

Hybrid Intelligent Algorithm for Determining Network Capacity with Transportation Time Reliability Constraints

Wangtu Xu*, Yuan Li and Hui Wang

School of Architecture and Civil Engineering, Xiamen University, Xiamen 361005, PR China

Peifeng (Patrick) Hu

*Department of Civil and Environmental Engineering, University of Nevada Reno
Reno, Nevada 89557-0152, USA*

Received 9 March 2011

Accepted 25 November 2011

Abstract

Motivated by a problem in the commodity transportation, a mathematical model is developed to calculate capacity of single-commodity network when the time reliability levels of transporting commodity between origin-destination pairs are constrained. We use a hybrid intelligent algorithm, in which genetic algorithm is embedded with Monte Carlo simulation to solve the optimization model. In the hybrid intelligent algorithm, the genetic algorithm is used to report the best path flow solutions and the Monte Carlo simulation is to check the feasibility of the chromosomes of genetic algorithm. With a computational experiment, the fact that network capacity decreases with the increase of the transportation time reliability level is validated. The efficacies of the developed procedures are examined by comparing the computational times of solving algorithm with that of previous work.

Keywords: network capacity, transportation time reliability, single-commodity flow, genetic algorithm, Monte Carlo simulation.

1. Introduction

Recently, considerable attentions have been given to study the determination of the Transportation Network Capacity (TNC). Several researchers have proposed various methods for different type of networks¹⁻³. Transportations on road network, communication network, and computer network are frequently suffered from serious congestions due to capacities constraints of the arcs⁴. On the other hand, much less attention has been devoted to TNC in the context of network of which arc capacities are uncertain, where every arc in the network has an uncertain capacity to transporting commodity⁵. The capacity determination problem in

this type of network is similar to the maximal throughput problem in stochastic flow network⁶.

In the daily lives, transporting commodity via commodity networks is a common occurrence. However, commodity transportation can be far from being pleasant due to a variety of reasons such as congestions, failures of facility function, and delays. The commodity transportation network consists of shippers, carriers, and transportation facilities including arcs, nodes and so on. The carriers or the transportation agencies would like to attract more shippers by improving the level of service (LoS) of their transportation network. On Time Transportation (OTT) (or On Time Delivery (OTD)) has been used as a key performance indicator of the commodity transportation's LoS. From a shipper's perspective, it

* Corresponding author: ato1981@xmu.edu.cn.

would count up what time the shipments are dispatched and what proportion is late. From a carrier's perspective, if the customer wishes to change their order then it can often do this profitably with a time penalty. Also if some circumstances occur whereby a carrier would know that it cannot deliver as originally intended, it may redefine the delivery schedule. Transportation time reliability (TTR) (or Delivery time Reliability (DTR)) is a broadly used standard indicator measurement in freight transportation industry to measure the reliability of a carrier's delivery time confirmation for a customers OTT requirement⁷.

The objective of this paper is to determine the transportation capacity on a single-commodity network with required transportation time reliability level constraints on the condition of uncertain arc capacities. Taking into account of how reliable the commodity could be completely transported, we aiming to give out the maximum attainable throughput between Origin-Destination pairs on the given commodity network on which capacities of arcs are stochastic. Transportation Time Reliability is defined as the probability that the network could successfully transport a given amount of commodities at a certain time of interest. This problem in this paper is named as Network Capacity with Transportation Time Reliability Constraints (NCTTRC). The network capacity determined model presented in this paper yield to both arc capacity constraint and TTR constraint in contrast to the algorithms developed in Ref. 8 and Ref. 9 which only take account of one such constraint. A solution technique using genetic algorithm (GA) embedded with Monte Carlo simulation (MCs) is employed to solve the presented formulation. The rest of this paper has been organized as follows. In Sec. 2, we describe the studied problem firstly then give out the relevant mathematical formulation as well as the solution methodology for the proposed model is shown. Computational experiment and computational time analysis is presented in Sec. 3. Conclusions and future research issues are outlined in Sec. 4.

2. Problem Statements

2.1. General Statements

The capacity of a network indicates the maximum attainable throughput of the given network⁸. In the

conventional network, the capacity problem is very easy to state: if a network has capacities on arc flows, how to send as much flow as possible from an origin node to a destination node without exceeding the capacity of any arc. This is the well-known single commodity capacity problem of a network. It has been well established with the mathematics model with its solution technique⁹⁻¹⁰.

It the context of pervious researches, the networks involved capacity have been assumed having deterministic arc capacities. However, in real life, the arc capacities of networks are often random due to the uncertain events such as accidents, maintenances, capacity failures, etc.¹¹. When dealing with the network with uncertain arc capacities, how to compute its capacity is a complex problem and always appeal to the TTR index¹¹, which has been discussed elaborately in terms of various methods¹²⁻¹³. Moreover, the network capacity computation with constraint of TTR has been proved to be a NP-hard problem¹⁴.

On the other hand, network capacity problem is an important issue with transportation time reliability constraints and worth to be researched for several reasons. First, it could be used to provide the shipping reliability information such as how many commodities could be transport certainly in a give time between Origin-Destination nodes. Second, network capacity considering the time reliabilities constraints between two nodes indicates the connectivity of this pair of nodes and is a key factor for adjusting the network configuration by the decision-maker. The network capacity with transportation time reliability requirements provides shipper the information of selecting carriers base on which carrier has the most reliable transportation capacity. Moreover, network capacity with transportation time reliability constraint problem could be used in conjunction with the carrier selection system. Finally, the problem is very useful extension of the maximal flow problem, in particular when considering stochastic networks and service performance evaluation.

2.2. Mathematical Formulation

Define a single-commodity network as graph $G = (P, A)$, where P is the set of nodes, and A is the set of arcs. Denote \bar{A} as the total number of arcs of the network, N as the total number of origin-destination (OD) pairs, and $n = 1, 2, \dots, N$ as index of the Origin-Destination pair.

Let q_n be the amount of commodities transported between the n th Origin-Destination pair; $\mathbf{Q}=[q_n], n=1,2,\dots,N$, be the matrix of commodities amounts transported between all Origin-Destination pairs in the network. Denote K_n as the number of paths connecting origin node and destination node of the n th Origin-Destination pair, and k as index of the path. Let x_k^n be the commodity flow on the k th path between the n th Origin-Destination pair, and x_a be the commodity flow on arc a , $n=1,2,\dots,N$, $a \in A$; t_k^n be the transportation time on the k th path between the n th Origin-Destination pair, and $t_a(x_a)$ be the transportation time on arc a when x_a amounts of commodities are transported. Apparently, $t_k^n = \sum_{a \in A} \delta_{a,k}^n t_a(x_a)$, $k=1,2,\dots,K_n$, $n=1,2,\dots,N$,

$\delta_{a,k}^n$ is the path-arc incidence variable, if arc a is included on the k th path between the n th Origin-Destination pair, it equals to 1, otherwise, it is set as 0. Moreover, we consider the stochastic arc capacity and congestion effect on an arc, and represent the time of transporting x_a units' commodities on arc a by BPR

function¹⁵: $t_a(x_a) = t_a^0 \left[1 + \eta \left(\frac{x_a}{\zeta_a} \right)^\lambda \right]$; where, t_a^0 is

the free-flow transportation time on arc a . η , λ are the BPR parameters¹⁵.

Generally, TTR between the n th Origin-Destination pair is defined as q_n units' commodities could be successfully transported at T_n units' time, and it could be displayed it as probability like:

$$L_n = \text{Prob}\{t(q_n, \zeta, n) \leq T_n\} = \text{Prob}\left\{\max_{k=1,2,\dots,K_n} \{t_k^n\} \leq T_n\right\} \quad (1)$$

where, $\zeta = \{\zeta_a\}$ denotes the stochastic arc capacity vector. Given the stochastic arc capacity vector ζ , let event $\{t(q_n, \zeta, n) \leq T_n\}$ denote that network could successfully transport q_n units' commodities between the n th Origin-Destination pair in T_n unit's time constraint. $t(q_n, \zeta, n)$ is the maximal path travel time when transporting q_n units' commodities between the n th Origin-Destination pair under stochastic arc capacity vector ζ , and

$$t(q_n, \zeta, n) = \max_{k=1,2,\dots,K_n} \{t_k^n\} \quad (2)$$

For a decision-maker, it should make sure that the time reliability of transporting commodities between an Origin-Destination pair is higher than a confidence level γ_n to assure the service quality, such that

$$L_n = \text{Prob}\{t(q_n, \zeta, n) \leq T_n\} = \text{Prob}\{t_n^{\max} \leq T_n\} \geq \gamma_n \quad (3)$$

To evaluate the network transportation capacity, time reliability constraints for an entire network's commodities transportation could be written as a matrix representation

$$\mathbf{L} = \text{Prob}\{\zeta | t(\mathbf{Q}, \zeta, \mathbf{N}) \leq \mathbf{T}\} \geq \gamma \quad (4)$$

where, $\mathbf{L}=[L_n]$, $\mathbf{Q}=[q_n]$, $\mathbf{T}=[T_n]$, $\gamma=[\gamma_n]$, $\zeta=[\zeta_a]$, $n=1,2,\dots,N$, $\forall a \in A$.

Aiming at finding out the maximal commodity transportation capacity of the network under TTR constraint, a mathematical model is set as follow.

$$\begin{cases} \max z = \sum_{n=1}^N q_n \\ \text{s. t.} \\ \mathbf{L} = \text{Prob}\{\zeta | t(\mathbf{Q}, \zeta, \mathbf{N}) \leq \mathbf{T}\} \geq \gamma \\ \sum_{k=1}^{K_n} x_k^n = q_n, n=1,2,\dots,N \\ x_a = \sum_{n=1}^N \sum_{k=1}^{K_n} \delta_{a,k}^n x_k^n, \forall a \in A \\ x_a \leq c_a^{\max}, \forall a \in A \\ x_k^n \geq 0, k=1,2,\dots,K_n, n=1,2,\dots,N \end{cases} \quad (5)$$

In the formulation above:

- the objective function is to maximize the total amounts of commodities transported between all Origin-Destination pairs.
- the first equation is the set of TTR constraints. They state that the probabilities which commodities are successfully transported during time between all Origin-Destination pairs should exceed the given confidence level.
- the second and the third equations are the flow conservation constraints.
- the fourth equation says that flow on an arc should not exceed its maximal practical arc capacity.
- the fifth equation is the non-negative condition for the flow on each path between each Origin-Destination pair.

In our model, the TTR constraints contain a series of uncertain variables, i.e., the uncertain arc capacities. Due to these uncertain variables, our model differs from those mentioned in the previous literatures. How to treat these uncertain variables and then solve the model has not been dealt with before.

2.3. Solution Methodology

Generally, the nonlinear programming with stochastic function constraints as Eq. (5) could not be converted into a deterministic form. There is limited deterministic algorithm to solve this type of formulation¹⁹. We design a hybrid intelligent algorithm (HIA) for it. This algorithm integrates MCs into GA. Genetic algorithm (GA) is a stochastic search method for optimization problems based on the mechanics of natural selection and natural genetics¹⁶. Detailed descriptions and applications of GA could be seen in Ref. 17. Monte Carlo simulation methods are a class of computational algorithms that rely on repeated random sampling to compute their results. Because of their reliance on repeated computation and random or pseudo-random numbers, MCs are most suited to be used when it is unfeasible or impossible to compute an exact result with a deterministic algorithm¹⁸.

Here, the methods and procedures used to process the stochastic function are synthesized together in a MCs framework. By simulating the stochastic behaviour of the arc capacity through repeated sampling from random variables with the given probability distribution, MCs could be employed to analyse the TTR between the Origin-Destination pairs for solving the model of Eq. (5). Main steps of this algorithm are shown as follows.

Step.1 Design the uncertain function as $L(\mathbf{x}) : \mathbf{x} \rightarrow \Pr\{\zeta | t(\mathbf{Q}, \zeta, N) \leq \mathbf{T}\}$. Denote the chromosome as \mathbf{v} to represent a solution of the optimization problem \mathbf{x} .

Step.2 Randomly initialize pop_size chromosomes whose feasibilities should be checked via MCs.

Step.3 Update the chromosomes by crossover and mutation operations in which the feasibilities of the offspring are also needed be checked.

Step.4 Calculate the objective values for all chromosomes.

Step.5 Compute the fitness of each chromosome based on the objective values.

Step.6 Select the chromosomes by spinning the roulette wheel¹⁹⁻²⁰.

Step.7 Repeat the third to the sixth steps for a given number of cycles.

Step.8 Report the best chromosome as the satisfactory solution.

Go into details, operations of the GA and MCs are as follows:

- Operations of the GA:
 - (i) **Chromosome representation.** Code the chromosome as $\mathbf{v} = [x_1^1, x_1^2, \dots, x_1^{K_1}, \dots, x_n^k, \dots, x_n^{K_N}]$ to represent a solution $\mathbf{x} = [x_1^1, x_1^2, \dots, x_1^{K_1}, \dots, x_n^k, \dots, x_n^{K_N}]$ of the optimization problem, where x_n^k denotes the commodity flow on the k th path of the n th Origin-Destination pair.
 - (ii) **Chromosome feasibility checking.** The feasibility of a chromosome is determined by both transportation time reliability constraint set and arc capacity constraint set. We employ a 0-1 function, *Reliability_Check*(\mathbf{v}) to check whether a chromosome satisfies the transportation time reliability constraint. For a chromosome \mathbf{v} , if *Reliability_Check*(\mathbf{v}) = 1, and that the commodity flow on each arc, x_a is smaller than its maximal practical capacity c_a^{\max} , such that $x_a \leq c_a^{\max}$, $\forall a \in A$. We could say that \mathbf{v} is feasible, otherwise it is not. That is, feasibilities of the chromosomes are test via running the procedure of the function *Reliability_Check*(\mathbf{v}), if the return value is 1, \mathbf{v} is feasible, otherwise is not.
 - (iii) **Chromosome initialization.** Define an integer *pop_size* as the population size of the chromosomes¹⁹. The *pop_size* chromosomes will be initialized in the following manner. Firstly generate a feasible initial chromosome, $\mathbf{v}_0 = [\dots, x_a^k, \dots]$ via MCs. Then randomly select a direction \mathbf{d} in R^n and define a chromosome \mathbf{v} as $\mathbf{v}_0 + M \cdot \mathbf{d}$ if it is checked to be feasible. Otherwise, set M as a random number between 0 and M until $\mathbf{v}_0 + M \cdot \mathbf{d}$ is feasible. Note that a feasible chromosome can be found in a finite time by choosing a new random number M since \mathbf{v}_0 is an interior point. Repeat this process *pop_size* times and produce *pop_size* initial feasible chromosomes $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{pop_size}$.
 - (iv) Another relevant operations in GA including Evaluation, selection, crossover and mutation operation are referred to Ref. 19 and Ref. 20.
- Operations of the MCs:

- (v) MCs is employed to initialize the chromosome set according to an Uniform distribution. $\mathbf{v} = [\dots, x_n^k, \dots]$, where $x_n^k \sim u[x_{\min}, x_{\max}]$, $k = 1, 2, \dots, K_n, n = 1, 2, \dots, N$ represents that x_n^k is generated uniformly from a space of which the lower bound is x_{\min} and the upper bound is x_{\max} .

That is x_n^k follows an Uniform distribution, and is generated via MCs.

- (vi) MCs is used to generate the set of uncertain arc capacities, ζ .

3. Computational Experiments

3.1. Computational experiment using a single network

A single-commodity network shown in Fig. 1 is employed to test the proposed model. There are 9 points, 12 arcs and 4 Origin-Destination pairs (1-8, 1-9, 2-8, 2-9). Capacity of each arc is stochastic. The basic information of the arc is also shown in Fig. 1.

In Fig. 1., the data on an arc, $N(\mu, \sigma^2)$ denotes the arc capacity ζ_a following a Normal distribution whose expected value is μ and variance is σ^2 . In this network, there are four Origin-Destination pairs. The first Origin-Destination pair has 3 paths, and, the second and the third have two and the fourth has three. The details of the paths between Origin-Destination pairs are also shown in Figure 3. Then the chromosome for the GA in HIA could be designed as $\mathbf{v} = [x_1^1, x_1^2, x_1^3, x_2^1, x_2^2, x_2^3, x_3^1, x_3^2, x_3^3]$. The detailed computational steps

of GA in the HIA to solve the problem in this paper could be found at the website²¹, as well as the corresponding source code in C language.

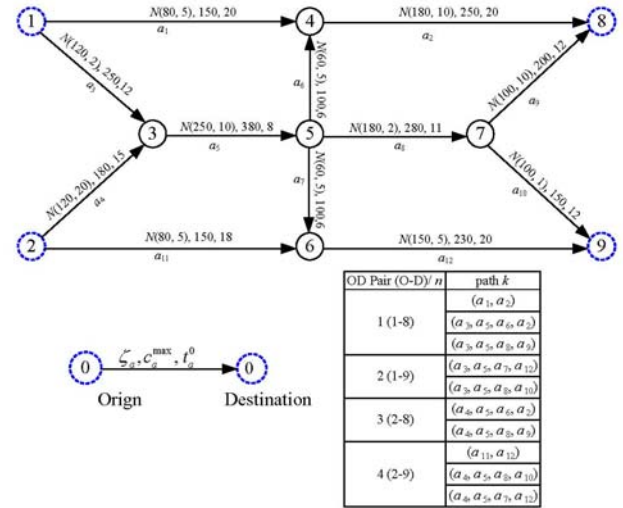


Fig. 1 Single-commodity network for computational experiment

Versus different transportation time reliabilities confidence level vectors (γ) and different required transportation time vectors (\mathbf{T}), several runs of the HIA (5000 cycles in *Reliability_Check* (v), 30 population size, 1000 generations, 0.1 crossover probability and 0.3 mutation probability in GA) show the optimal solutions and the objective function values in Table 1.

Table 1. The optimal solutions and the objective function values versus different γ and \mathbf{T}

Strategy #	\mathbf{T}	γ	The optimal solution (x)	The objective function value (max z)
1	[55, 55, 55, 55]	[0.90, 0.90, 0.90, 0.90]	No solution	N/A
2	[50, 60, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[95.33, 0.47, 51.37, 13.29, 33.72, 30.76, 19.76, 119.97, 88.32, 37.10]	465.12
3	[50, 70, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[95.14, 2.36, 63.35, 10.55, 21.22, 35.85, 26.17, 119.94, 91.05, 24.96]	490.58
4	[50, 80, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[95.14, 2.36, 63.34, 10.54, 21.22, 35.85, 26.17, 119.94, 91.05, 24.96]	510.58
5	[50, 75, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[85.71, 30.89, 25.60, 38.22, 48.53, 16.43, 44.45, 119.74, 93.35, 25.38]	528.31
6	[60, 80, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[114.10, 33.11, 36.86, 41.74, 49.72, 17.15, 63.70, 119.96, 70.34, 25.68]	572.36
7	[70, 70, 70, 70]	[0.90, 0.90, 0.90, 0.90]	[109.71, 49.02, 45.73, 16.28, 76.92, 29.67, 18.77, 119.88, 63.27, 41.82]	571.07
8	[70, 80, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[107.37, 38.79, 60.48, 22.55, 72.99, 25.14, 26.65, 119.98, 73.17, 46.77]	593.92
9	[80, 80, 80, 80]	[0.90, 0.90, 0.90, 0.90]	[133.63, 51.67, 83.23, 31.05, 40.45, 26.97, 50.74, 110.72, 53.25, 41.05]	622.78
10	[90, 90, 90, 90]	[0.90, 0.90, 0.90, 0.90]	[142.49, 79.18, 91.31, 15.18, 21.56, 2.12, 40.86, 119.99, 73.24, 56.53]	642.45
11	[150, 150, 150, 150]	[0.90, 0.90, 0.90, 0.90]	[150.00, 50.08, 100.35, 50.45, 5.13, 12.65, 32.60, 120.00, 86.21, 42.52]	650.00
12	[60, 60, 60, 60]	[0.10, 0.10, 0.10, 0.10]	[117.80, 15.60, 66.89, 27.91, 79.03, 43.52, 53.21, 109.23, 28.60, 27.20]	568.98
13	[60, 60, 60, 60]	[0.20, 0.20, 0.20, 0.20]	[116.93, 14.73, 66.01, 27.03, 78.16, 42.64, 52.33, 108.35, 27.73, 26.32]	560.25

14	[60, 60, 60, 60]	[0.30, 0.30, 0.30, 0.30]	[116.40, 14.20, 65.49, 26.50, 77.63, 42.11, 51.80, 107.82, 27.20, 25.80]	555.01
15	[60, 60, 60, 60]	[0.40, 0.40, 0.40, 0.40]	[114.95, 12.75, 64.04, 25.05, 76.18, 40.67, 50.36, 106.38, 25.75, 24.35]	540.47
16	[60, 60, 60, 60]	[0.50, 0.50, 0.50, 0.50]	[114.30, 12.10, 63.39, 24.41, 75.53, 40.02, 49.71, 105.73, 25.10, 23.70]	533.98
17	[60, 60, 60, 60]	[0.60, 0.60, 0.60, 0.60]	[113.42, 11.22, 62.50, 23.52, 74.64, 39.13, 48.82, 104.84, 24.21, 22.81]	525.10
18	[60, 60, 60, 60]	[0.70, 0.70, 0.70, 0.70]	[112.99, 10.79, 62.08, 23.09, 74.22, 38.71, 48.40, 104.42, 23.79, 22.39]	520.87
19	[60, 60, 60, 60]	[0.80, 0.80, 0.80, 0.80]	[112.77, 10.57, 61.86, 22.87, 74.00, 38.48, 48.17, 104.19, 23.57, 22.17]	518.64
20	[60, 60, 60, 60]	[0.90, 0.90, 0.90, 0.90]	[112.52, 10.32, 61.61, 22.62, 73.75, 38.23, 47.92, 103.94, 23.31, 1.91]	516.13

In Table 1, Strategy 2 denotes that we use transportation time vector $\mathbf{T} = [50, 60, 80, 80]$ and transportation time reliability constraint vector $\gamma = [0.90, 0.90, 0.90, 0.90]$ for a run of HIA. Finally we get a feasible chromosome \mathbf{v} as the solution $\mathbf{x} = [95.14, 2.36, 63.35, 10.55, 21.22, 35.85, 26.17, 119.94, 91.05, 24.96]$. The objective function value $\max z = \sum_{n \in N} q_n = \sum_{n \in N} \sum_{k \in K_n} x_n^k = 490.58$.

It could be seen from Table 1 that the objective function values differ to each other. In the cases of fixed transportation time reliability constraints (Strategy 1-11), the objective function value (namely, the transportation capacity of network) increases when the required transportation time increase. It is easy to understand this situation that if the transportation time constraint is large, the decision-maker would have more flexibility to use the network resources. While if the transportation time is limited, the transportation mission might not been successfully finished on the network. In the cases of fixed transportation time constraints (Strategy 12-20), the objective function value decreases with the increase of the transportation time reliability. This tells us that network capacity would decreases if customers ask the high level reliabilities of successfully transporting commodities between all Origin-Destination pairs in limited time requirements.

3.2. Comparing with the published work

In Ref. 19 and Ref. 20, Liu also developed a hybrid intelligent algorithm to deal with the optimization problem with stochastic constraints. This algorithm integrates a back-propagation neural network (BPNN) in GA. When solving an optimization problem with Liu's HIA, the BPNN is firstly trained with a large number of sampling data. Then the trained BPNN is used in GA to approximate the stochastic function value instead of MCs in our HIA. Details of this HIA could be found in Ref. 19 and Ref. 20. Since that Liu's HIA is

similar to our algorithm proposed in this paper, and that it has been verified to have the better performance than other heuristic algorithm (e.g. advance Tabu search, SA) when being used to solve the stochastic optimization problem²⁰, necessarily, we could just compare our HIA's performances than that of Liu's.

To validate the computational efficacy of the procedure, we compared our algorithm against Liu's HIA using a real transportation network shown in Fig.2.

This real transportation network is taken from the street network of New York City, USA. The area which this network is included has a ZIP Code ranges from 10001 to 10036. In this street network, there are 790 number of nodes, 1,473 number of arcs, 20 number of origin-destination nodes which consist of $20 \times 20 = 400$ Origin-Destination pairs. For convenience, all the components of the transportation time vector are set as 100 unit times. That is, set $\mathbf{T} = [100, 100, \dots, 100]$. At the same time, set all the components of the transportation time reliability constraint vector as 0.9, i.e., let $\gamma = [0.90, 0.90, \dots, 0.90]$.

The computing times of these two HIAs for network capacity calculation were obtained by changing a variety of parameters. The results of this computational experience are shown in Table 2.

For this real commodity network, it can be observed that for small size problems tested, the computational time for the three procedures was relatively the same. However, as the problem size become relatively large, the HIA algorithm developed in this paper determining the network capacity much more rapidly in terms of the computational time than Liu's HIA algorithms. The reason is that the back propagation neural network has too many drawbacks such as easily being trapped into a local minima. We change Liu's solution technique via using MCs to approximate the uncertain functions so that the search capacity of the algorithm is enhanced.

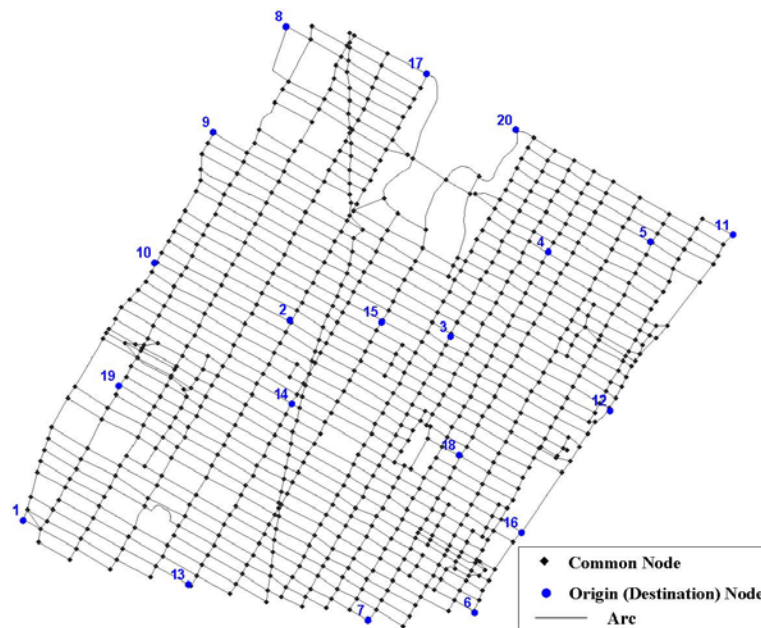


Fig. 2 The real road network for comparative computational time analysis

Table 2. Computational times of HIAs in a real transportation network

The real transportation network		
No. of common nodes		790
No. of origin-destination nodes		20
No. of OD pairs		400
No. of arcs		1473
The maximal practical capacity of each arc $a \in A$, C_a^{\max}		1.0
The minimal practical capacity of each arc $a \in A$, C_a^{\min}		0.0
Parameters in GAs of HIAs		
Cycles in <i>Reliability Check</i> (v),		5,000
Population size		30
Generations		2000
Crossover probability		0.1
Mutation probability		0.3
Computing time of HIAS for NCTTRC		
For determining transportation capacity of a sub-network with one Origin-Destination pair between which there is one path	Algorithm ①	30.54 sec
	Algorithm ②	36.10 sec
For determining transportation capacity of a sub-network two Origin-Destination pairs, between each Origin-Destination pair there are five paths	Algorithm ①	350.61 sec
	Algorithm ②	389.95 sec
For determining transportation capacity of a sub-network five Origin-Destination pairs, between each Origin-Destination pair there are five paths	Algorithm ①	800.03 sec
	Algorithm ②	1231.80 sec
For determining transportation capacity of a sub-network ten Origin-Destination pairs, between each Origin-Destination pair there are ten paths	Algorithm ①	3569.48sec
	Algorithm ②	5688.22 sec
For determining transportation capacity of a sub-network twenty Origin-Destination pairs, between each Origin-Destination pair there are ten paths	Algorithm ①	8002.33 sec
	Algorithm ②	13470.89 sec

Algorithm ①=Our HIA Algorithm; Algorithm ②=Liu's HIA Algorithm²⁰

4. Conclusions

The focus of this paper is to develop a mathematical program to determine the maximal total throughput, that is the capacity of a transportation network and to study the impact of uncertainties on network capacity. The network capacity is defined as the maximum commodities which will be transported across every origin-destination pair in the network. A solution technique embedded genetic algorithm with Monte Carlo simulation is used to determine the capacity of a single-commodity transportation network. The computational time of the proposed algorithm has been compared with that of the previous work and finds that the performance of the algorithm in this paper is more or less improved.

The second major contribution of this paper is to systematically study the impact of uncertainty of arc capacity in the single-commodity network. One should note that even though the transportation capacity will vary depending on the network configuration and demand levels, the fact that uncertain arc capacity occurs is valid for most networks. However such a situation has not been mathematically considered into the transportation network capacity determination.

This algorithm is general and is applicable to any variation of the problem studied in this paper. The sampling based strategy can be easily extended to handle transportation time uncertainty also. This work could be extended in multiple directions. There is need for accounting for level of service constraints in the arcs in the network and to account for variation in arc performance based on flow using link performance functions²². In practice, we would look more closely at the following directions:

- (i) Network capacity computation with other constraints such as level of service or capacity robustness.
- (ii) Consideration of other stochastic events such as transportation accidents during determining the network capacity.
- (iii) Compatibility between the increase and the use of the network capacity.

Acknowledgement

This research is supported by the Fundamental Research Funds for the Central Universities (Granted No. 2010121074), the National Natural Science Foundation of China (Granted No. 40801158 & No. 41071101), and Natural Science Foundation of Fujian Province (Granted No. 2010J05099)

References

1. J.J. Wu, H.J. Sun and Z.Y. Gao, Cascading failures on weighted urban traffic equilibrium networks, *Phys. A* 386 (2007) 407–413.
2. W. Wang, Y. Mao, J. Jing, X. Wang, H. Guo, X. Ren, and I. Katsushi, Driver's various information process and multi-ruled decision-making mechanism: a fundamental of intelligent driving shaping model, *International Journal of Computational Intelligence Systems*, 4 (2011) 297–305.
3. Y. Zhang, Network structure and capacity requirement: The case of China, *Transport. Res. E* 46 (2010) 189–197.
4. S. Kim and R. Uzsoy, Heuristics for capacity planning problems with congestion, *Comput. & Oper. Res.* 36 (2009) 1924–1934.
5. W. Wang, H. Guo, K. Ikeuchi and H. Bubbs, Numerical simulation and analysis procedure for digital driving dependability in intelligent transport system, *KSCE J. Ci. Eng.*, 15(2011) 891–898.
6. A. Chen, H. Yang, H.K. Lo and W. Tang, A capacity related reliability for transportation networks, *J. Adv. Transp.* 33(2) (1999) 183–200.
7. K. Higginbotham (eds.), *American Shipper's TMS Benchmark Report: Status Quo — At What Cost?* American Shipper, (Hames Blaeser, Publisher, 2008).
8. H. Yang, M.G.H. Bell and Q. Meng, Modeling the capacity and level of service of urban transportation networks, *Transport. Res. B* 34 (2000) 255–275.
9. A. Chen, H. Yang, H.K. Lo and W.H. Tang, Capacity reliability of road network: an assessment methodology and numerical results, *Transport. Res. B* 36 (2002) 225–252.
10. R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows: Theory, Algorithms and Applications*, (Prentice Hall, Englewood Cliffs, NJ. 1993).
11. Y. K. Lin, A simple algorithm for reliability evaluation of a stochastic -flow network with node failure, *Comput. Oper. Res.* 28 (2001) 1277–1285.
12. C. C. Jane, J. S. Lin and J. Yuan, On reliability evaluation of a limited-flow network in terms of minimal cutsets, *IEEE Trans. Reliab.* 42 (1993) 354–361.
13. Y. K. Lin, Study on the Multicommodity Reliability of A Capacitated-Flow Network, *Comput. & Math. with Appl.* 42 (2001) 255–264.

14. P. Doulliez and J. Jamoulle, Transportation networks with random arc capacities, *Oper. Res.* 3 (1972) 45–60.
15. W. Wang, W. Zhang, H. Guo, H. Bubb and Katsushi Ikeuchi, A safety-based behavioural approaching model with various driving characteristics, *Transportation Research Part C-Emerging Technologies*, accepted article, doi:10.1016/j.trc.2011.02.002. .
16. S.S. Chaudhry, M.W. Varano and L. Xu. Systems Research, Genetic Algorithms and Information Systems, *Syst. Res. & Behav. Sci.* 17(2000) 149–162.
17. A. Fasih, J. C. Chedjou and K. Kyamakya, Cellular Neural Networks-Based Genetic Algorithm for Optimizing the Behavior of an Unstructured Robot, *International Journal of Computational Intelligence Systems*, 2(2) (2009) 124–131.
18. N. Castin, L. Malerba and R. P. Domingos, Use of computational intelligence for the prediction of vacancy migration energies in atomistic kinetic Monte Carlo simulations, *International Journal of Computational Intelligence Systems*, 1(4) (2008) 340–352.
19. B. Liu, *Theory and Practice of Uncertain Programming*. (Physica-Verlag, Heidelberg, 2002).
20. B. Liu, *Uncertainty theory: An introduction to its axiomatic foundations*. (Springer-Verlag, Berlin, 2004).
21. <http://www.orsc.edu.cn/~liu/resources.htm>, (1st, Aug., 2010).
22. A. Unnikrishnan and S. T. Waller, Freight Transportation Network Design Problem for Maximizing Throughput under Uncertainty, in CD-ROM of Annual Meeting Transportation Research Board, Washington D.C. USA (2009).