Node Dependency in Multi-Commodity Flow Problem with Application to Transportation Networks

Weibin Dai, Jun Zhang, Xiaoqian Sun*, Sebastian Wandelt
School of Electronic and Information Engineering, Beihang University,
No. 37 Xueyuan Road, 100191 Beijing, China

Abstract

In this research, we study the Multi-Commodity Flow Problem (MCFP) in the context of air transportation systems. MCFP deals with assigning a variety of goods to flow from sources to their destinations in a network. While many optimization problems in transportation networks can be formulated as classic MCFP, previous research mostly considered the edge capacity as a network flow constraint. Based on traditional path-flow model and edge-flow model, this research proposes new modifications with the consideration of node capacity in the network. In addition, we implement new optimization heuristics improving the path-finding stage of the algorithm. These optimizations allow us to solve the MCFP for networks with around one hundred nodes. Based on these results, we define and compute the node-dependency relationship in MCFP networks. For preliminary evaluation, our novel techniques are evaluated on an air transportation network consisting of 164 nodes. The experiment showed that dependencies on a node are the results of joint influence of the network structure factors and flows. Moreover, the dependencies in our network come in geographical clusters.

Keywords: multi-commodity flow problem, path-finding, dependency

I. Introduction

The multi-commodity flow problem (MCFP) deals with the assignment of commodity flows from sources to destinations in a network. If the commodities don't influence each other, then the problem can be solved by solving each single-commodity problem. However, in practice, commodities often share the same set of arcs and nodes. For example, in air transportation, passengers with different origin-destination pairs might share the same physical flight. Thus, solving the single-commodity problem is not an option, if we are interested in optimal results[1].

MCFP has been studied by a number of researchers with a variety of methods, such as column generation, Lagrangian relaxation, and Dantzig-Wolfe decomposition. Tomlin (1966) first proposed the column generation approach which is one of the frequently-used methods [2]. He was also one of the early users of the general Dantzig-Wolfe decomposition approach. In the next half a century, a number of new algorithms based on column generation idea and Dantzig-Wolfe decomposition have been presented. The branch-and-bound approach based on column generation can be used to solve MCFP. Further, Barnhart et al. (2000) proposed a modified version for origin-destination integer multi-commodity flow problems [3]. They improved the algorithm for the path-flow model by presenting a new branching rule and adding cuts. The cuts can observably reduce the computational complexity in majority situation. The Dantzig-Wolfe decomposition has also been improved by researchers. Karakostas (2008) proposed polynomial approximation approaches based on Dantzig-Wolfe decomposition for MCFP [4]. The schemes were based on some previous algorithms [5][6] and the computation time depended least on the number of commodities. Models based on paths were usually used while solving MCFP in the past. However, Pierre-Olivier et al. (2013,2015) presented a point that path flows could be represented by other variables. In order to solve the maximum concurrent flow problem that is a portion of MCFP, they proposed a generic aggregate model based on column generation [7][8]. Instead of generating paths for each commodity, they generate groups of paths in appropriate ways. They introduced 3 ways in their paper that the commodities can be aggregated into trees, into a single set, or into several trees. After comparing with primary models, the generic aggregate model was shown to be good in computing time. Based on previous research, MCFP can also be modified, such as the bi-objective problem. In order to solve this problem, Moradi et al. (2015) proposed a column generation approach [9]. The algorithm was a combination of simplex method and Dantzig-Wolfe decomposition. It was shown that the average computation time doesn't necessarily increase with increasing the number of commodities. In addition to the research above, there are also researchers studying the application, such as [10][11][12][13][14][15]. In our work, based on the column generation approach, we present new modifications of the path-flow model and show how this leads towards effective quantification of node dependencies in networks.

In this paper, two modifications of previous model for solving MCFP are proposed. Most previous research on MCFP only considered the traffic capacity on edge as one network flow constraint. In the real-world transportation networks, the number of passengers who can get through a node within a certain time is limited. We incorporate this information into our model. Moreover, new algorithms for reducing computation time are also implemented. Based on these algorithms, we are interested in studying the importance of nodes in multi-commodity flow networks. This importance can be assessed by the effect of one node's absence on the whole network. Similarly, the effect of one node's absence on one other node can be used to denote the dependency of the second node on the first one. Node dependencies have been studied by several researchers. A portion of previous research was based on a structure of the node's local surrounding [16]. However, Kenett et al. (2012) proposed a new method to analyse the activity and topology dependencies between nodes in directed networks [17]. The approach was able to show hidden properties on the network structure. Despite all this, the node dependencies based on network flows have not been studied in previous research. Therefore, dependencies between nodes that denotes effect of one node on the passenger flows from the other node in MCFP networks is obtained in this paper.

This paper is organized as follows. The problem description and modifications of two models are proposed in Section II. Two improved algorithms for finding paths are presented in Section III. Dependencies between nodes and node groups in MCFP are quantified in Section IV. In Section V, an air transportation network including 164 nodes and over 4000 edges is chosen as a case study. This paper concludes with Section VI.

II. Problem description and model modifications

In this section, MCFP is described and two modifications based on previous models [10] are presented.

A. Problem description

We introduce the MCFP in the context of passenger transportation below. Let $G = (V, E)$ be a directed graph with n nodes and m edges, where V and E are the sets of nodes and edges, respectively. Each edge (i, j) has a length of c_{ij} and a capacity of cap_{ij}. Now, passengers need to be transported from origin nodes to their destination nodes. Let the passengers with the same origin and same destination be in a group and K be the set of these groups. For each group $k \in K$, $o(k)$ and $d(k)$ represent the origin and destination of k, respectively. d^k is the travel demand (i.e. the passenger number) of k. Now, we need to find the minimum value of the total path length for all passengers, satisfying all travel demand in set K. The problem above is the traditional MCFP; it only considers the edge capacity as a network flow constraint. In this paper, the node capacity is considered as a new restriction. Assume the capacity of node i is $capn_i$. To solve the new problem, we improve the old model [10] and then get the new 'path-flow model' and 'edge-flow model'.

B. Path-flow model

The path-flow model is based on an obvious fact that every group k needs at least one path from its origin to destination, and these paths consist of a series of edges laid end to end. For each group k, we assume that H^k is the set of paths from the origin to destination of k, and f_h^k is the travel flow on the path $h \in H^k$. Then we have the path-flow model:

$$minimize\ z(f) = \sum_{(i,\ j)\in E} \sum_{k\in K} \sum_{h\in H^k} f_h^k \delta_{ij}^h c_{ij} \tag{1}$$

$$subject\ to \quad \sum_{h\in H^k} f_h^k = d^k,\ \forall k \in K \tag{2}$$

$$\sum_{k\in K} \sum_{h\in H^k} f_h^k \delta_{ij}^h \le cap_{ij},\ \forall (i,j) \in E \tag{3}$$

$$\sum_{k\in K} \sum_{h\in H^k} f_h^k \widetilde{\delta}_i^h \le capn_i,\ \forall i \in V \tag{4}$$

$$f_h^k \ge 0,\ \forall k \in K,\ \forall h \in H^k \tag{5}$$

where $f = [..., f_h^k, ...]_{h\in H^k,\ k\in K}$ is the flow vector on all the paths.

The total path length of all passengers is represented in equation (1). The equation (2)-(5) represent travel demand constraint, edge capacity constraint, node capacity constraint, and non-negative constraint, separately. $\Delta = [\delta_{ij}^h]$ in equation (3) and $\widetilde{\Delta} = [\widetilde{\delta}_i^h]$ in equation (4) are the incidence matrixes between paths and edges as well as paths and nodes. They can be defined as the follows:

$$\delta_{ij}^h = \begin{cases} 1,\ if\ edge\ (i,j)\ is\ in\ path\ h \\ 0,\ otherwise \end{cases} \tag{6}$$

$$\widetilde{\delta}_i^h = \begin{cases} 1,\ if\ node\ i\ is\ in\ path\ h \\ 0,\ otherwise \end{cases} \tag{7}$$

With the model above, MCFP can be solved generally. However, there may be a case that the travel demand in equation (2) can not be completely satisfied due to the edge or node capacity. At this time, the maximum number of passengers that can be transported should be computed first. Assume the number is n_m. Then the MCFP can be solved by adding the constraint (8) and replacing $'='$ with $'\le'$ in (2). In addition, new algorithms for finding paths is proposed in Section V-A. It can reduce path-finding time significantly.

$$\sum_{k\in K} \sum_{h\in H^k} f_h^k = n_m \tag{8}$$

C. Edge-flow model

For each group k, the travel flow on edge (i,j) is represented by g_{ij}^k, so we have:

$$minimize\ z(g) = \sum_{(i,j)\in E} \sum_{k\in K} g_{ij}^k c_{ij} \tag{9}$$

$$subject\ to \quad \sum_{j\in N_i^+} g_{ij}^k - \sum_{j\in N_i^-} g_{ji}^k = b_i^k,\ \forall i \in V,\ \forall k \in K \tag{10}$$

$$\sum_{k\in K} g_{ij}^k \le cap_{ij},\ \forall (i,j) \in E \tag{11}$$

$$\sum_{k\in K} (\sum_{j\in N_i^+} g_{ij}^k + \sum_{j\in N_i^-} g_{ji}^k) \le capn_i,\ \forall i \in V \tag{12}$$

$$g_{ij}^k \ge 0,\ \forall k \in K,\ \forall (i,j) \in E \tag{13}$$

where $g = [..., g_{ij}^k, ...]_{(i,j)\in E,\ k\in K}$. is the flow vector on all the edges.

In a same way, equation (9) represents the total length for all passengers, and the equation (10)-(13) represent node flow constraint, edge capacity constraint, node capacity constraint, and non-negative constraint, separately. $N_i^+ = \{j \in V : \exists(i,\ j) \in E\},\ N_i^- = \{j \in V : \exists(j,\ i) \in E\}$. b_i^k in equation (10) represents the change number of remained passengers on node i, and it can be defined as follows:

$$b_i^k = \begin{cases} d^k, & if\ i\ is\ the\ origin\ of\ k \\ -d^k, & if\ i\ is\ the\ destination\ of\ k \\ 0, & otherwise \end{cases} \tag{14}$$

D. *Comparison of two models*

After comparing the formulations of the two models, it is shown that there is a significant difference between them. In a given MCFP, the dimension of variable g in edge-flow model is constant and it is equal to the product of edge number and group number. However, the dimension of variable f is uncertain. It is the number of selected paths. It is able to take all the paths into account if the network size is small. However, the path number can increase rapidly as the growth of the network scale. This indicates that the computational complexity of the model can be very large. There exists the same trouble in the edge-flow model.

In this case, the advantages of the path-flow model come in handy. The complexity of the problem can be kept outside the optimization. The path set is also allowed for scaling. It does not need to take all the paths into account before the optimization. If we can find an appropriate subset of paths and solve the problem with it, the computational complexity of the problem can be reduced greatly [10]. Although the solution found with this method may be not optimal in general, it can be acceptable in value.

III. Heuristic for path-finding

The 'path-flow' model is used in this study, and the first step during computation is path-finding. For each group k, it is necessary to construct a set of available paths from its origin to its destination. However, because of large numbers of nodes and edges, the number of paths can be very large. A naive implementation at this stage will significantly limit the applicability to real world networks. The advantage of path-flow model, proposed in Section II-D, can be nearly seen here. Instead of finding all the paths for each passenger group k, we just need to construct a *promising subset* of paths. The optimization problem is then solved on this subset. The goal of MCFP is to get the minimum total path length solution. Therefore, short paths are more likely to be used in the optimal solution. The passenger flows on long paths are often zero anyways; e.g., air passengers having tickets from origin to destination with more than three stopover are very uncommon.

Therefore, in the path-finding stage, the constraints of the path length and edge number are considered in the algorithm. The path length for each group k should be less than α times of the shortest path length. $\alpha > 1$ is a coefficient. The edge number of each path must not exceed a given number N_s.

As shown in Fig. 1, dis is the distance matrix of the network and $next\_nodes(k)$ is a set of nodes that have edges from the last node of $path(k)$.

The algorithm can be simplified further because the path number for some groups can be still a lot after selection by the algorithm above (For example, in Section V, when we set $\alpha = 1.5$ and $N_s = 3$, there are hundreds or over one thousand feasible paths for almost each group k.). Thus, after the selection by edge number constraint, a certain number of shortest paths can be chosen to the path subset directly. This number is defined as N_p.

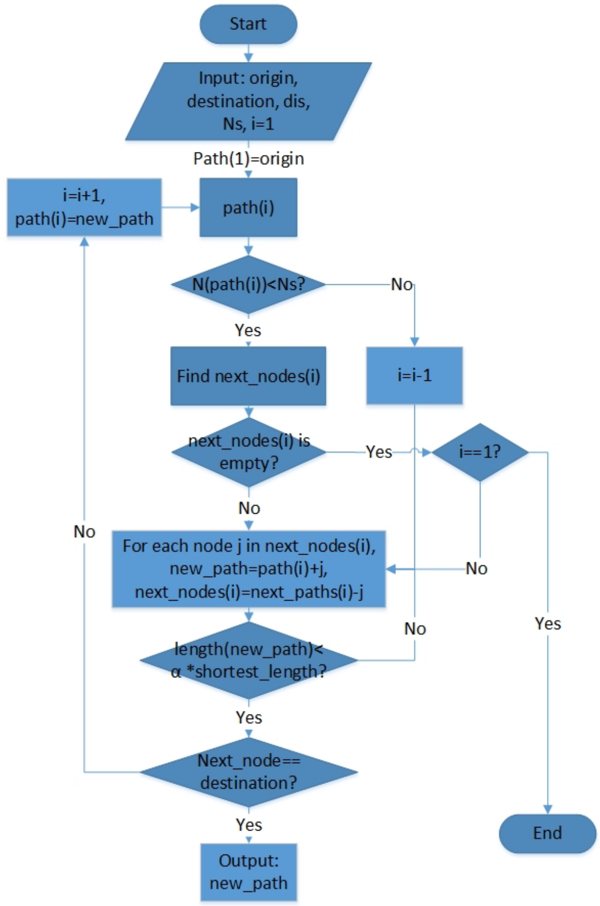


Figure 1. path-finding algorithem

We name the two algorithms above after α-algorithm and N_p-algorithm, separately. With these two algorithms, we can find the subset of all available paths for each passenger group k. The result got with this subset may not be the theoretically optimal solution. However, the flows on most paths of optimal solution are zero, and what we choose are the short paths (in term of length and edge number), so the result with smaller path set can be close to the theoretically optimal solution if the parameters α and N_p have appropriate values.

IV. Node-dependency

Dependencies between nodes have been studies by several researchers. However, most of them focused on the structure or topology of networks. Therefore, dependencies based on network flows are studied in this section. Dependency between nodes denotes effect of one node on the passenger flow from the other node. Dependency between node groups is similar. Interestingly, dependency of a node on the all-node set can show this node's importance to the whole network flow. In this section, we will introduce the dependency between nodes and node groups in MCFP networks.

A. *Model modification*

When we consider the dependency of node s on node j in MCFP, the basic method is to study the effect of absence of node j on the passenger travel situation from node s. Thus, we must get the optimal assignments which transport the most number of passengers from node s to their destinations, within the shortest path length in the case that node j is present or absent.

Therefore, we just need to consider the passengers sending out from node s. Let K^s represent the set of passenger groups whose origin is node s, then the path-flow model can be modified as follows:

$$minimize\ z(f) = \sum_{(i,\ j)\in E} \sum_{k\in K^s} \sum_{h\in H^k} f_h^k \delta_{ij}^h c_{ij} \tag{15}$$

$$subject\ to \quad \sum_{h\in H^k} f_h^k = d^k,\ \forall k \in K^s \tag{16}$$

$$\sum_{k\in K^s} \sum_{h\in H^k} f_h^k \delta_{ij}^h \leq cap_{ij},\ \forall (i,\ j) \in E \tag{17}$$

$$\sum_{k\in K^s} \sum_{h\in H^k} f_h^k \widetilde{\delta}_i^h \leq capn_i,\ \forall i \in V \tag{18}$$

$$f_h^k \geq 0,\ \forall k \in K^s,\ \forall h \in H^k \tag{19}$$

B. Quantification of node dependency

In the model above, $h \in H^k$ is a path connecting node s and destination node of k. The value of f_h^k is the flow on this path. Let L_h^k be the length of path h. We set two parameters $DP(s|j^+)$ and $DP(s|j^-)$ that can denote the situation of passenger travel when the node j is present and absent, separately. They can be formulated as follows:

$$DP(s|j^+) = \sum_{k\in K^s} \sum_{h\in H^k} \frac{f_h^k(j^+)}{L_h^k} \tag{20}$$

$$DP(s|j^-) = \sum_{k\in K^s} \sum_{h\in H^k} \frac{f_h^k(j^-)}{L_h^k} \tag{21}$$

where $f_h^k(j^+)$ and $f_h^k(j^-)$ are the passenger flows along path h when node j is present and absent, separately. The values of $f_h^k(j^+)$ can be found in Section IV-A, and we can get $f_h^k(j^-)$ in Section IV-C.

The aim of our model is to transport the most number of passengers within the shortest path length, so for each path h with non-zero passenger flow, we sum up the ratios of f_h^k and L_h^k. Thus, we have the dependency of node s on node j:

$$D(s|j) = DP(s|j^+) - DP(s|j^-) \tag{22}$$

Note that the numbers of passengers starting from different origins may be very different, which results in differences of the value of the dependency of each node on itself. Obviously, for each node, the dependency on itself is the largest. So, we can standardize $D(s|j)$:

$$d(s,j) = \frac{D(s|j)}{\underset{j\in V}{max}\, D(s|j)} = \frac{D(s|j)}{D(s|s)} \tag{23}$$

Thus, $d(s,j)$ is the dependency of node s on node j, and the range of its value is $[0,1]$.

C. The method with absence of node j

The path-flow model may have no feasible solution if the node j is absent, i. e. the travel demand in equation (16) may not be satisfied completely. Thus, we should calculate the maximum number of passengers starting from node s who can be transported to their destinations,

$$maximize\ n_j(f) = \sum_{k\in K^s}\sum_{h\in H} f_h^k \tag{24}$$

$$subject\ to \quad \sum_{h\in H^k} f_h^k \le d^k,\ \forall k \in K^s \tag{25}$$

$$\sum_{k\in K^s}\sum_{h\in H^k} f_h^k \delta_{il}^h \le cap_{il},\ \forall (i,l) \in E \tag{26}$$

$$\sum_{k\in K^s}\sum_{h\in H^k} f_h^k \widetilde{\delta}_i^h \le capn_i,\ \forall i \in V \tag{27}$$

$$f_h^k \ge 0,\ \forall k \in K^s,\ \forall h \in H^k \tag{28}$$

$$f_h^k = 0,\ if\ j \in h,\ \forall h \in H^k,\ \forall k \in K^s \tag{29}$$

Assume that the value of the optimal solution $n_j(f)$ obtained from the algorithm above is n_m, then we solve the minimum total path length problem:

$$minimize\ z(f) = \sum_{(i,l)\in E}\sum_{k\in K^s}\sum_{h\in H^k} f_h^k \delta_{il}^h c_{il} \tag{30}$$

$$subject\ to \quad (25)(26)(27)(28)(29)(32) \tag{31}$$

$$\sum_{k\in K^s}\sum_{h\in H} f_h^k = n_m \tag{32}$$

The solution $f_h^k(h \in H^k,\ k \in K^s)$ of this programme is just the $f_h^k(j^-)$ required in Section IV-B.

D. Dependency between node groups

Similarly with the algorithm above, the dependency of node group S on node group J in MCFP is related to the effect of absence of all nodes in J on the passenger travel situation starting from nodes in S. Thus, we define $DP(S|J^+)$ and $DP(S|J^-)$ as follows:

$$DP(S|J^+) = \sum_{k\in K^S}\sum_{h\in H^k} \frac{f_h^k(J^+)}{L_h^k} \tag{33}$$

$$DP(S|J^-) = \sum_{k\in K^S}\sum_{h\in H^k} \frac{f_h^k(J^-)}{L_h^k} \tag{34}$$

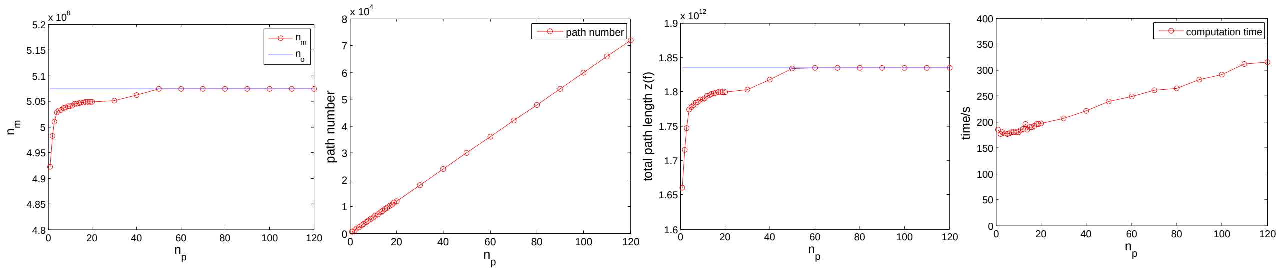
where $f_h^k(J^+)$ and $f_h^k(J^-)$ are the passenger flows along the path h when all nodes in J are present and absent, separately. Their values can be obtained similarly with $f_h^k(j^+)$ and $f_h^k(j^-)$ in Section IV-A and Section IV-C. Then, we have

$$D(S|J) = DP(S|J^+) - DP(S|J^-) \tag{35}$$

After standardizing $D(S|J)$, and we get the dependency of node group S on node group J:

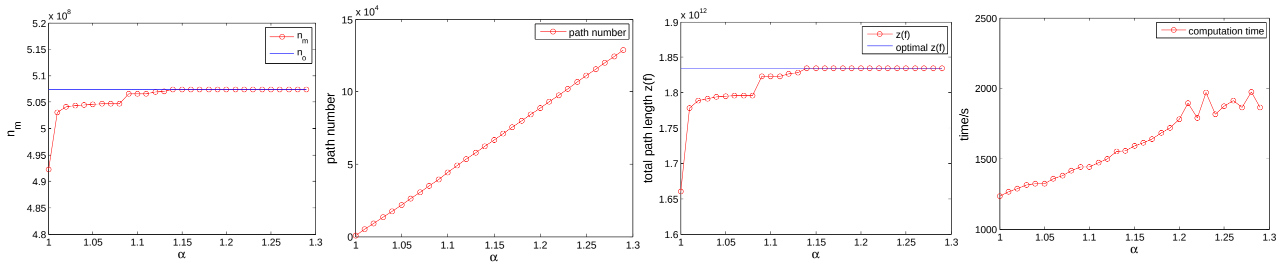
$$d(S,J) = \frac{D(S|J)}{\max\limits_{J\subset V} D(S|J)} = \frac{D(S|J)}{D(S|S)} \tag{36}$$

The range of its value is $[0,1]$.



(a) n_m and n_o (b) path number (c) total path length $z(f)$ (d) computation time

Figure 2. MCFP results for 599 groups I: The change of n_m, path number, $z(f)$, and computation time while N_p increases from 1 to 120.



(a) n_m and n_o (b) path number (c) total path length (d) computation time

Figure 3. MCFP results for 599 groups II: The change of n_m, path number, $z(f)$, and computation time while α increases from 1 to 1.29.

V. Evaluation

A. Description of the experiment

Our model is evaluated on an air transportation network as a case study. We select an air transportation network consisting of 164 countries in the world. We simplify each countries to a node at its centre and assume that all the passengers start from the centre of one country to the centre of another country. Thus, we can construct a network with 164 nodes. Flights between them are represented by 4496 edges. In total, there are 13420 groups of passengers between these nodes.

TABLE I
25 SELECTED NODES

US: United States	GB: United Kingdom	DE: Germany	ES: Spain	CN: China
FR: France	IT: Italy	JP: Japan	AE: United Emirates	TH: Thailand
RU: Russian	CA: Canada	IN: India	KR: South Korea	TR: Turkey
SG: Singapore	CH: Switzerland	NL: Netherlands	MY: Malaysia	TW: Taiwan
SA: Saudi Arabia	ID: Indonesia	MX: Mexico	AU:Australia	BE: Belgium

MCFP in this air transportation network can be solved with the path-flow model. However, note that the number of all groups 13420 is large. In order to obtain appropriate values of α and N_p for this network, a sub-problem with a subset of passenger groups can be solved first. The subset consists of 599 passenger groups between 25 'busiest' nodes. The 25 nodes are shown in Table I. After solving the sub-problem, the primal problem with 13420 groups can be computed.

B. The results and analysis

1) MCFP solution: With matlab, the MCFP solution can be computed. In this section, $N_s = 3$ is fixed and results with different values of α and N_p will be discussed.

Results for 599 groups are shown in Figure 2 and Figure 3. Different values of N_p and α are selected. In Figure 2, N_p shortest paths are chosen for each passenger group. In Figure 3, path

TABLE II
MCFP RESULTS FOR 13420 GROUPS

N_p	n_m	n_o	$z(f)$	path number	time/s
1	936,936,526	991,083,149	2.9252E+12	13,420	3640.9
2	958,070,402	991,083,149	3.0807E+12	26,834	3748.8
3	966,342,867	991,083,149	3.1510E+12	40,248	3890.8
4	970,031,210	991,083,149	3.1876E+12	53,662	4128.3
5	973,611,421	991,083,149	3.2219E+12	67,076	4268.9
6	975,346,709	991,083,149	3.2394E+12	80,486	4652.6
7	976,971,015	991,083,149	3.2527E+12	93,890	4974.8
8	977,874,862	991,083,149	3.2613E+12	107,290	5186.1
9	978,742,266	991,083,149	3.2685E+12	120,690	5115.0
10	979,410,264	991,083,149	3.2738E+12	134,084	5348.4

length constraint $length(path) \leq \alpha * length(shortestpath)$ is also satisfied. n_m and n_o are the actual transportation number and total travel demand, respectively. $z(f)$ is the optimal total path length.

It is shown in Figure 2 that $n_m/n_o = 97\%$ of travel demand can be satisfied when $N_p = 1$. It indicates that most of passengers in these 599 groups can be transported with the 599 shortest paths. The actual transportation number n_m increases slowly with increase of N_p.

Note that $n_m = n_o = 507,425,672$ when $N_p = 60$ (When $N_p = 50$, $n_m = 507,380,143 < n_o$, although it seems they are very close in the figure.). It indicates all passengers can be transported in this case. Then, when $N_p = 70$, the value of $z(f)$ decreases. It shows that extra 10 paths for each group are used to get better assignment instead of to transport more passengers before. When $N_p \geq 70$, the optimal total path length $z(f) = 1.8343E + 12$ is constant (When $N_p = 60$, $z(f) = 1.8344E + 12$). It has the same property in Figure 3 when $\alpha \geq 1.14$.

All travel demand can be satisfied and the optimal solution can be obtained with both two algorithms. However, there is a wide difference between their computation time (about 10 times!). That is because it must do a comparation between the shortest paths and each generated path in the α-algoritm and it spends much time. Therefore, N_p-algorithm should be selected for solving the primal problem with 13420 groups

Results of the primal problem are shown in Table II. With N_p increasing from 1 to 10, over 98.8% passengers can be transported to their destinations. Computation time also increases with N_p.

2) Dependency between nodes: With the algorithm in Section IV, the dependencies between any two nodes in this MCFP network with 13420 groups of passengers can be obtained. As shown in Table III, the number in ith row and jth column is the dependency of node i on node j.

On the one hand, note that for one node i, dependencies on most of other 24 nodes are much less than 1. It is also shown that these nodes with little influence on i are almost all far from node i. In other words, nodes with significant influence on i are close to node i in general. For example, there are only two numbers lager than 0.1 in the first row: $d(US, CA) = 0.37$ and $d(US, MX) = 0.21$. From Table I, we can see that node US is United States and node CA, MX are Canada, Mexico, respectively. Node CA and MX are the only two nodes close to node US. This 'close effect' is also applied to other areas. In Western Europe, there are significant dependencies between node GB (United Kingdom), DE (Germany), ES (Spain), FR (France) and IT (Italy). In East Asia, node CN (China), JP (Japan), KR (South Korea) and TW (Taiwan Area) have large influence on each other as well.

On the other hand, there exists a case that one node i is close to some nodes but its dependencies on these nodes are tiny. For example, node GB (United Kingdom), DE (Germany), FR (France)

TABLE III
DEPENDENCIES BETWEEN 25 NODES

node	US	GB	DE	ES	CN	FR	IT	JP	AE	TH	RU	CA	IN	KR	TR	SG	CH	NL	MY	TW	SA	ID	MX	AU	BE
US		0.04	0.02	0.01	0.01	0.02	0.01	0.02	0.00	0.00	0.00	**0.37**	0.01	0.01	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.00	**0.21**	0.00	0.00
GB	0.02		**0.12**	**0.15**	0.00	**0.1**	**0.05**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	**0.05**	**0.11**	0.00	0.00	0.00	0.00	0.00	0.00	0.02
DE	0.01	**0.15**		**0.13**	0.00	**0.07**	**0.08**	0.00	0.00	0.00	0.01	0.00	0.00	0.00	**0.05**	0.00	**0.09**	**0.05**	0.00	0.00	0.00	0.00	0.00	0.00	0.03
ES	0.00	**0.24**	**0.17**		0.00	**0.11**	**0.08**	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.04	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.04
CN	0.01	0.01	0.01	0		0.00	0.00	**0.12**	0.00	**0.1**	0.01	0.00	0.01	**0.22**	0.00	0.04	0.00	0.00	0.03	**0.31**	0.00	0.02	0.00	0.01	0.00
FR	0.01	**0.17**	**0.1**	**0.12**	0.00		**0.13**	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	**0.09**	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.04
IT	0.01	**0.12**	**0.15**	**0.11**	0.00	**0.15**		0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.01	0.00	**0.05**	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.04
JP	0.03	0.00	0.00	0.00	**0.19**	0.00	0.00		0.00	0.03	0.00	0.00	0.00	**0.51**	0.00	0.01	0.00	0.00	0.01	**0.13**	0.00	0.01	0.00	0.01	0.00
AE	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00		0.00	0.01	0.00	**0.11**	0.00	0.01	0.00	0.00	0.00	0.00	0.00	**0.12**	0.00	0.00	0.00	0.00
TH	0.00	0.01	0.01	0.00	**0.21**	0.00	0.00	0.04	0.01		0.01	0.00	0.04	0.04	0.00	**0.15**	0.00	0.00	**0.08**	0.02	0.00	0.02	0.00	0.01	0.00
RU	0.01	0.02	**0.07**	0.04	0.02	0.02	0.04	0.00	0.02	0.01		0.00	0.00	0.00	**0.05**	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
CA	**0.82**	0.02	0.01	0.00	0.01	0.01	0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00
IN	0.02	0.03	0.01	0.00	0.03	0.01	0.01	0.01	**0.3**	**0.07**	0.01	0.00		0.00	0.00	**0.05**	0.00	0.00	0.03	0.00	**0.09**	0.00	0.00	0.01	0.00
KR	0.01	0.00	0.00	0.00	**0.35**	0.00	0.00	**0.47**	0.00	0.03	0.00	0.00	0.00		0.00	0.01	0.00	0.00	0.01	0.04	0.00	0.01	0.00	0.00	0.00
TR	0.00	0.04	**0.23**	0.01	0.00	0.02	0.02	0.00	0.01	0.00	0.03	0.00	0.00	0.00		0.00	0.02	0.02	0.00	0.00	0.02	0.00	0.00	0.00	0.01
SG	0.00	0.00	0.00	0.00	**0.1**	0.00	0.00	0.02	0.00	**0.15**	0.00	0.00	0.03	0.01	0.00		0.00	0.00	**0.27**	0.02	0.00	**0.2**	0.00	0.02	0.00
CH	0.01	**0.15**	**0.22**	**0.08**	0.00	**0.15**	**0.07**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00		0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.03
NL	0.01	**0.36**	**0.13**	**0.09**	0.00	**0.06**	**0.06**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	**0.05**		0.00	0.00	0.00	0.00	0.00	0.00	0.01
MY	0.00	0.00	0.00	0.00	**0.08**	0.00	0.00	0.01	0.00	**0.09**	0.00	0.00	0.02	0.01	0.00	**0.28**	0.00	0.00		0.02	0.00	**0.33**	0.00	0.02	0.00
TW	0.01	0.00	0.00	0.00	**0.65**	0.00	0.00	**0.17**	0.00	0.02	0.00	0.00	0.00	**0.05**	0.00	0.02	0.00	0.00	0.02		0.00	0.01	0.00	0.00	0.00
SA	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	**0.24**	0.00	0.00	0.00	**0.07**	0.00	0.03	0.00	0.00	0.00	0.00	0.00		0.01	0.00	0.00	0.00
ID	0.00	0.00	0.00	0.00	**0.05**	0.00	0.00	0.02	0.00	0.03	0.00	0.00	0.00	0.01	0.00	**0.26**	0.00	0.00	**0.43**	0.02	0.01		0.00	**0.05**	0.02
MX	**0.8**	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**0.08**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00		0.00	0.00
AU	0.03	0.02	0.00	0.00	**0.08**	0.00	0.00	0.03	0.01	0.04	0.00	0.00	0.02	0.01	0.00	**0.09**	0.00	0.00	**0.07**	0.01	0.00	**0.13**	0.00		0.00
BE	0.01	**0.1**	**0.13**	**0.14**	0.00	**0.13**	**0.12**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	**0.06**	0.01	0.00	0.00	0.00	0.01	0.00	0.00	

and IT (Italy) are all close to node CH (Switzerland), NL (Netherlands) and BE (Belgium), but dependencies on these nodes in the 2nd, 3rd, 6th and 7th rows of Table III are small. It is because of the passenger flow factors. This phenomenon shows that the structural factor of the network can not guarantee a considerable dependency in this multi-commodity flow network. Dependencies on one node are the result of joint influence of network structure and flows. It is an interesting finding.

We provide a visual summarization of dependencies between countries in Figure 4.

Figure 4. Dependencies between countries: An arrow is drawn from country A to country B, if the dependency of A from B is larger than 5%. The strength of the arrow is proportional to the dependency value.

3) Dependency between node groups: For further discussion, dependency between node groups can be studied. Here we let one of the node groups be the set of 25 selected nodes denoted by A and there are 3389 groups of passengers from A to all the 164 nodes. The other one is an arbitrary node in A, that is denoted by j. Dependency of set A on node j is shown in Table IV.

TABLE IV
DEPENDENCY OF SET A

node j	$d(A,j)$	node j	$d(A,j)$
US	0.1041	KR	0.0822
GB	0.2428	TR	0.0337
DE	0.2133	SG	0.0501
ES	0.1759	CH	0.0947
CN	0.1239	NL	0.0878
FR	0.1518	MY	0.0512
IT	0.1225	TW	0.0619
JP	0.0781	SA	0.0199
AE	0.0309	ID	0.0406
TH	0.0418	MX	0.0330
RU	0.0169	AU	0.0094
CA	0.0591	BE	0.0472
IN	0.0273		

Next, the number of sum passengers that start from or arrive at node j which is denoted by $p_n(j)$ is shown in Table V.

Similar to the Section V-B2, Table IV shows that dependencies on those nodes with several close neighbours and appreciable passenger flows are considerable in general. This phenomenon is obvious in some Western European nodes, such as node GB (United Kingdom), DE (Germany), ES (Spain), FR (France) and IT (Italy). These nodes occupy important positions in set A.

Another interesting thing is that node US has a close number of passengers to node GB ($p_n(US) = 96756227$ and $p_n(GB) = 96791610$), but the difference between dependencies $d(A, GB)$ and $d(A, US)$ is over twice! Go back to Table I, it shows that node US is United States and node GB is United Kingdom. In set A, node GB can be seen as a hub node and several close neighbours help it get an important position. However, node US is a spoke node and it is in the margin of the 25 nodes. This finding shows one node is important to a node set if and only if this node has considerable number of close neighbours and large passengers flows in the set.

TABLE V
SUM PASSENGERS OF EACH NODE

node	passengers	node	passengers
GB	96,791,610	TW	28,437,185
US	96,756,227	MY	26,153,812
ES	84,558,791	ID	25,799,113
CN	81,157,741	TR	25,781,874
DE	80,269,238	CH	24,152,254
IT	51,665,756	MX	23,729,680
FR	47,755,201	AE	22,975,566
JP	45,290,605	NL	22,230,311
TH	34,139,534	RU	18,920,872
KR	34,110,903	AU	18,694,758
CA	32,759,138	BE	16,317,261
SG	32,489,286	SA	13,598,676
IN	30,315,952		

VI. CONCLUSIONS

In this paper, node capacity constraints are considered in MCFP. Based on that, new modification of path-flow model and edge-flow model were proposed. In order to shorten the time for finding paths, we presented two algorithms: N_p-algorithm and α-algorithm. In addition, the node-dependency relationship in MCFP networks was studied. It was used to assess the importance of nodes or node groups. Based on the model for MCFP, the dependency parameters with network flows were presented.

In order to evaluate the algorithms, an air transportation network consisting of 164 nodes and 4496 edges was chosen as a case study. The implemented algorithms for path-finding were shown to be efficient as long as the parameters N_s and α were assigned appropriately. After comparing computation time of the two algorithms, N_p-algorithm was shown to be better than α-algorithm. In the experiment, on one hand, there were three large node clusters: Western Europe, East Asia and North America. In each of them, nodes have high dependencies on each other. It showed the influence of network structure on the dependencies. On the other hand, importance of some nodes with several close neighbour nodes was shown to be insignificant. It was because of their limited passenger flows. Therefore, in general, a node has considerable influence if and only if it has good structure position and appreciable flows. For future work, we think that our techniques can be applied for resilience analysis of multi-commodity flow networks. The importance of nodes can also be used in the stability of network flows under attack.

VII. ACKNOWLEDGEMENT

This research is partly supported by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China [Grant No. 61521091].

REFERENCES

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, "Network flows," DTIC Document, Tech. Rep., 1988.

[2] J. Tomlin, "Minimum-cost multicommodity network flows," *Operations Research*, vol. 14, no. 1, pp. 45–51, 1966.

[3] C. Barnhart, C. A. Hane, and P. H. Vance, "Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems," *Operations Research*, vol. 48, no. 2, pp. 318–326, 2000.

[4] G. Karakostas, "Faster approximation schemes for fractional multicommodity flow problems," *ACM Transactions on Algorithms (TALG)*, vol. 4, no. 1, p. 13, 2008.

[5] N. Garg and J. Konemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," 1998.

[6] L. K. Fleischer, "Approximating fractional multicommodity flow independent of the number of commodities," *SIAM Journal on Discrete Mathematics*, vol. 13, no. 4, pp. 505–520, 2000.

[7] P.-O. Bauguion, W. Ben-Ameur, and E. Gourdin, "A new model for multicommodity flow problems, and a strongly polynomial algorithm for single-source maximum concurrent flow," *Electronic Notes in Discrete Mathematics*, vol. 41, pp. 311–318, 2013.

[8] ——, "Efficient algorithms for the maximum concurrent flow problem," *Networks*, vol. 65, no. 1, pp. 56–67, 2015.

[9] S. Moradi, A. Raith, and M. Ehrgott, "A bi-objective column generation algorithm for the multi-commodity minimum cost flow problem," *European Journal of Operational Research*, vol. 244, no. 2, pp. 369–378, 2015.

[10] P. Bartodziej, U. Derigs, and M. Zils, "O&d revenue management in cargo airlinesła mathematical programming approach," *Or Spectrum*, vol. 29, no. 1, pp. 105–121, 2007.

[11] B. Zhang, L. Qi, and E. Yao, "A multi-commodity production and distribution model in supply chain," in *Supply Chain Management and Information Systems (SCMIS), 2010 8th International Conference on*. IEEE, 2010, pp. 1–5.

[12] C. Archetti, A. M. Campbell, and M. G. Speranza, "Multicommodity vs. single-commodity routing," *Transportation Science*, 2014.

[13] X. Wang, S. Niu, J. Gao, and J. Zhang, "A study on the highway network key segments identifying method based on the structural characteristics," in *CICTP 2014@ sSafe, Smart, and Sustainable Multimodal Transportation Systems*. ASCE, 2014, pp. 1819–1829.

[14] Y. Liu and N. Zhu, "A multi-objective detector location optimization approach," in *CICTP 2014@ sSafe, Smart, and Sustainable Multimodal Transportation Systems*. ASCE, 2014, pp. 1788–1800.

[15] S.-y. Li, D.-z. Zhang, X.-h. Liu, and X. Lu, "Research on multiple-mode freight transport flow assignment optimization with consideration of carbon emissions cost," in *CICTP 2014@ sSafe, Smart, and Sustainable Multimodal Transportation Systems*. ASCE, 2014, pp. 2883–2893.

[16] S. Zehnalova, Z. Horak, M. Kudelka, and V. Snael, "Local dependency in networks," in *Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on*. IEEE, 2013, pp. 250–254.

[17] D. Y. Kenett, T. Preis, G. Gur-Gershgoren, and E. Ben-Jacob, "Dependency network and node influence: application to the study of financial markets," *International Journal of Bifurcation and Chaos*, vol. 22, no. 07, p. 1250181, 2012.