Models and Genetic Algorithms for the Optimal Riding Routes with Transfer Times Limited in Urban Public Transportation

Ruichun He* Yinzhen Li Zhongfu Zhang

Lanzhou Jiaotong University, Lanzhou 730070, P. R. China

Abstract *With the population increase and the traffic development, urban traffic systems are becoming more and more complicated and gigantic.In this paper, we proposed a 0-1 programming model and a hypergraph model for the shortest path problem with transfer times limited. And we design the improved priority-based encoding genetic algorithm for the hypergraph model. Finally, we give a numerical example.*

Keywords urban transportation, transfer, optimal riding route, hypergraph model, genetic algorithm

1 Introduction

With the population increase and the traffic development, urban traffic systems are becoming more and more complicated and gigantic. Especially, the traffic jam is more serious, and urban public transportation is playing a more important role than before. Therefore, the urban public passenger transportation is an important part of the urban traffic systems. For those passengers that they often choose public transportation, they always pay attention to the optimal riding routes[6,9]. In fact, the criteria to the optimal riding routes are quite different for the different decision makers. For example, some passengers expect the minimum travel time[10], some passengers probably expect the minimum cost or transfer times, etc. K.Goczyla,J.Cielatkowski[6] designed the optimal routing graph model while the timetable is known, and proposed different algorithms for different objectives. He[3] proposed the shortest travelling time model and transfer penalty model, then put forward to the polynomial-time algorithms of the models.

Some passengers dislike frequent transfer, that is to say, they always pay attention to both the minimum travel time and the transfer times. We call it the

*Corresponding author. E-mail address: herc@mail.lzjtu.cn

shortest path problem with transfer times limited. In this paper, we propose the mathematical model on this kind of problem. Considering the model is nonlinear and is also of NP complexity, we design the genetic algorithms.

2 0-1 Programming Models

Firstly, we assume that:

(1).The traffic network is connected, which means public vehicles can arrive at any station in the traffic network.

(2).The travel time is independent on the vehicle bus number, that is to say, the travel time is same from station i to station j for all public vehicles.

Now, we define the decision variables as follows:

$$X_{i,j}^{k} = \begin{cases} 1, & \text{from station i to station j by } k^{th} \text{ bus} \\ 0, & others. \end{cases}$$

Then, the shortest travel time with transfer times limited can be formulated as follows:

$$Min \sum_i \sum_j \sum_k X_{i,j}^k W_{i,j}^k + (\sum_i \sum_j \sum_k X_{i,j}^k - 1)\delta,$$

$$\text{s.t.} \begin{cases} \sum_j \sum_k X_{1,j}^k = 1, \\ \sum_i \sum_k X_{i,j}^k - \sum_i \sum_k X_{j,i}^k = 0, \\ \sum_i \sum_k X_{i,n}^k = 1. \end{cases}$$

$$\sum_i \sum_j \sum_k X_{i,j}^k \leq C - 1,$$

$$x_{i,j}^k = 0 \text{ or } 1.$$

where, the first part in the objective function is the travel time from node i to node j, the second is the delay time of transfer. $w_{i,j}^k$ is the travel time if node i and node j are direct connected by the bus k, otherwise $w_{i,j}^k = \infty$. δ is a constant given. The second is the constraints of the path from node 1 to node n.The third is the constraints of the transfer times.

It is obvious that the model is 0-1 programming model and is NP-Hard. Many researchers proposed the intelligent evolution algorithms to solve the integer programming problem in the last few decades. There was 0-1 coding genetic algorithms for 0-1 programming model too. But the method is very complicated and the calculational cost is tremendous.At some case, n-to-1 mapping may occur for the encoding.Thus, we put forward to transforming the model into a hypergraph model in the following way. By the way, because of the speciality of the problem it is possible that there are no feasible solutions for improper value of C.

3 Hypergraph Model and Genetic Algorithms

Generaly, we can transform the 0-1 programming model into a hypergraph model.

Let $G = \{V, E, W\}$ be a directed graph, $V = \{v_i | i = 1, 2, \cdots, n\}$ is the set of nodes,

$$E = \{e_{i,j} | i, j = 1, 2, \ldots, n, i \neq j\}$$

is the set of edges(arcs),

$$W = \{w_{i,j} : e_{i,j} | i, j = 1, 2, \cdots, n, i \neq j\}$$

is the set of weights of edges. Then, the definitions can be defined as follows:

Definition 3.1 *For two adjacent node i and j, if there exists an arbitrary bus from the node i to the node j, then we call the node i is arc tail of the arc(i, j), and the node j is arc head of the arc(i, j), the node i is the preorder node of the node j, the node j is the next node of the node i. G is defined as multi-weights graph if exists any arc(i, j) that related with more than one bus.*

Definition 3.2 *For a graph $H = \{V, E, W\}$, where arc $e_{i,j} \in E$, define $w^k_{i,j} \in W$ as the k^{th} weight of the arc $e_{i,j}$, the number of nodes related with the arc is called the cardinal number of the arc, remarked as $|e_{i,j}|$,$|e_{i,j}| = |k| + 1$, if $|e_{i,j}| > 2$, we call $e_{i,j}$ is hyperarc. If there are hyperarc in a graph H, H is hypergraph. The size of H is the sum of all hyperarc cardinal number:*

$$size(H) = \sum_{e_{i,j} \in E} |e_{i,j}|.$$

Apparently, the paths from node s to node t in the hypergraph are hyperpath and consists of the sequence of nodes,arcs and weights, the paths are following that:

$$P_{s,t} = (s, (s, i), w^k_{s,i}, i, (i, j), w^l_{i,j}, \cdots, m, (m, t), w^n_{m,t}, t)$$

Definition 3.3 *Assume $R_{s,t}$ is the set of all paths from node s to node t in the hypergraph H. The model such that*

$$p^*_{s,t} = min \sum w^k_{i,j} | w^k_{i,j} \in P_{s,t}, P_{s,t} \in R_{s,t},$$

$$|k| \leq C + 1,$$

is called the hypergraph model of the shortest path with transfer times limited. where $p^*_{s,t}$ is called the shortest hyperpath, $|k|$ is the number of different weights.

Professor Gen proposed the priority-based encoding method for SP as early as 1998[1]. The priority-based encoding genetic algorithm is a very effective method. Then he solved the bicriterion SP problem by this method in 2004[2,9]. But he only considered single weight network. To unconstraint hyperpath problem, the

$O(size(H))$ algorithm is proposed in[3,4,5,7,8]. The model in this paper is constraint hypergraph model and is NP-hard, so there can not exists any polynomial algorithm.

We design the genetic algorithm for the above model.

By the use of the idea of the priority-based encoding, aiming at hyperpath with constraints,we design the improved priority-based encoding genetic algorithm.

Firstly, we define some symbols and variables as follows:

$D(i,j)$: the j^{th} bus number departing from node i, $j = 1, 2, \cdots, N(i)$;

$N(i)$: the number of bus departing from the node i;

$C(k,j)$: the next arriving node(station) of the j^{th} bus that departing from the node k;

$d(i)$: the random priority of the node i, that is the random bus number departing from the node i;

$p(i)$: the bus number departing from the node i in the path;

$pn(i)$: the i^{th} node number in the path.

Then algorithm steps as follows:

Step 1. *Generate the initial population. Generate a random integer* $r_1 | r_1 \in \{1, 2, \cdots, N(i)\}$ *for any node* $i | \forall i \in V \setminus \{n\}$ *and let* $d(i) = r_1$, *then* $d(i) | i = 1, 2, \cdots, n-1$ *is a chromosome and the length is* $n-1$ *. Repeat this process until we get* $size$ *chromosomes.*

Figure 2 shows the chromosome code to a simple urban hyper network shown in Figure 1. The priority of node is bus number generated randomly.

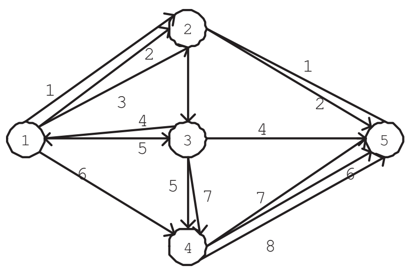

Figure 1: A simple example of a hyper network

Node i	1	2	3	4			
Priority n(i)	5	3	2	2			

Figure 2: The code of a chromosome

Step 2. *Crossover operator. Select chromosomes by crossover probability* P_c *and randomly group them by pairs. In this way we can get* $\lfloor \frac{size}{2} \rfloor$ *pairs as crossover parents.*

Step 2.1. *Random multi-position crossover is adopted here. We can get an offspring by crossing one pairs. For example, an offspring is generated by selecting preserved genes of $parent1$ at random and replace unpreserved genes of $parent1$ with corresponding genes of $parent2$. Another offspring can be get by the same process. The process is shown in figure 3:*

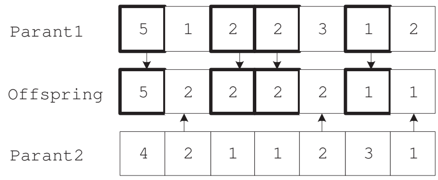

Figure 3: Example of OX operator

Step 2.2. *Examine the feasibility of the offspring generated. For each gene in the offspring chromosome, if $d(i) \leq N(i)$ then the offspring is feasible, otherwise, generate random integer r_1 ($r_1 \in \{1, 2, \cdots, N(i)\}$), let $d(i) = r_1$.*

Step 3. *Mutation operator. Select chromosomes by mutation probability P_m . The swap mutation operator is used here.*

Step 3.1. *Select two gene positions of a mutation parent randomly and then swap them. The process is shown in Figure.4.*

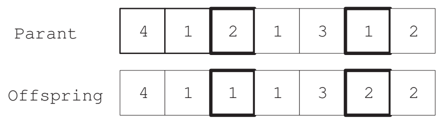

Figure 4: Example of mutation operator

Step 3.2. *Examine the feasibility of the offspring generated in Step 3.1. For each gene in the two swap positions, if $d(i) \leq N(i)$ then the offspring is feasible, otherwise, generate random integer $r_1 | r_1 \in \{1, 2, \cdots, N(i)\}$, let $d(i) = r_1$.*

Step 4. *Decode.Obviously the transform at the node 3 in figure 2 is illogicality. It is preferable to select bus 5 rather than transform bus 5 with bus 7 to arrive at the node 4 directly. Therefore, the mend operator is performed during the decoding process(The mended chromosome is showed in figure 5).*

(1).Let $p(1) = d(1), pn(1) = 1, i = 1, j = 1, h = 0, P = \phi$;

$P = 1$

Node i	1	2	3	4			
Priority n(i)	5	3	1	2			

Figure 5: The mended code of a chromosome

(2). $i = i+1, j = pn(i-1), pn(i) = C(pn(i-1), p(i-1); P^c = \{pn(i)\}; P + \{pn(i)\} \longrightarrow P;$

if $pn(i) = n$ *then goto (5) else (3)*

(3).$p(i) = d(pn(i));$

if $C(pn(i), p(i)) \in P$*, then mend* $d(i)$*; else goto (4).*

Mend operator for d(i): if $\{1, 2, \cdots, N(i)\} - P^c \neq \phi$*, then generate a random integer* $r_1 | r_1 \in \{1, 2, \cdots, N(i)\} - P^c$*, let* $d(pn(i)) = r_1$*, goto (3); else the node* $pn(i)$ *is proved to be dead node which means the path can not arrive at node n. Then we punish the chromosome, let* $h = C + 1$*, goto (5).*

(4). if $(D(pn(i-1), p(i-1)) \neq D(pn(i), p(i)))$ *then*

if $(D(pn(i-1), p(i-1)) \in D(i,k) | k = 1, 2, \cdots, N(i)$ *then*

$d(pn(i) = p(i-1), p(i) = p(i-1)$

else $h = h + 1;$

goto(2)

(5).If $h > C$*, We are sure the corresponding chromosome is unfeasible. The unfeasible chromosome can turn into feasible one or contribute to the evolve process according with the Evolution Strategies theory. Hence, we would not exclude the chromosome from the population rather than punish the chromosome and let L(j)=M. Where L(j) is the path length of the chromosome* j *and* M *is the maximum integer.*

If $h \leq C$*, then the decoding process is succeed, where* $pn()$ *is the node number along with the path from node 1 to node n and* $D(pn(), p())$ *is the corresponding bus number. Then we can calculate the path length L(j).*

Step 5. *Evaluate the fitness function of chromosomes group.*

The fitness function of chromosome k is as follows:

$$eval(k) = \frac{\frac{1}{L(k)}}{\sum\limits_{j=1}^{size} \frac{1}{L(j)}}, k = 1, 2, \cdots, size.$$

Step 6. *We select* $size$ *chromosomes as the offspring chromosomes by wheel approach.*

The probability of k^{th} *chromosomes to be selected is*

$$q(k) = \sum_{j=1}^{k} \frac{eval(j)}{\sum\limits_{i=1}^{size} eval(i)},$$

Generated the random real $r \in [0,1]$, if $q(k-1) \le r \le q(k)$ then the k^{th} chromosome is selected.

Step 7. *Repeat step 2-step 6 until the predetermined rounds is run. The best solution is the optimal feasible chromosome reserved during the iteration.*

4 Numerical Example

A urban public transportation network with 45 edges and 16 nodes is given for numerical experiment in Figure 6. The travel time list in table 1 and bus number list in table 2. The problem is to find out the optimal riding routes with transfer times limited in urban public transportation.We designed the computer program for the algorithm. Where transfer times $C = 3$, $P_c = 0.4$, $P_m = 0.5$. The computing results are:

the shortest path:1,3,8,12,15,16;

the transfer bus number of shortest path:$1\overset{3}{—}3\overset{8}{—}8\overset{8}{—}12\overset{11}{—}15\overset{12}{—}16$;

the shortest travel time: 41.

Figure.7 shows the evolutional process of optimal paths. Obviously,the convergence of the algorithm is good.

5 Conclusion

The problem of transform in urban public transportation is a very interesting and realistic research topic and is still an open kind one. Based on urban public transportation networks in this paper, the optimal riding routes with transfer times limited model is formulated in which not only the travel time of paths but also the transfer times is considered. The experiment result shows that the genetic algorithm designed has good robust and convergence. But we only consider the transfer problem with certain factors in this paper. The problems with the uncertain factors need to be researched further.

Acknowledgement

This work is supported by Qinglan Project of Lanzhou Jiaotong University(QL-05-06A).

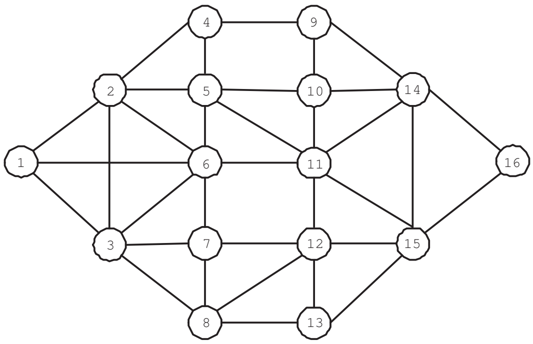

Figure 6: An example of a urban transportation network

Table 1: The bus travel time

arc	travel time	arc	travel time	arc	travel time
1,2	5	4,9	11	9,14	9
1,3	10	5,6	7	10,11	7
1,6	7	5,10	10	10,14	9
2,3	6	5,11	5	11,12	10
2,4	8	6,7	8	11,14	8
2,5	7	6,10	9	11,15	7
2,6	10	7,8	10	12,13	6
3,6	8	7,12	12	12,15	8
3,7	6	8,11	9	13,15	11
3,8	5	8,13	10	14,15	10
4,5	10	9,10	6	14,16	13
15,16	9				

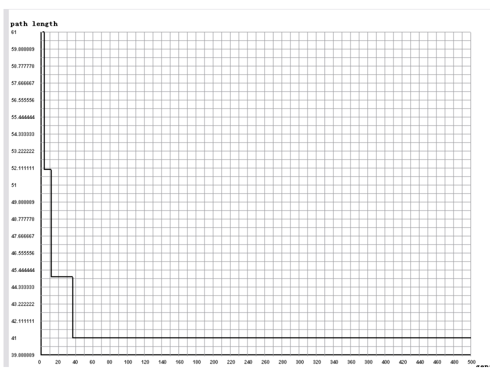

Figure 7: The evolutional process of optimal paths

Table 2: The bus number and the riding route

bus number	riding route	bus number	riding route	bus number	riding route
1	1,2,4,9,10	5	6,5,4,9,10,14	9	11,6,5,4,2,1
2	1,6,5,11	6	7,8,13,12,11,14	10	11,6,5,2,1
3	1,3,7,8,12	7	11,10,9,4,2	11	6,7,8,12,15
4	2,6,7,8,13	8	11,12,8,3	12	6,3,8,13,15,16

References

[1] R.Cheng,M.Gen.Apriority based encoding and shortest path problem[R].Ashikaga Institute of Technology,1998

[2] M.Gen. Bicriterion network design problem[R]. the 3d International Information and Management Science Conference, Dunhuang, P.R.of China,2004: 5, 419–425.

[3] Ruichun He. Problem of the Optimum Riding-convert Route in Urban Mass Transit System[R].the 3d International Information and Management Science Conference, Dunhuang, P.R.of China,2004: 5, 269–272.

[4] D.Pretolani.A directed hypergraph model for random time dependent shortest paths[J].European Journal of Operational Research 2000:123, 315-324.

[5] S.Nguyen,S.Pallottino.Hyperpaths and shortest hyperpaths[M].Combinatorial Optimization Lecture Notes in Math.,Vol.1403, 258-271.

[6] K.Goczyla,J.Cielatkowski.Optimal routing in a transportation network[J].European Journal of Operational Research,1995:87, 214-222.

[7] N.M. Ruphail, et al. A Dss for dynamic pre-trip route planning[A]. Applications of advanced technologies in transportation engineering[C]: Proc of the 4th Intl. Conference,1995: 325–329.

[8] Qun Yang . Research on the path selection based on multiple rational paths[J]. Journal of managing Engineering, 2004: 4, 42–45.

[9] R.Cheng, M.Gen.Resource constrained project scheduling problem using genetic algorithms[J]. Inter. J. of Intelligent Automation and Soft Computing, 1997:3, 273–286.

[10] X.Cai, T.kloks, C.K.Wong. Shortest path problems with time constraints[C]. In: Proc 21st international symposium on mathematical foundations of computer science, Cracow, Poland, 1996, 255–266.

[11] B.V. Cherkassky, Andrew V. Goldberg, Tomasz Radzik. Shortest paths algorithms:Theory and experimental evaluation[J]. Mathematical programming, 1996:73, 129–174.