A Genetic Algorithm Based Railway Scheduling Model

G. Nirmala[1], D. Ramprasad[2]

[1]Head and Associate Professor in Mathematics, PG and Research Department of Mathematics,
K.N.G. Arts College for Women (Autonomous), Thanjavur. Tamilnadu, India

[2]Assistant Professor in Mathematics, PG and Research Department of Mathematics,
A.V.V.M. Sri Pushpam College (Autonomous), Poondi, Thanjavur, Tamilnadu, India

Abstract: ***In this paper we present an optimization model for train scheduling. This model constitutes one of the three major components of a solution approach for solving the transit network design problem. The problem of scheduling can be defined in the following general terms. Given the origin destination matrix for the train trips for design period, the underlying train network characterized by the overlapping routes. How optimally to allocate the trains among these routes? The train scheduling problem is solved in two levels. In the first level minimum frequency of trains required on each route. With the guarantee of load feasibility, is determined by considering each route individually. In the second level, the fleet size of first level is taken as upper bound and fleet size is again minimized by considering all routes together and using GAs. The model is applied to a real network, and results are presented.***

Keywords: Train scheduling, optimization, genetic algorithms.

1. Introduction

The design of train transit system may be considered as a systematic decision process consisting of five stages: network design, frequency setting, time table development, train scheduling and driver scheduling. However, the two most fundamental elements, namely, the design of routes and setting of frequencies, critically determine the system's performance from both the operator and user point of view. Significant savings in resources can be made by reorganization of train routes and frequency to suit the actual travel demand. The solution framework for transit network design consists of three major components, namely, transit route design, transit assignment and transit scheduling. In this paper transit trains scheduling problem is formulated and solved in two phases. In the first phase trains are assigned to individual routes by an interactive procedure. In the second phase, an attempt is made to further reduce the fleet size and genetic algorithms are used as an optimization tool. Genetic algorithms are search algorithms that are based on concepts of natural selection and natural genetics. The genetic algorithm method differs from other search methods in that it searches among a population of points and works with a coding of parameters, rather than the parameter value themselves. The transition scheme of the genetic algorithm is probabilistic, whereas traditional methods use gradient information. Finally the model is applied to a real network, and results are presented.

2. Proposed Methodology

First general formulation for optimal train allocation problem is given. In the present methodology a bi-level optimization is used to solve this problem. In the first level, minimum frequency of trains (then the number of trains) required on each route with guarantee of load feasibility is determined by considering each route individually. Then by summing up the number of trains each route fleet size is determined. In second level by taking the fleet size of first level as upper bound, the fleet size is again minimized by considering all routes together and using GAs.

2.1 General Formulation

In the present formulation, general model for train scheduling problem is adopted similar to Han and Wilson (1982) and given as follows:

Objective Minimize J= J(q_{ij}^{k}, f_k ,A_k)

Subject to Passenger flow assignments

$q_{ij}^{k} = g_{ij}^{k}(V^{ab}, f_r, A_r)$

$\forall$ k $\in SR, ij \in L_k$, r$\in x_{ij}$ and a,b$\in N$

Load feasibility: CAP× $f_k \geq (q_{ij}^{k})_{max}$ $\forall$ k $\in SR$

Fleet size: $\sum_{k=1}^{SR} T_k \times f_k \leq M$

Where

f_k – frequency of trains operating on route k.

A_k – set of other attributes associated with train route k.

CAP - capacity of trains operating on the networks routes

q_{ij}^{k} - passenger flow on link i-j of train route. K.

g_{ij}^{k} - general function form which determines passenger flow assignment on link i-j of train route k.

V^{ab} - origin destination flow between nodes a and b

N - set of nodes on the train network.

L_k - set of links on train route k.

SR - set of train routes.

T_k - round trip time of route k(including lay over time)

X_{ij} - set of routes offering same service between nodes i and j

M - total number of trains available.

The objective function in the general case should include wait time and crowding levels for all passengers. Since many trains will be operating close to or at capacity on portions of their trips, the specification of accurate wait time and crowding level function are extremely difficult (Han and Wilson, 1982). For this reason the simplified objective of minimizing the occupancy level at the most heavily loaded link on any route in the system is adopted here. This objective is different from, but related to minimizing wait times and crowding levels throughout the system, and is similar to the objective currently used by many operators in allocating trains in heavily utilized system.

The load feasibility constraint requires that in a given period of time passengers should not be prevented from boarding a train on their preferred route because inadequate of capacity has been allocated to that route. This does not, of course imply that every passenger will be able to board the first train on that route because random fluctuation in the load will mean that some train will be full at the heaviest points on each route. Some passengers who cannot board the first train on their preferred route may, in fact subsequently board an alternate route. Passenger path choice is based on an assumed flow assignment rule "Where there is one or more alternatives whose trip time is within a threshold of the minimum trip time a frequency share rule is applied". This is an allocation formula that reflects the relative frequencies of service on alternative paths.

2.2 First Level Optimization

The problem for first level optimization may be formulated as:

Objective Minimize Z = $\sum_{k=1}^{SR}(T_k \times f_k)$

Subject to Passenger flow assignments

$q_{ij}^{k} = g_{ij}^{k}(V^{ab}, f_r, A_r)\ \forall k \in SR$

Load Feasibility

CAP × $f_K \geq (q_{ij}^{k})_{max}\ \forall\ k \in SR$

The following algorithm is used to solve this problem.

Step 1 For the given origin destination transit demand matrix and transit route network, assume the same number of trips on each route. N = l ; f_k^n

Step 2 Assign the origin destination transit demand matrix on the train transit network using the assignment model discussed in next section.

Step 3 For each route, find out the link carrying the maximum flow and determine the number of trips on each route using the formula.

$$f_k^{\,n+1} = \frac{(q_{ij}^{k})_{max}}{2\times \text{train capacity}}$$

These numbers of trips are rounded off to next higher integer

Step 4 If $f_k^{\,n+1}$ is very small for all routes, go to Step 5; otherwise set n=n+1 and go to step 2.

Step 5 Output the number of trips required on each route.

Step 6 Find the number of train required on each route to cater to these trips using the formula:

$$N_k = \frac{f_k \times T_k}{\text{time period}}$$

Again these numbers of train are rounded off to next higher integer.

Step 7 Find the base fleet size (Wo) by summing up the number of trains on each route.

2.3 Second Level Optimization

In the first level optimization, the base fleet size has been determined by considering individual route's capacity and no attempt is made to get the minimum fleet size on global bases (i.e. considering all the routes together). The reason why one can still reduce the fleet size below the base fleet size may be attributed to the extensive overlapping of the routes. If there is no overlapping of the routes, on can't hope to reduce the fleet size below the base value. Though there are various reasons of, how extensive overlapping of routes may help to reduce the fleet size, two reasons are discussed below.

(a) In the routers shown in Figure I there is overlapping for many links. Suppose the links which carry the maximum flow (i.e. used for minimum number of train's determination in the first level of optimization) are (3) – (4) and (6) – (7) for routes RI and RII, respectively. If one train is reduced on route RI, there will be violation of the load feasibility at link (3) – (4), but because this link is common with route RII and reserve capacity is available on this link as this link is not the maximum flow carrying link for route number RII, the extra demand of link (3)-(4) may be taken care of by trains on route RII.

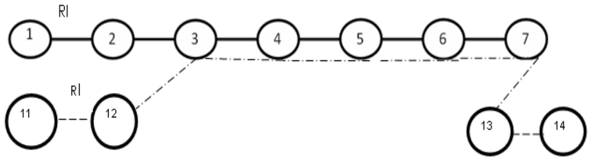

Figure 1: Example Network

(b) Take the example of two overlapping routes shown in Figure 2. Here route RI is assumed to be much longer than route RII. The numbers of trains required on a route to make a fixed number of trips are directly proportional to the length (round trip time) of the route. It is further assumed that route RI is along the overlapping portion of the two routes.

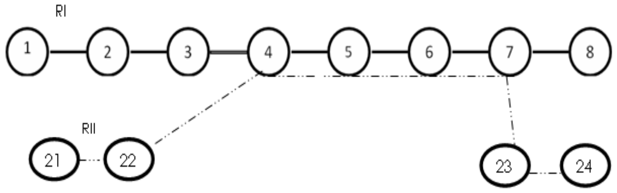

Figure 2: Example Network To Illustrate Train Reduction

Then if two trains are reduced on route RI and one train in increased on route RII, the number of trips on overlapping portion will remain the same and load feasibility constraint may not be violated even after one train is reduced below the base fleet size, these possibilities of reducing the base fleet size are investigated in the second level optimization using genetic algorithms(GAs).As GAs are not very common for transportation engineering applications, it is imperative at this stage to discuss their principles.

3. Genetic Algorithms

The idea of genetic algorithms (GAs) was first conceived by Professor John Holland of the University of Michigan in

1975. Genetic algorithms are computer based search and optimization algorithms which work on the mechanics of natural genetics and natural selection (Goldberg, 1989).The mechanics of a simple genetic algorithm are simple involving copying strings and swapping partial strings. The explanation of why this simple process works is subtle yet powerful. Simplicity of operation and implicit parallelization are two of the main attractions of the genetic algorithm approach.

3.1 Working Principle

GAs begins with the population of string structures created at random. Thereafter, each string in the population is evaluated. The population is then operated by three main operators- reproduction, crossovers and mutation- to create a hopefully better population. The population is further evaluated and tested for termination. If the termination criteria are not met, the population is again operated by above three operators and evaluated. This procedure is continued until the termination criteria are met. One cycle of these operators and the evaluation procedure is known as a generation in GA terminology.

```
begin
Initialize population of strings;
Computer fitness of population;
Repeat
        Reproduction;
        Crossover;
        Mutation;
        Computer fitness of population;
        Until (termination criteria);
end.
```

3.2 Termination Criteria

When the average fitness of all the strings in a population is nearly equal to the best fitness, the population is said to have converged. When the 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 GA, maximum number of generations is generally used as the termination criteria. The same has been used in the present study.

4. Problem Formulation for Second Level Optimization

As we know the number of trains on each route (N_k) from the first level optimization, it is sensible to make the search around these N_k values, in order to avoid otherwise a meaningless search. Therefore, a window is decided around the previously determined N_k values and search is made only in that window. In the present study, it is decided to search within a window of 8 trains around the previously determined N_k values. For example, if for a route k, value of N_k is 20 trains, then the search will be made only between 16 to 23 trains for this route. There is a rationale for selecting the window of 8 trains for search. In GA, variables are coded as binary strings and n bits are required for 2^n different values of a variable. Therefore, one bit will be required for 2 trains, 2 bits for 4 trains, 3 bits for 8 trains and n bits for 2^n trains. If we decide to take 2 bits for representing train window of 4 on each route for searching optima, it will give rise to a narrow search space and optima may lie outside this window. On the other hand if we decide to take 4 bits for representing train window of 16 on each route to search for an optima, it will exponentially increase the search space and may give many infeasible (i.e. Zero or negative train on a route) values. After it is decided to use the window size of 8 trains, 3 bits will be required for each route. Therefore for routes, a string of length 3×k bits will be required. Because the window size is 8, the lower and upper limits on the number of trains for a route will be,

$N_{kmin} = N_{ko} - 4$
$N_{kmax} = N_{ko} + 3$: respectively

Where N_{ko} is the number of trains on route k from first level optimization. The problem for the second level optimization may be started as below

Objective Minimize $Z = \sum_{K=1}^{SR} N_k$

Subject to

$N_{kmin} \leq N_k \leq N_{kmax} \; \forall \; k \in SR$

$\sum_{k=1}^{SR} N_k < W_o$

Passenger flow assignment

Load feasibility: CAP x $f_k \geq (q_{ij}^{k})_{max} \; \forall \; k \in SR$

Where,
N_k- Number of trains on route k, and
W_O- base fleet size (found in first level optimization)

5. Genetic Algorithm For Solution

In the above problem decision variables (number of trains on each route) can take only integer values and last two constraints are highly non-linear. Therefore, GAs which is best suited for such problems are used for solution. GA steps are given below.

Step 1. Computer N_{kmin} values for each route as : $N_{kmin} = N_{ko} - 4$

Step 2. Choose a selection operator, a crossover operator and a mutation operator. Choose population size, crossover probability and mutation probability. Choose a maximum allowable generation number.

Step 3. In this step, GA creates an initial population of strings randomly. Based on the number of routes the required string length can be calculated. For example, if there are k routes, the string length should be k × 3

Step 4. In the fourth step, string is decoded and the actual train number for each route are obtained using the formula $N_k = N_{kmin} +$ decoded value of k^{th} 3 bits of the string.

For example , if there are three routes and N_{kmin} values for these routes are 6,9 and 8, then for a typical string 110010101 the N_k values for the three routes will be 12(6+6), 11(9+2) and 13 (8+5), respectively :

Step 5. Calculate the fleet size by summing up N_k values for all routes. If this fleet size is greater than or equal to the base fleet size, assign fitness a very small value: otherwise proceed with next step.

Step 6. Compute frequencies using N_k values for each route. Assign the passengers on different links of the network using assignment model.

Step 7. If the load feasibility constraint is violated, assign fitness a very small value; otherwise computer fitness using the formula.

Fitness = $\frac{C}{1+\sum N_k}$

Where C is a constant used to normalize the objective function.

Step 8. If the entire population of strings is processed, compute the best and average fitness value in the generation and test for termination criteria; otherwise evaluate the next string in the population.

Step 9. If the current generation is equal to the maximum number of generations assumed, the program is terminated and the scheduling giving the minimum fleet size is considered as the optimal schedule; otherwise the GA operators –reproduction , crossover and mutation are applied on the current population to obtain a new population is processed again.

6. Summary and Conclusion

The optimal allocation of trains with a conventional approach poses considerable difficulties owing to the combinatorial nature of the problem and the complex nature of the route choice model. Hence genetic algorithms (GAs) are proposed as the computational tool because of their ability to handle large and complex problems. The solution framework for the present problem involves two phases: (1) Allocation of trains on individual routes with maximum link flow as the criteria, and (2) further reduction of trains on network basis making use of genetic algorithms as an optimization tool. The present study may also be extended by exploring the suitability of different GA parameters.

References

[1] Baaj, M.H. (1990). The Transit Network Design Problem: An AI-Based Approach, Ph.D. Thesis, Department of civil Engineering, University of Texas, Austin, Texas.

[2] Chakroborty, P., Deb, K. and Subrahmanyam, .P. S. (1995). Optimal Scheduling of Urban Transit systems Using Genetic Algorithms, ASCE Journal of Transportation Engineering, Vol. 121, No.6, pp 544-553.

[3] Furth, P. G. and Wilson, N. M. H. (1981). Setting frequencies on Bus Routes: Theory and Practice, Transportation Research Record 818, Transportation research Board, Washington, D. C., pp 1 – 7.

[4] Goldberg, D. E. (1989). Genetic Algorithms in search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Inc., Mass., USA.

[5] Dr.G.Nirmala and D.Ramprasad "Genetic Algorithm Approach for minimizing waiting times in Railway Networks" Narosa Publication (2010) ISBN 978-81-8487-074-9.

[6] Dr.G.Nirmala and D.Ramprasad "Genetic Operators with Algebra for equivalence relations" International conference on Algebra and its Applications PMU Journal, 2011.

[7] Dr.G.Nirmala and D.Ramprasad "Application of Genetic Algorithm in Graph Theory" International Journal of Scientific and Research Publications, Volume 2, Issue (July 2012), ISSN 2250 – 3153.

[8] Dr.G.Nirmala and D.Ramprasad "Innovative PMX – CX – Operators for GA to TSP" International Journal of Scientific and Research Publications, Volume 2, Issue 10 (October 2012), ISSN 2250 – 3153.

[9] Dr.G.Nirmala and D.Ramprasad "Genetic Algorithm with Topology to Solve minimizing waiting time of ALBP" Aryabhatta Journal of Mathematics and Informatics. (Accepted).

[10] Dr.T.Ramaraj and G.Nirmala "Genetic Algorithm for Maximum model discrimination design – Matrix – Represented Solutions" Science and Technology – 2004.

Author Profile

Dr. (Mrs) G. Nirmala, is Head and Associate Professor in Mathematics, PG and Research Department of Mathematics, K.N.G. Arts College for Women (Autonomous), Thanjavur. Tamilnadu, India

D. Ramprasad is working as Assistant Professor in PG and Research Department of Mathematics, A.V.V.M. Sri Pushpam College (Autonomous), Poondi, Thanjavur, Tamilnadu, India.