating cost per minute

2.1 *Steps Involved for Model Development*

Step: 1
Inputs are identified for the model like:

- Number of nodes in the network and their connectivity along with travel time details between nodes.
- Demand matrix for above network.
- Design parameters like bus seating capacity, maximum and minimum load factors, fleet size etc.

Step 2:
Node pairs are identified and sorted with the following criterions

- As per production & attraction of nodes - The nodes having more than above average production and attraction are selected and paths connecting these nodes are considered for further scrutiny.
- Route Length - From the above the paths having length between 15 minutes to 33 minutes are considered.
- Average Route Flow Values (ARFV) - After route length criteria ARFV a criterion is applied and paths having more than Average Route Flow Value are selected. Thus nodes present on the above paths are selected for further analysis.

Step 3:
K-Shortest Paths are generated between above identified node pairs.

Step 4:
Assignment on Routes - After identifying the direct node pairs each alternate is considered for assigning the load to the link, based on exponential function of frequency and distance. Thus for example as in Fig. 1, if there are three acceptable routes R_1, R_2 and R_3 between nodes i and j and frequencies are f_1, f_2, f_3 and distances d_1, d_2, d_3 respectively. Load on each route can be calculated as follows :

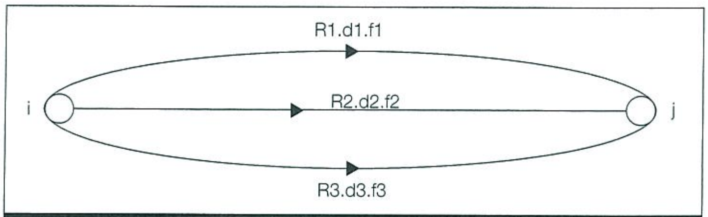

Fig. 1: example network

$$\text{Route R1 demand}=\left(\frac{e^{v1}}{e^{v1}+e^{v2}+e^{v3}}\right)\times dij$$

Where

$$v1=\{ff\times f_1 - df\times d_1\}$$

ff = Frequency distribution factor (percentage of trip makers prefer frequency based choice
df = Distance distribution factor (percentage of trip makers prefer the shorter length Criterion as compare to frequency.
dij = Demand between node i and j
n = Number of route pass between node i and j
f_1, f_2, f_3= Frequency of route R_1, R_2, and R_3 resperctively
d_1,d_2,d_3= lenghts of route R_1, R_2, and R_3 resperctively

Thus load is assigned to links on the route. This flow incurs average waiting time (in minutes)

$$=\left\{\frac{60.0}{2.0\,(f_1+f_2+f_3)}\right\}$$

This means that for any of the alternative the waiting time incurred is the same as the average waiting time. This rule is based on the assumption that arriving trip makers' board the first bus, which arrives at the stop. The waiting time calculation further ignores stochastic nature in bus headway. Further from all the available alternative routes by one transfer, shortest route is chosen using the distance criteria. The demand (dij) is assigned on the links of the path followed by the transfer. The waiting time at transfer point ($t_{wt.ij}$) is calculated based on frequencies on routes before and after transfer.
Thus total travel time is calculated as

$$t_{ij}=t_{invttR1}+t_{invttR2}+t_{wt.ij}+\text{Transfer penalty}$$

Where,
$t_{invttR1}$=In - vehicle time from node i to transfer point through R_1 route

$t_{invttR2}$ = In - vehicle time from transfer point to node j through R_2 route

Step 5:
Application Of Genetic Algorithm - Genetic Algorithm is applied for different alternatives using random frequencies. Finally a set of optimal routes and schedules are selected.

Step 6:
Computation of number of buses - Required number of buses are calculated using following formula

2.2 *Application of Genetic Algorithm for Model Development*

2.2.1 *Overview of Genetic Algorithm*

Genetic Algorithms (GAs) are computerised search and optimization algorithms based upon the principles of Darwinian evolution. The concept of the survival of the fittest is used in a structured, yet randomised information exchanges to form a robust search algorithm. Genetic algorithms efficiently exploit historical information to locate search points with improved performance. The basic idea behind GAs is to generate an initial pool of solutions, represented as string structures, and then thorough continuous copying, swapping, and modifying of partial strings in a manner similar to natural genetic evolution, to allow the solution pool to evolve towards better and better solutions.

Working Principle

The working principle of GAs is illustrated in the form of pseudocode as follows:

```
begin
Initialise population of strings
Compute fitness of population
Repeat:
        Reproduction,
        Crossover,
        Mutation,
        Compute fitness of population
Until termination criterion is achieved
end
```

Coding

An important characteristic of genetic algorithms is the coding of variables that describe the problem. The most common coding method is binary coding (Goldberg 1989). The decision variables are usually mapped and represented by a string (chromosome) of binary alphabets (genes). The length of the string is usually determined according to the desired solution accuracy. Once the coding of variables has been done, the corresponding point can be found using a fixed mapping rule, usually the following linear mapping rule is used (Goldberg 1989).

$$X_i = X_{i.min} + \frac{X_{i.max} - X_{i.min}}{2^{l_i} - 1} * decoded\_value\ (S_i)$$

Where,

$X_{i.min}$ = is lower bound on decision variable X_i

$X_{i.max}$ = is upper bound on decision variable Xi

The variable X_i is coded in a substring S_i of length l_i. Decoded value (S_i) is $\sum_{i=0}^{l-1} 2^i S_i$ where $S_i \in (0,1)$ and the string is represented as (S_{l-1}, S_{l-2}, ...S_2,S_1,S_0). When all the decision variables are decoded using above mapping rule, the function value can also be calculated by substituting the variables in the given objective function f(x). The obtainable accuracy of the variable for a l_i-bit coding is

$$\frac{X_{i.max} - X_{i.min}}{2^{l_i} - 1}$$

Fitness function

Since GAs mimic the survival of the fittest principal of nature to make a search process. Therefore, GAs are naturally suitable for solving maximisation problems. Minimisation problems are usually transformed into maximisation problems by some suitable transformation. After all the values of variables are obtained, they can be used to calculate the objective function value. In general, a fitness function F (x) is first derived from the objective function and used in successive genetic operations. For maximisation problems, the fitness function can be considered to be the same as objective function i.e. F(x) = f(x). For minimisation problems the fitness function is an equivalent maximisation problem chosen such that the optimum points remains unchanged. The following fitness function is usually used (Deb 1995).

$$F(x) = \frac{1}{1 + f(x)}$$

This transformation does not alter the location of minima but converts a minimisation problem to an equivalent maximisation problem. The fitness function value of a string is known as the string's *fitness*.

The operation of GAs begins with population of random strings representing design of decision variables. Thereafter, each string is evaluated to find the fitness value. Three main operators then operate the population: reproduction, crossover and mutation to create a new population of points. The new population is further iteratively operated by above three operators and evaluated. This procedure is continued and tested for termination. One cycle of these operations and subsequent evaluation procedure is known as a *generation*.

GA Operators

Reproduction (or selection) is an operator that makes more copies of better strings in a population. The commonly used reproduction operator is the proportionate reproduction operator where a string is selected for mating pool with a probability proportional to its fitness. Thus, the i^{th} string in the population is selected with a probability proportional to F_i. The probability for selecting the i^{th} string is

$$P_i = \frac{F_i}{\sum_{j=1}^{n} F_j}$$

Where n is population size and F_i is fitness of i^{th} string.

After the reproduction phase is over, the population is enriched with good strings. Reproduction makes clones of good strings, but does not create any new string. A *crossover* operator is used to recombine two strings with a hope of creating a better string. In the crossover operator, exchanging information among strings of mating pool creates new strings. In most crossover operators, two strings are picked from the mating pool at random and some portion of the strings is exchanged between the strings. A single-point crossover operator is performed by randomly choosing a crossing site along the string and by exchanging all bits on the right side of the crossing site as shown in fig 2.

```
0 0 : 0 0 0         0 0 : 1 1 1
              =>
1 1 : 1 1 1         1 1 : 0 0 0
```

Fig. 2.0: crossover operation

The two strings participating in the crossover operation are known as parent strings and the resulting strings are known as children strings. It is intuitive from this construction that good sub strings from parent strings can be combined to form a better child string, if an appropriate site is chosen. Since the knowledge of an appropriate site is usually not known beforehand, a random site is often chosen. Reproduction operator takes this random site selection, because if good strings are created by crossover, there will be more copies of them in the next mating pool, otherwise they will not survive beyond next generation. The total number of strings participating in the mating pool can be controlled by specifying crossover probability, p_c. Another operator, *mutation* is used sparingly. Under the action of this operator this operator, a 1 will change to a 0 and vice versa with a small probability. Mutation also creates a new string, but its effect is considered secondary. It introduces diversity in the population whenever the population tends to become homogeneous due to iterative use of selection and crossover operators. Furthermore, for local improvement of a solution, mutation may be found useful. Since this operator disrupts a string, the probability of mutation p_m is kept very low.

Termination Criterion

When the average fitness of all strings in a population is equal to the best fitness, the population is said to have converged. When population is converged, the GA is terminated. The same can be done by fixing maximum number of generations, the number of generations at which population will converge. In GAs maximum number of generations is generally used as the termination criteria.

2.2.2 Implementation of Genetic Algorithm for Simultaneous Routing and Scheduling

Genetic Algorithm is found to be most suitable for this problem due to multi objective nature of objective function, large number of variables, non linearity involved in objective function and in constraints.

Coding

Routes and the frequencies of each pair are coded into a single string with the desired precision. For example Fig. 3 shows binary digits coding for route no. 5 and route no.3 with frequencies 6 and 21. First four bits show route and last six bits show corresponding frequency in a string.

Pair no 1										Pair no 2									
Route				Frequency						Route				Frequency					
0	1	0	1	0	0	0	1	1	0	0	0	1	1	0	1	0	1	0	1
Route no 5				Frequency 6						Route no 3				Frequency 21					

Fig. 3: binary digit coding

Fitness function: -

Fitness function is taken as minimisation of objective function i.e. summation of users and operator's costs. Constraints are considered by imposing penalties for their violation. These penalties are added in Objective function and thus penalised objective function is calculated. The value of penalised objective is used for optimisation. Weights to the penalties are given in terms of function of objective function value and as per their relative importance.

Therefore

Fitness function = Minimize (Objective function + penalties 1 to 6) Where,

Penalty 1 = penalty for the unsatisfied demand,

Penalty 2 = penalty for the one transfer demand satisfied,

Penalty 3 = penalty for the Minimum load factor,

Penalty 4 = penalty for the Maximum load factor,

Penalty 5 = penalty for the Fleet size,

Penalty 6 = penalty for the link over loading,

The program for fitness is developed in object-oriented environment in C++. Fig. 4 gives flow chart for implementation of GA for our problem.

3 Application of model for case study

The Model developed here is tested on the Mandl' s Swiss network of fifteen nodes. Mandl (1980) originally reported this benchmark network. The total demand is 15570 transit trips. The highest node pair demand being 880 transit trips. In this matrix, 82% of the node pairs have non-zero demand. The same network also used by Baaj et al (1990), Kidwai (1999) and Muralidhar (1999). The results of the proposed algorithm are compared with the previous researcher's results. The network is small and dense; it comprises only 15 nodes within a 33 minutes shortest travel time between the two farthest nodes. Although this network may not be very representative of many real-world urban bus transit networks, it is still useful possibly as regional sub network. Mandl's network is based on a real network in Switzerland and the demand matrix shows the

0	400	200	60	80	150	75	75	30	160	30	25	35	0	0
400	0	50	120	20	180	90	90	15	130	20	10	10	5	0
200	50	0	40	60	180	90	90	15	45	20	10	10	5	0
60	120	40	0	50	100	50	50	15	240	40	25	10	5	0
80	20	60	50	0	50	25	25	10	120	20	15	5	0	0
150	180	180	100	50	0	100	100	30	880	60	15	15	10	0
75	90	90	50	25	100	0	50	15	440	35	10	10	5	0
75	90	90	50	25	100	50	0	15	440	35	10	10	5	0
30	15	15	15	10	30	15	15	0	140	20	5	0	0	0
160	130	45	240	120	880	440	440	140	0	600	250	500	200	0
30	20	20	40	20	60	35	35	20	600	0	75	95	15	0
25	10	10	25	15	15	10	10	5	250	75	0	70	0	0
35	10	10	10	5	15	10	10	0	500	95	70	0	45	0
0	5	5	5	0	10	5	5	0	200	15	0	45	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 1: demand matrix for mandi's network

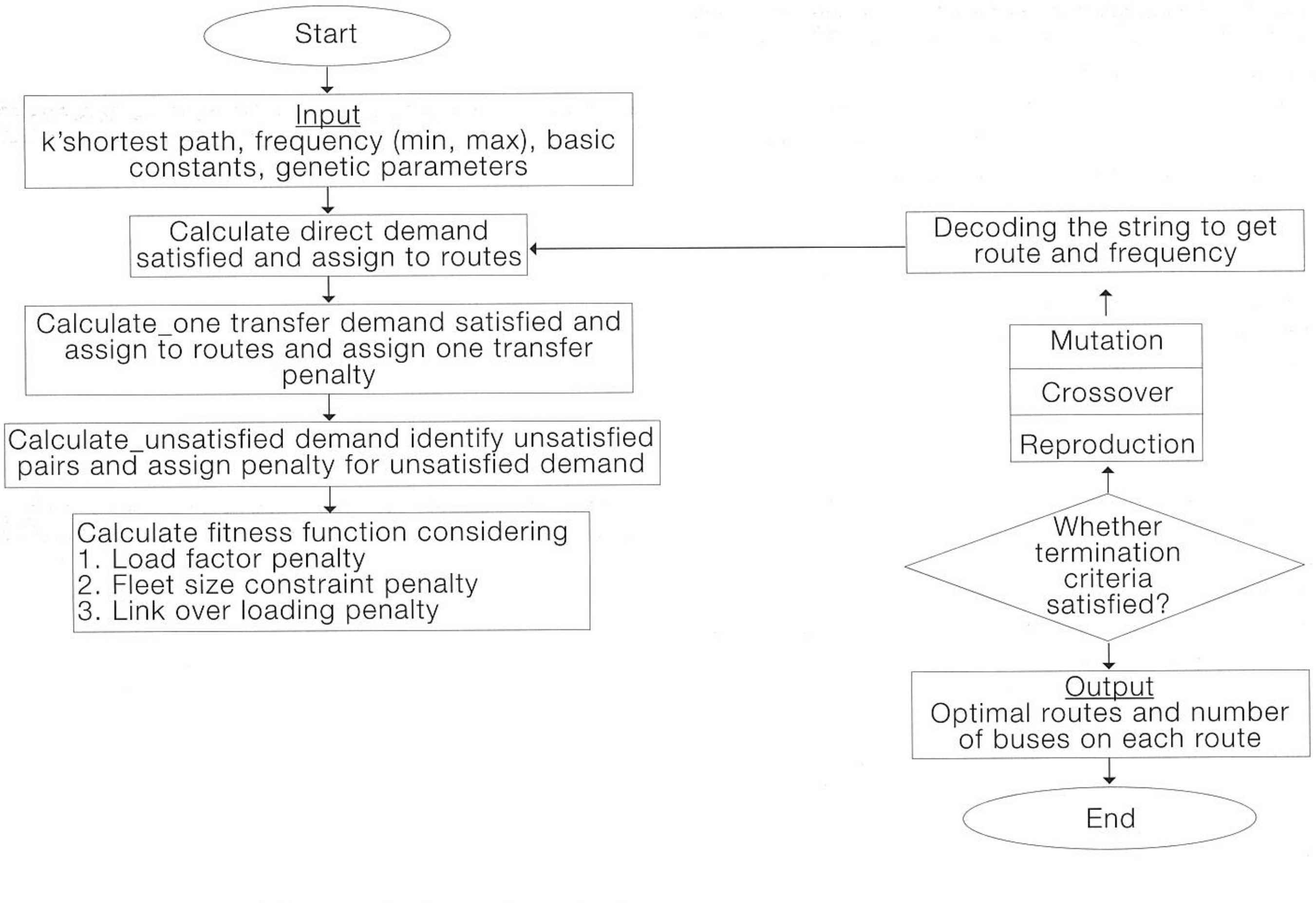

Figure 4: flow chart for developing optimal routes and number of buses on each route using Genetic Algorithm

average number of passengers per hours. The table 1 shows the demand matrix of the network and figure 5 shows the mandl' s network with time taken between each node in minutes.

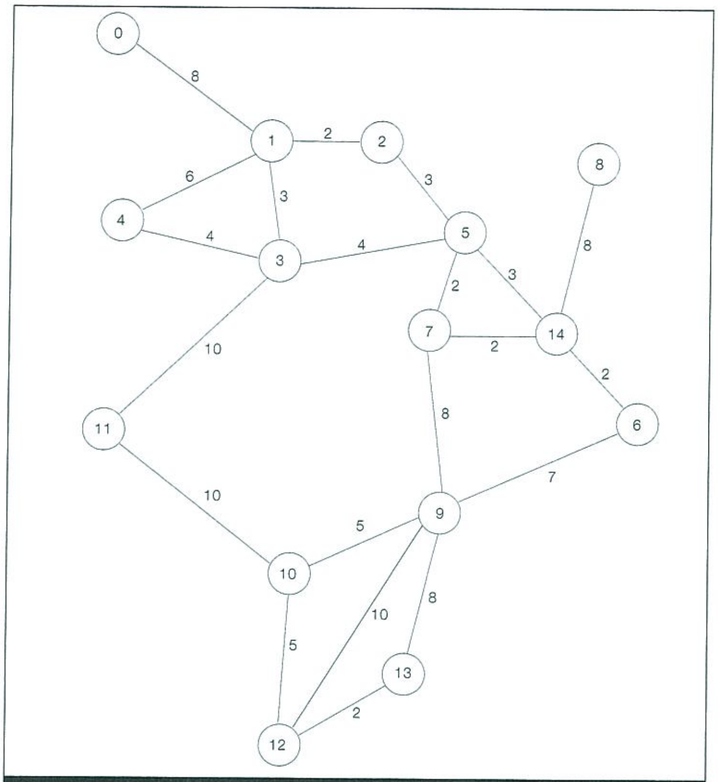

Fig. 3: binary digit coding

3.1 Values of different Parameters

For calculating the fitness function all the penalties are assigned to the objective function. The various penalties weights are decided after several runs and observing the effect of the penalty on the search of fitness value for the model. For present case following penalties are considered.

Unsatisfied demand = 0.8 times of objective function per trip

One transfer = 0.7 times of objective function per trip

More than one transfer = 0.8 times of objective function per trip

Over loaded link constraints = 0.01 times of objective function per link

Load Factor MAX constraints = 0.6 times of objective function per route

Load Factor MIN constraints = 0.6 times of objective function per route

Penalty for the Fleet Size constraints = 0.1 times of objective function .

Cost of in-vehicle time taken as Rs 0.22 per minute, cost of transfer Rs. 0.13 per minute and the cost of waiting time is adopted as Rs. 0.13 per minute (Draft Report of MMPG on Mumbai Metro Study, 1997). Running time cost of BEST buses is taken as Rs. 6.8 per minute (Monthly statistical review part-2 of August, 1999). An average speed of the bus 15 km/hr. is adopted for calculation.

4 Results and discussions

4.1 Selection of appropriate value of GA Parameters

The various genetic parameters are fixed for the fitness function by observing their effect on fitness value. Population size 40, Cross Over Probability 0.95, Mutation Probability 0.1 give better fitness values. Refer figs 6,7 and 8. It is observed that considering above parameters the GA converges within 100 generation refer Fig. 9. However maximum generation was allowed up to 500 for the present study.

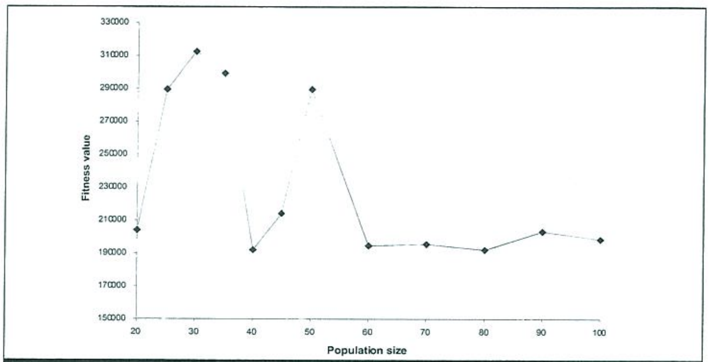

Fig. 6: effect of population size on fitness value

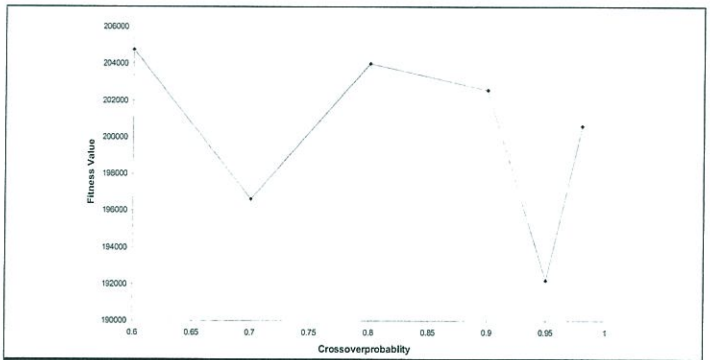

Fig. 7: effect of Crossover probability on fitness value

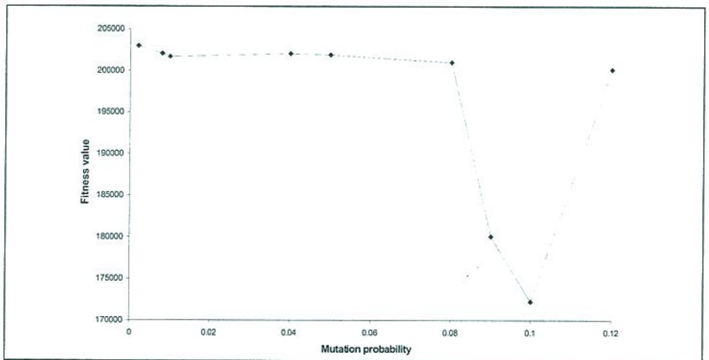

Fig. 8: effect of mutation probability on fitness value

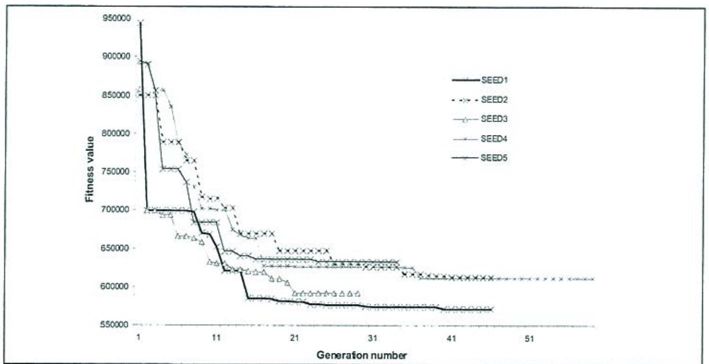

Fig. 9: effect of various seeds on generation number and fitness value

4.2 *Comparison of Results*

The results obtained by the model compared with the Mandl and Baaj (1980) solution as the same network is used for analysis. Developed Routes detail by Mandl and Bajj and from proposed model is given as follows. Table 2 shows that direct demand satisfied in all the cases are higher than the Mandl and Baaj's solution. The reason is that the previous one is heuristic approach where as in the present study the Genetic Algorithm is used to get the optimal solutions. It

Set of routes	%Demand satisfied through transfer				In-veh. Travel Time (min)	Transfer Time (min)	Waiting Time (min)	Total Travel Time (min)	Number Of Buses
	Zero	One	Two	Un satisfi ed					
1. Mandl I	68.78	31.22	0.0	0.0	211210	24330	14672	250212	128
2. Mandl II	69.85	29.97	0.17	0.0	177752	23590	17598	218940	111
3. Baaj I	78.3	21.69	0.0	0.0	169138	16970	20007	206115	103
4. Baaj II	79.65	20.35	0.0	0.0	166461	15915	26675	209051	93
5. Baaj III	80.88	19.12	0.0	0.0	180620	14910	22156	217686	100
6. Mandl II	72.95	26.91	0.13	0.0	177006	21070	21070	216469	114
7. Baaj I	77.92	19.62	2.3	0.0	166167	18690	18690	205220	110
8. Baaj II	85.67	14.32	0.0	0.0	161540	11150	11940	199000	92
9. Baaj III	85.54	14.45	0.0	0.0	162680	11250	12430	199911	84
RESULTS OF PROPOSED MODEL									
10. ARFV 300**	93.38	6.62	0.0	0.0	160521	5150	18154	183825	75
13. ARFV 400**	89.92	10.08	0.0	0.0	173394	7850	16446	197690	88
14. ARFV 500**	88.89	11.11	0.0	0.0	173616	8650	21049	203315	88

**Capacity of buses is 50
ARFV : Average Route Flow Value

Table 2: comparison of results with Mandl and Baaj's solution

shows that Genetic Algorithm gives the better results moreover the developed model find optimal routes and schedules simultaneously in contrast to previous approaches. It is also evident that the total travel time and the number of buses are considerably reduced. Number of buses required on each route for different ARFV is given in table 3. It is observed that increasing in ARFV the total travel time also increases as shown in table 4.

ARFV	Total travel time per person in minutes
ARFV 300	11.80
ARFV 400	12.70
ARFV 500	13.05

Table 4: relation of total travel time per person with ARFV

ARFV	Route No.	LENGTH (Min.)	TRIPS	MAXIMUM LINK FLOW	LOAD FACTOR	BUSES ON ROUTE	TOTAL BUSES
300	1	33	1530	500	1.11	10	75
	2	30	2431	658	1.2	12	
	3	28	2442	613	1.12	11	
	4	27	2370	655	1.19	10	
	5	19	3938	1072	0.98	14	
	6	25	1311	422	0.94	8	
	7	24	1548	604	1.00	10	
400	1	33	3038	853	0.95	19	88
	2	30	5145	1362	1.18	23	
	3	28	3175	994	1.1	16	
	4	27	3252	968	0.74	23	
	5	25	915	517	1.15	7	
500	1	33	7138	1320	1.2	28	88
	2	28	2688	795	0.88	20	
	3	29	585	189	0.75	6	
	4	22	3751	1132	0.87	24	
	5	25	1408	517	1.15	10	

Table 3: scheduling of buses with different arfv.

4.2.1 Comparisons of routing results

Mandl' s Solution: -

(a) Initial set of routes (before improvement)

RI: 10 - 12 - 13 - 9 - 7 - 14 - 5 - 2 - 1 - 0
RII: 6 - 14 - 5 - 3 - 4
RIII: 11 - 3 - 5 - 14 - 8

(b) Final set of routes (after improvement)

RI: 0 - 1 - 2 - 5 - 7 - 9 - 10
RII: 4 - 3 - 5 - 7 - 14 - 6
RIII: 11 - 3 - 5 - 14 - 8
RIV: 12 - 13 - 9

Baaj's Solution:-

(a) First set of routes generated by Baaj

RI: 6 - 14 - 7 - 9 - 10 - 11
RII: 6 - 14 - 5 - 7 - 9 - 13 - 12
RIII 0 - 1 - 2 - 5 - 7
RIV: 8 - 14 - 6 - 9
RV: 4 - 3 - 5 - 7 - 9
RVI: 0 - 1 - 2 - 5 - 14 - 8
Minimum Demand Satisfied directly = 50%

(b) Second set of routes generated by Baaj

RI: 0 - 1 - 3 - 11 - 10 - 13
RII: 2 - 5 - 7 -14 - 6 - 9
RIII: 9 - 10 - 12
RIV: 9 - 10 -11
RV: 7 - 9 - 13
RVI: 0 - 1 - 3 - 5 - 7
RVII: 8 - 14 - 5 - 7 - 9
RVIII: 4 - 1 - 2 - 5 - 14 - 6 - 9
Minimum Demand Satisfied directly = 50%

(C) Third routes generated by Baaj

RI: 9 - 12
RII: 9 -10 - 11
RIII: 9 - 13
RIV: 0 - 1 - 2 - 5 - 7 - 9
RV: 8 - 14 - 6 - 9
RVI: 4 - 3 - 5 - 7 - 9
RVII: 0 - 1 - 3 - 4
Minimum Demand Satisfied directly = 70%

Proposed algorithm solution

(a) First set of routes for ARFV (Average Route Flow Value) of 300

RI: 0 - 1 - 2 - 5 - 7 - 9 - 13 - 12
RII: 0 - 1 - 3 - 5 - 7 - 9 - 10
RIII: 4 - 3 - 5 - 14 - 6 - 9
RIV: 8 - 14 - 6 - 9 - 10
RV: 0 - 1 - 2 - 5 - 7 - 14 - 6
RVI: 9 - 13 -12 -10 -11

(b) Second set of routes for ARFV of 400

RI: 0 - 1 - 2 - 5 - 7 - 9 -13 -12
RII: 0 - 1 - 5 -14 - 6 - 9 - 10
RIII: 4 - 3 - 5 - 7 - 14 - 6 - 9 - 10
RIV: 8 - 14 - 5 - 7 - 9 - 10
RV: 9 - 13 - 12 - 10 - 11

(b) Third set of routes for ARFV of 500

RI: 0 - 1 - 2 - 5 - 7 - 9 - 13 - 12
RII: 4 - 3 - 5 - 7 - 14 - 6 - 9 - 10
RIII: 8 -14 - 5 - 7 - 9 - 10
RIV: 9 - 13 -12 - 10 -11

5 Conclusions

The results show improvements over previous researcher works for the same network problem and demand matrix. It is also observed that Genetic Algorithm gives reasonably good values even with lesser pool size. However if pool size is increased results can be refined but it increases the computational time. Selection of ARFV value is affecting the total travel time per person. As it is observed that higher the value of ARFV higher the total travel time per person. However it is to be noted that time taken for solving the problem is not compared for the given network. For smaller network as considered in study there may not be much difference but computational burden increases with larger networks so computational time will also increase. The results have proved that simultaneous routing and scheduling using Genetic Algorithm for optimization has better edge over other existing approaches for routing and scheduling problems specially in the domain of Public Transportation.

References

Baaj, M.H. and Mahmassani, H.S. (1990) TRUST: A LISP Program for Analysis of Transit Route Configurations. Transportation Research Record 1283, pp.125-135.

Chakrobarthy Partha, Kalyanmory Deb and Subrahmanayam, S. (1995) Optimal Scheduling of Transit Systems Using genetic Algorithms, Journal of Transportation Engineering, Vol.121(6),pp.544-552.

Deb Kalyanmoy (1995) Optimization for Engineering Design, Algorithms and Examples Prentice Hall of India Pvt. Ltd., New Delhi, 381p

Dhasora Mahesh, (1994) Expert System for Route Network Design and Scheduling For Urban Bus Service. Ph.D. thesis, Department of Civil engineering, I.I.T.Bombay,353p.

Dhingra, S.L., (1980) Simiulation of Routing and Scheduling of city bus transit Network . Ph.D. Thesis, Department of Civil Engineering, I.I.T. Kanpur.

Draft Report of the MMPG on Mumbai Metro Study (1997) submitted to Government of India and Government of Maharastra. Pp.75

GOLDBERG, D.E. (1989) Genetic Algorithms in Search Optimisation and Machine learning, Addision Wesley Publishing Company , Mass London.

HASSELSTORM D. (1981) Public Transportation Planning - A Mathematical Programming Approach. Ph.D. thesis, Department of Business Administration, University of Gothenburg, Sweden.

KIDWAI, F.A. (1998) Optimal Design of Bus Transit Network: A Genetic Algorithm Based Approach. Ph.D.Thesis, Department of Civil Engineering, I.I.T. Kanpur.130p.

LAMPKIN, W., AND SAALMANS, P.D. (1967) The Design of Routes, Service Frequencies and Schedules for a muncipal Bus Undertaking: Case Study. Operation Research Quarterly,Vol.18,No.4,pp.375-397.

MANDL, C.E. (1980) Evaluation of optimization of urban public transport network. European journal of optimization research, Vol. 6, pp. 31-56.

MARWAH, B.R., UMRIGAR, F.S., AND PATTNAIK, S.B. (1984) Optimal Design of Bus Routes and Frequencies for Ahmedabad. Transportation Research Record 994. Transportation Research Board: Washington D.C., 12-25.

MURALIDHAR, S. (1999) Routing and Scheduling of Public Transportation Using Genetic Algorithms . M.Tech Thesis, I.I.T. Bombay.

Monthly Statistical Review (Part-II), August 1999, Volume -XXXIX, Bulletin -V, published by Planning and Control Section Traffic Department of BEST undertaking, pp.7

PATTNAIK, S.B. MOHAN, S. AND TOM, V.M. (1998) Urban Bus Transit Route Network Design Using Genetic Algorithm, Journal of Transportation engineering, ASCE.Vol 124.No 4,pp.368-375.