

A hybrid algorithm for large-scale service network design considering heterogeneous fleets and service capacity decision

Zujian Wang

Department of Industrial Engineering
Tsinghua University, Beijing 100084, China
wang-zj14@mails.tsinghua.edu.cn

Abstract

Service network design addresses decisions related to selecting transportation services and distributing origin-to-destination commodity flow. In this paper, we consider the usage of heterogeneous fleets to provide transportation services. The amounts of fleets employed by each service decide its capacity. We propose both arc-based and cycle-path based models to formulate the problem. A hybrid algorithm is presented to solve the problem. The method includes pricing and cutting techniques, as well as a local search algorithm to obtain upper bound. The computation study indicates the efficiency of the proposed algorithm.

Keywords: heterogeneous fleet; column generation; cutting plane; local search

1 Introduction

Freight transportation plays an increasingly important role because of rapid development of e-commerce. Carriers are facing tremendous pressure from growing competition and start to pay more attention to optimize their transportation network. Service network design (SND) formulations address issues related to the tactical planning for consolidation-based freight transportation systems. Specifically, such formulations make decisions on selecting transportation services and distributing origin-destination (O-D) commodity flow. Network links represent services with operational assets (e.g. vehicles, drivers, power units, etc.) in SND formulations. Asset management concentrates on the operation and schedule of assets which The main goal of research on such issue is to acquire an efficient allocation and utilization of assets under the circumstance of satisfying the origin-destination demand.

The models with asset management introduced in the literature only take single type of vehicles into consideration.

It is more realistic to employ heterogeneous fleets in application to freight transportation system for carriers when providing transportation services. However, only a small amount of literature is devoted to considering heterogeneous fleets, e.g. [Kim et al. \(1999\)](#) and [Li et al. \(2016\)](#). Since service network

design problem is NP-hard, it is difficult to find a feasible solution of large-scale instance. However, in real-life application, carriers are confronted with huge transportation network and need efficient algorithms to reduce costs.

Our goal of this paper is to address this issue about heterogeneous fleets and propose two kinds of formulations. And a hybrid algorithm combining both exact and heuristics techniques is introduced to obtain high-quality feasible solutions in a short time. Meanwhile, the algorithm provide lower bounds on each instance to show how far the solution is away from the optimal value.

The outline of this paper is as follows. In Section 2, a brief literature review is elaborated. We describe the problem and propose both arc-based and cycle-path formulations in Section 3. Section 4 is devoted to the proposed hybrid algorithm, whereas experimental results is presented in Section 5. Concluding remarks is given in Section 6.

2 Literature review

Service network design(SND) contains the tactical planning decisions on selecting transportation services to satisfy O-D demands which is generally formulated as fixed-cost capacitated multicommodity network design problem (CMND), see [Magnanti and Wong \(1984\)](#), [Minoux \(1989\)](#). In addition to scheduling services and distributing flow, asset management is marginally considered in [Crainic \(2000\)](#), [Smilowitz et al. \(2002\)](#) and [Crainic \(2003\)](#). Since carriers attach increasing importance to utilize assets economically and efficiently, SND with asset management attracts increasing attention in the literature ([Andersen et al. 2009b](#), [Teyfaz et al. 2010](#)). Afterwards design-balanced constraints are proposed by [Pedersen et al. \(2009\)](#), which state that the number of assets entering each terminal must equal the number of leaving. The design-balanced constraints induce a cyclic structure for the assets movements to complete service operations and flow distribution.

With respect to the way to formulate SND, it is general to introduce arc-based formulation in the literature. While [Andersen et al. \(2009b\)](#) present four alternative formulations and analyze strengths and weaknesses of each formulation. The experimental results show that formulations based on cycle variables outperform other formulations in both solution quality and time. However, the amount of generated cycles grows exponentially and requires more efficient enumeration algorithm. Consequently, [Andersen et al. \(2011\)](#) propose an efficient branch-and-price scheme for the cycle-based formulation to avoid the enumeration of cycles. Their algorithm contains two subproblems for the dynamical construction of paths for commodities and cycles for vehicles.

Since the SND problem is NP-hard, heuristic methods are preferred choices with respect to the large scale of instances. [Pedersen et al. \(2009\)](#) present a two-phase tabu search meta-heuristic framework for the arc-based formulation. The proposed algorithm includes exploration phase to build neighborhoods and feasibility phase to implement an infeasibility-monitoring procedure for reaching good feasible solutions. [Teypez et al. \(2010\)](#) present a three-step decomposition heuristic to solve large-scale freight transportation formulation and build comprehensive solutions for real-life size instances. [Minh et al. \(2013\)](#) propose a three-phase meta-heuristic method combining neighbourhood-based and exact methods. Additionally, computational results on large-scale benchmark instances indicate the efficiency of the algorithm. Another meta-heuristic combining a cutting-plane scheme to obtain lower bound and a variable xing procedure to cut down the dimension of the problem is introduced in [Chouman and Crainic \(2015\)](#). [Crainic et al. \(2016\)](#) propose a new service network design model with resource constraints and design a solution approach combining column generation and slope scaling methods to find high-quality solutions.

A single asset type is considered in the aforementioned literature. As for heterogeneous assets, especially multiple types of vehicles, [Kim et al. \(1999\)](#) introduce three different formulations and two sets of valid inequalities. However, no efficient algorithm is mentioned to solve those formulations in that paper. [Li et al. \(2016\)](#) propose a tabu search based meta-heuristic to solve arc-based formulation for design-balanced capacitated multicommodity network design with heterogeneous vehicles. The amounts of each type of vehicle used by each service are not decided explicitly in their paper. And the experimental instances solved by the proposed algorithm cannot satisfy the real-life demand.

Our research makes the following contributions to the literature: First, we present both arc-based and cycle-path formulations for service network design considering heterogeneous fleets. Second, a hybrid algorithm including column generation, cutting plane and local search methods is introduced. Moreover, the proposed algorithm provides high-quality both lower bounds and feasible solutions for large-scale instances efficiently.

3 Problem statement and model formulation

In this section, we describe the problem mathematically and propose two equivalent formulations based on arc and cycle-path respectively.

3.1 Problem description

Since most literature assume homogeneous fleet is employed which may not accord with the common practices, we take heterogeneous fleet into consideration in this paper. The example network with heterogeneous fleet operating is shown in Fig.1. Under this assumption, the amount of different type of vehicles should be decided on each arc. In Fig.1, there two types of fleet providing services and arcs in solid line are labeled by the amount of specific type of vehicles. At each node the numbers of same type of vehicle entering and leaving equals. A cycle consists of a sequence of arcs satisfying design-balance requirements of each type of vehicle, i.e., $\{(A, B), (B, C), (C, A)\}$. A path starts at the origin of a commodity and end at its destination to satisfy the demand. Suppose there is a commodity with origin A and destination G , thus $\{(A, D), (D, G)\}$ and $\{(A, D), (D, F), (F, G)\}$ are two paths.

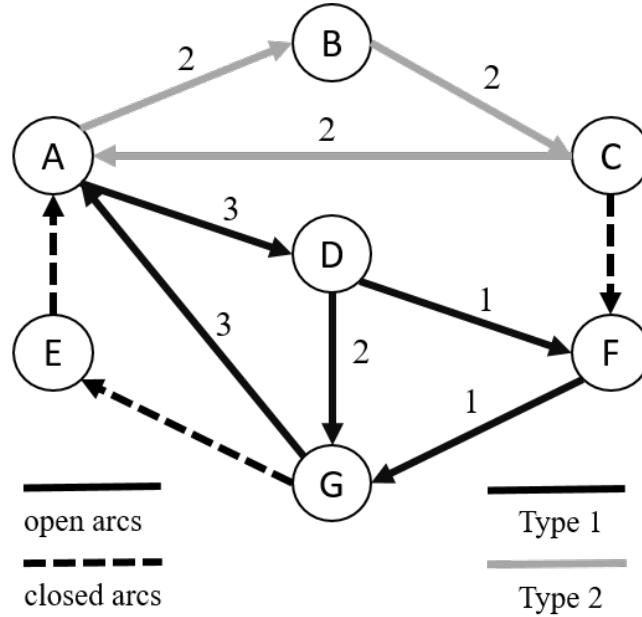


Figure 1: Example of heterogeneous fleet

Given a directed graph $G = (N, A)$ with node set N and arc set A . Each commodity k in commodity set K has a known demand d^k to be transited from its origin node $o(k)$ to destination node $d(k)$. Let F denote the set of fleet types and each type $f \in F$ has a certain capacity u^f .

3.2 Arc-based formulation

Let h_{ij}^f denote the fixed cost of utilizing fleet type $f \in F$ with respect to arc (i, j) . The unit flow cost for transporting commodity $k \in K$ on arc (i, j) is denoted c_{ij}^k . The capacity of fleet type f is denoted by u^f . We define $N_i^+ : \{j \in N : (i, j) \in A\}$, $N_i^- : \{j \in N : (j, i) \in A\}$. For each commodity k and node

i , define

$$\omega_i^k = \begin{cases} d^k & \text{if } i = o(k) \\ -d^k & \text{if } i = d(k) \\ 0 & \text{otherwise} \end{cases}$$

Two sets of decision variables are defined as follows.

- Continuous variable: x_{ij}^k represents the amount of flow with respect to commodity $k \in K$ on arc $(i, j) \in A$;
- Integer variable: y_{ij}^f denotes the amount of fleet type $f \in F$ that provide transportation services on arc $(i, j) \in A$.

The model is formulated as follows:

$$\min \sum_{f \in F} \sum_{(i,j) \in A} h_{ij}^f y_{ij}^f + \sum_{k \in K} \sum_{(i,j) \in A} c_{ij}^k x_{ij}^k \quad (1)$$

s.t.

$$\sum_{j \in N_i^+} x_{ij}^k - \sum_{j \in N_i^-} x_{ji}^k = w_i^k \quad \forall i \in N, \forall k \in K, \quad (2)$$

$$\sum_{j \in N_i^+} y_{ij}^f - \sum_{j \in N_i^-} y_{ji}^f = 0 \quad \forall i \in N, \forall f \in F, \quad (3)$$

$$\sum_{k \in K} x_{ij}^k \leq \sum_{f \in F} u^f y_{ij}^f \quad \forall (i, j) \in A, \quad (4)$$

$$x_{ij}^k \geq 0 \quad \forall (i, j) \in A, \forall k \in K, \quad (5)$$

$$y_{ij}^f \in Z_0^+ \quad \forall (i, j) \in A, \forall f \in F, \quad (6)$$

The objective function (1) minimizes the sum of the fixed costs for selecting and operating transportation service with respect to heterogeneous fleet and the total commodity flow costs. Constrains (2) represent the flow conversation relations for each node and each commodity. Equations (3) are the design-balance constraints, which ensure the same number of each type of vehicle entering and leaving a node. Through constrains (4), we enforce the total flow cannot exceed the total capacity provided by all vehicles on each arc. Finally, variable-type restrictions appear in constrains (5) and (6).

3.3 Cycle-path formulation

Let Q^f represent the set of all design cycle with respect to fleet type $f \in F$. And we define the set of all paths with respect to commodity k as P^k . Two sets of decision variables are given as follows.

- Cycle integer variables: y_q^f denotes the amount of vehicles used by each cycle;
- Path continuous variables: x_p^k denotes the volume of commodity k on path p .

We define two set of indicator variables:

- α_{ij}^q : equals 1 if arc (i, j) is included in cycle q , and 0 otherwise;
- δ_{ij}^p : equals 1 if arc (i, j) is included in path p , and 0 otherwise.

The fixed cost of cycle q with respect to fleet type $f \in F$ is denoted h_q^f , which equals $\sum_{(i,j) \in A} h_{ij}^f \alpha_{ij}^q$. Similarly, the flow cost of path p with respect to commodity $k \in K$ is denoted c_p^k , which equals $\sum_{(i,j) \in A} c_{ij}^k \delta_{ij}^p$.

$$\min \sum_{f \in F} \sum_{q \in Q^f} h_q^f y_q^f + \sum_{k \in K} \sum_{p \in P^k} c_p^k x_p^k \quad (7)$$

s.t.

$$\sum_{k \in K} \sum_{p \in P^k} \delta_{ij}^p x_p^k \leq \sum_{f \in F} \sum_{q \in Q^f} u^f \alpha_{ij}^q y_q^f \quad \forall (i, j) \in A \quad (8)$$

$$\sum_{p \in P^k} x_p^k = d^k \quad \forall k \in K \quad (9)$$

$$x_p^k \geq 0 \quad \forall p \in P^k, \forall k \in K \quad (10)$$

$$y_q^f \in Z_0^+ \quad \forall q \in Q^f, \forall f \in F \quad (11)$$

The objective function (7) minimizes the total cost calculated as the sum of the flow costs on paths plus the fixed costs for selected cycles with respect to specific type of vehicle. Through constrains (8), we enforce service capacity restrictions on each arc. Constrains (9) ensure demand satisfaction corresponding to the flow conservation equations for each commodity. Finally, Constrains (10) and (11) gives variable-type restrictions.

4 Solution method

Since the service network design problem is NP-hard, the exact algorithm is difficult to solve the large-scale instances optimally. Heuristics and metaheuristics are the common choices in real-life application. But the drawback of heuristics algorithm is that the results cannot be evaluated by lower bound. Therefore we propose a hybrid method combining the exact and the heuristics algorithm to solve the problem. The exact part mainly contains column generation and cutting plane method to obtain lower

bound. We consider the linear solution provided by column-and-cut generation approach as a good neighborhood and use local search techniques to find better feasible solutions.

Algorithm 1: Solution method

```

1  while find columns or cuts do
2      |   column generation;
3      |   cutting plane;
4  end
5  return lower bound;
6  use the lower bound as initial neighborhood;
7  local search procedure;
8  return upper bound;
```

4.1 Column-and-cut generation

The column generation approach has been proved to be efficient to solve cycle-path formulation by [Andersen et al. \(2011\)](#). With respect to cutting-plane procedure, it is also proved to be effective in calculating tight lower bounds for CMND-related problems, see [Chouman et al. \(2009, 2011\)](#), [Chouman and Crainic \(2015\)](#). We combine the methods to strength the lower bound for the proposed formulations.

4.1.1 Column generation (CG)

The linear relaxation of the cycle-path formulation constitutes the master problem(MP) which is solved by the CG approach. Since the number of variables in the MP is exponentially increasing, it is necessary to concentrate on a restricted master problem(RMP) which contains only a subset of variables(columns) in the MP. The CG approach starts with some necessary cycles and paths to prevent the constraints violation. We generate one path for each commodity to ensure its O-D demand. A reversed path from the destination of each commodity to its origin is combined with the former path to constitute a cycle. Therefore, we add a path and a cycle for each commodity as the initial columns to ensure the RMP feasible. The method adds new paths and cycles dynamically to the RMP until no additional column with negative reduced cost can be found.

The dual variable values in the solution of the RMP are past on to the subproblem for generating new columns. Dual variables η_{ij} and σ_k are associated with constrains (8) and (9) respectively. The

reduced cost associated with cycle-related variable y_q^f and path-related variable x_p^k is given by:

$$RC_{fq} = h_q^f + \sum_{(i,j) \in A} u^f \alpha_{ij}^q \eta_{ij} = \sum_{(i,j) \in q} (h_{ij}^f + u^f \eta_{ij}) \quad (12)$$

$$RP_{kp} = c_p^k - \sum_{(i,j) \in A} \delta_{ij}^p \eta_{ij} - \sigma_k = \sum_{(i,j) \in p} (c_{ij}^k - \eta_{ij}) - \sigma_k \quad (13)$$

We search for cycles and paths with negative reduced cost and add them to the RMP. The column generation procedure is presented in 2. When searching for the shortest cycle, we add a virtual node which is as same as the starting node as the destination node. Such that we can use the shortest path algorithm to find the shortest cycle. Note that dual variable $\eta_{ij} \leq 0$, then the arc costs may be negative when searching for the shortest cycle. Therefore, we employ the label-correcting algorithm implemented by [Ahuja et al. \(1993\)](#) to identify the shortest path.

Algorithm 2: Column generation procedure

```

1  add initial sets of cycles and paths to the RMP;
2  while find columns with negative reduced cost do
3      solve the RMP;
4      determine dual variables  $\eta_{ij}$  and  $\sigma_k$ ;
5      foreach node do
6          find the shortest cycle with respect to arc costs  $h_{ij}^f + u^f \eta_{ij}$ ;
7          let  $\theta_q$  denote the cost of the shortest cycle;
8          if  $\theta_q < 0$  then
9              add the corresponding cycle to the RMP;
10         end
11     end
12     foreach commodity do
13         find the shortest path with respect to arc costs  $c_{ij}^k - \eta_{ij}$ ;
14         let  $\theta_p$  denote the cost of the shortest path;
15         if  $\theta_p - \sigma_k < 0$  then
16             add the corresponding path to the RMP;
17         end
18     end
19 end

```

4.1.2 Cutting plane

The cutting plane procedure employs the generation of valid inequalities to strengthen the lower bound in relatively short time comparing with state-of-the-art software. We introduce two sets of valid inequalities including strong inequalities and cutset inequalities. Strong inequalities(SI) have been widely used to improve the quality of the LP lower bound, see [Gendron and Crainic \(1994\)](#), [Gendron et al. \(1999\)](#) and [Chouman and Crainic \(2015\)](#). We extend SI to be suitable for the proposed formulations in our paper which are defined in inequalities (14). The main effects of SI is to restrict the flow of every commodity on an arc with no fleet to be 0.

$$x_{ij}^k \leq d^k \sum_{f \in F} y_{ij}^f \quad \forall (i, j) \in A, \forall k \in K. \quad (14)$$

Cutset inequalities(CI) have been used by [Chouman and Crainic \(2015\)](#) to strengthen the lower bound for the DBCMND. As for the circumstance of considering heterogeneous fleet, CI have been presented by [Kim et al. \(1999\)](#) which are defined as

$$\sum_{f \in F} u^f Y_{S, \bar{S}}^f \geq D_{S, \bar{S}} \quad (15)$$

where we define cutset (S, \bar{S}) by partitioning the node set N into any nonempty subset S and its complement $\bar{S} = N \setminus S$. An arc (i, j) that connects node i in S to node j in \bar{S} belongs to the cutset (S, \bar{S}) . Let $Y_{S, \bar{S}}^f$ denote the total amount of type f vehicles used on the cutset (S, \bar{S}) arcs, i.e., $\sum_{(i, j) \in (S, \bar{S})} y_{ij}^f$ for the arc-based formulation and $\sum_{q \in Q^f} \sum_{(i, j) \in (S, \bar{S})} \alpha_{ij}^q y_q^f$ for the cycle-path formulation. Let $D_{S, \bar{S}}$ denote the aggregate demand of all commodities with their origin in S and destination in \bar{S} . CI ensure the total demand that must flow from S to \bar{S} will be satisfied by enough capacity on the arcs of (S, \bar{S}) . In general, the LP solution will not violate CI. Therefore, we lift CI by using a integer rounding procedure to produce *Chvátal-Gomory* (C-G) cuts which are proved to be valid by [Kim et al. \(1999\)](#). The impact of adding cuts on the pricing problem is negligible, thus there is no need to modify arc costs.

$$\sum_{f \in F} \left(\left\lceil \frac{u_f}{u_l} \right\rceil Y_{S, \bar{S}}^f \right) \geq \left\lceil \frac{D_{S, \bar{S}}}{u_l} \right\rceil \quad \forall l \in F \quad (16)$$

The challenge in using cutset inequalities is the large number of potential cutsets, it is impractical to enumerate all the associated inequalities. Moreover, it becomes more computationally expensive to lift the inequalities as the scale of the problem increases. Therefore, we only generate single-node cutset inequalities, see [Chouman et al. \(2016\)](#) for the details.

The dynamic generation of valid inequalities is embedded in the column generation approach. Once there exist violated cuts, column generation procedure is executed again. The price-and-cut loop stops only if neither negative reduced cost columns nor violated cuts are found.

4.2 Local search

Based on the linear solution obtained from lower bound, we consider it as a good neighborhood to search for feasible solutions. Furthermore, since the application of SND employing heterogeneous fleets by logistics enterprises will deal with large-scale network, it is necessary to present efficient algorithms to find high-quality feasible solutions efficiently. Therefore, we propose a two-stage local search algorithm to obtain feasible solutions.

After we obtain the lower bound, the CG and cutting plane method will provide the value of \bar{x}_{ij}^k and \bar{y}_{ij}^f . Then define the arc subset including arcs with nonzero flow as A_1 , i.e., $A_1 = \{(i, j) \in A : \sum_{k \in K} \bar{x}_{ij}^k > 0\}$, and another arc subset including arcs with nonzero fleets as A_2 , i.e., $A_2 = \{(i, j) \in A : \sum_{f \in F} \bar{y}_{ij}^f > 0\}$. It is easy to find that $A_1 \subseteq A_2 \subseteq A$ because of the capacity constraints. The first stage satisfy O-D demands and determine flows on each arc $(i, j) \in A_1$. The second stage ensures the balance of fleets and provides enough capacity for flows on each arc $(i, j) \in A_2$. The corresponding formulations in two stages are defined as follows.

4.2.1 Stage 1: flow distribution problem

Without taking service selection and fleet assignment into account, the formulation of flow distribution problem is actually a capacitated multi-commodity minimum cost flow problem (CMCF). The values of $\sum_{(i,j) \in A_2} u^f y_{ij}^f$ decide the arc capacity u_{ij} in A_1 .

$$\min \sum_{(i,j) \in A_1} \sum_{k \in K} c_{ij}^k x_{ij}^k \quad (17)$$

s.t.

$$\sum_{j \in N_i^+} x_{ij}^k - \sum_{j \in N_i^-} x_{ji}^k = w_i^k \quad \forall i \in N, \forall k \in K, \quad (18)$$

$$\sum_{k \in K} x_{ij}^k \leq u_{ij} \quad \forall (i, j) \in A_1, \quad (19)$$

$$x_{ij}^k \geq 0 \quad \forall (i, j) \in A_1, \forall k \in K. \quad (20)$$

After solving the CMCF and redistributing the flow on each arc, the arcs without flow are removed from A_1 , i.e., $A_1 = A_1 \setminus \{(i, j) \in A_1 : \sum_{k \in K} x_{ij}^k = 0\}$.

4.2.2 Stage 2: fleet assignment problem

After the flow distribution to satisfy demands of all O-D commodities has been completed, we need to assign heterogeneous fleets to provide transportation services. Let Ω_{ij} be the total flow of all commodities distributed on arc $(i, j) \in A_2$, which equals $\sum_{k \in K} x_{ij}^k$ provided by solving CMCF in the first stage. The mathematical formulation of Stage 2 defined on arc set A_2 is as follow:

$$\min \sum_{f \in F} \sum_{(i,j) \in A_2} h_{ij}^f y_{ij}^f \quad (21)$$

s.t.

$$\sum_{j \in N_i^+} y_{ij}^f - \sum_{j \in N_i^-} y_{ji}^f = 0 \quad \forall i \in N, \forall f \in F \quad (22)$$

$$\sum_{f \in F} u^f y_{ij}^f \geq \Omega_{ij} \quad \forall (i, j) \in A_2 \quad (23)$$

$$y_{ij}^f \in \mathbb{Z}_0^+ \quad \forall (i, j) \in A_2, \forall f \in F \quad (24)$$

Constraints (23) ensure each service arc will provide sufficient vehicle capacity for the total flow.

4.2.3 Cycle-based neighborhoods

After the two formulations are solved, we obtain a feasible solution of the original model. The quality of the feasible solution depends on the two arc subsets A_1 and A_2 . To find promising A_1 and A_2 , we introduce cycle-based neighborhoods to search for better feasible solutions. The cycle-based neighborhoods have been presented and proved to be efficient when solving CMND-related problems, see [Ghamlouche et al. \(2003, 2004\)](#), [Li et al. \(2016\)](#).

The cycle-based neighborhoods aim at redirecting flow on one path to another, and exploring the space of arc subsets A_1 and A_2 through opening or closing some arcs. To illustrate, consider the partial network in [2](#), there is a cycle formed by paths $\{(A, B), (B, C), (C, F)\}$ and $\{(A, E), (E, F)\}$. If we redirect the flow from the former to the latter, a new solution will be generated by opening arcs $(A, E), (E, F)$ and closing empty arcs $(A, B), (B, C), (C, F)$. The procedure of finding a cycle-based neighborhood can be summarized as follows.

- Identify a cycle containing two paths connecting two points.
- Redirecting the flow from one path to another and ensure that at least one open arc becomes empty.

- Open all formerly closed arcs in the cycle and close all formerly open arcs if they are empty after the flow redirection.

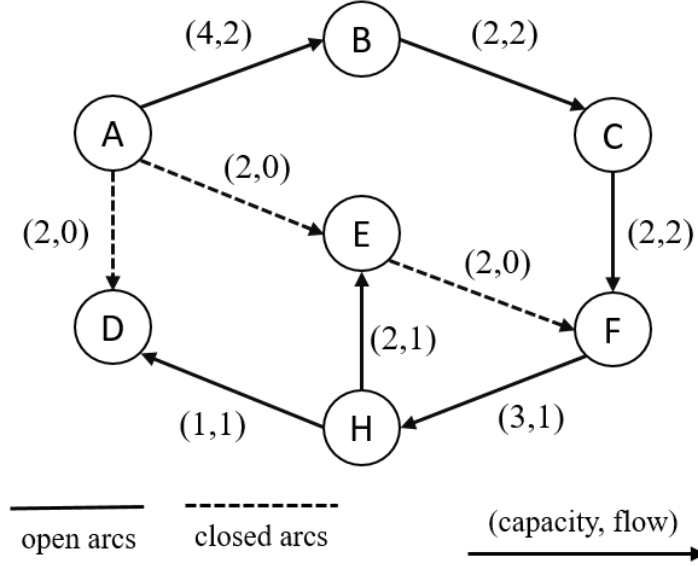


Figure 2: Example of cycle in partial network

It is very important to implement an efficient procedure to find a promising neighborhood since complete evaluation of all moves is not practical, especially facing huge networks. Note that since we have to close at least one previously open arc, it seems suitable to start from a nonempty open arc (i^*, j^*) with $\gamma = \sum_{k \in K} x_{i^* j^*}^k$ units of flow. Since the arc (i^*, j^*) will be closed after the flow redirection, the residual capacity of any cycle associated with (i^*, j^*) must be at least γ . A residual network must be built to efficiently search for low-cost cycles corresponding to promising moves without exhaustive exploration and exact evaluation.

To build such a residual network associated to (i^*, j^*) and γ units of flow redirection, we include two types of arc sets in the new network, one with positive arc cost and the other with negative arc cost. Each arc in the original network will be replaced by at most two arcs $(i, j)^+$ and $(j, i)^-$ from the two arc sets respectively. Define \bar{c}_{ij} as the average flow cost on arc (i, j) , i.e. $\bar{c}_{ij} = \sum_{k \in K} c_{ij}^k / |K|$. Let \bar{F} denote the average fixed cost to evaluate the impact of fleet assignment because of opening or closing an arc, which equals $\sum_{f \in F} h_{ij}^f / |F|$.

Arc $(i, j)^+$ will be included in the residual network if the residual capacity of arc (i, j) is no less than γ , i.e. $u_{ij} - \sum_{k \in K} x_{ij}^k \geq \gamma$. The cost c_{ij}^+ corresponding to $(i, j)^+$ approximates the additional cost of distribute γ units of flow on arc (i, j) . It is calculated as the additional flow cost, plus the average

fixed cost if arc (i, j) is closed at present, i.e. $(i, j) \notin A_2$.

$$c_{ij}^+ = \begin{cases} \bar{c}_{ij}\gamma + \bar{F} & \text{if } (i, j) \notin A_2, \\ \bar{c}_{ij}\gamma & \text{otherwise.} \end{cases}$$

Since arc $(j, i)^-$ is used to represent the flow reduction on arc (i, j) , it will be included in the residual network only if the current flow on (i, j) is no less than γ . Symmetrically, the cost c_{ji}^- associated to $(j, i)^-$ approximates the reduced cost of decreasing γ units of flow on arc (i, j) . It is calculated as the reduced flow cost, minus the average fixed cost if arc (i, j) will be empty after reduction of γ units of flow.

$$c_{ji}^- = \begin{cases} -\bar{c}_{ij}\gamma - \bar{F} & \text{if } \sum_{k \in K} x_{ij}^k = \gamma, \\ -\bar{c}_{ij}\gamma & \text{if } \sum_{k \in K} x_{ij}^k > \gamma. \end{cases}$$

Such a residual network aims at identifying a lowest-cost cycle containing arc (i^*, j^*) to redirect γ units of flow. Moreover, it takes into account the influence of opening or closure of arcs to search for better feasible solutions. To illustrate how to find the lowest-cost cycle in the residual network, consider the network example of 3. Suppose arc (A, B) is the candidate arc to be closed, the residual network associated with (A, B) is shown in the 3. The average flow costs of all arcs in the example are assumed to be 2, and the average fixed cost \bar{F} is 3. Arcs in the residual network are labeled by their costs. Note that there are two cycles associated with arc (A, B) : $(A, E), (E, F), (F, C), (C, B), (B, A)$ and $(A, E), (E, B), (B, A)$. The two cycles have costs of -7, 4, respectively. Since the former has the lowest cost, 2 units of flow are moved from path $(A, B), (B, C), (C, F)$ to path $(A, E), (E, F)$. Therefore, a new neighborhood is obtained by opening arcs $(A, E), (E, F)$ and closing empty arcs $(A, B), (B, C), (C, F)$.

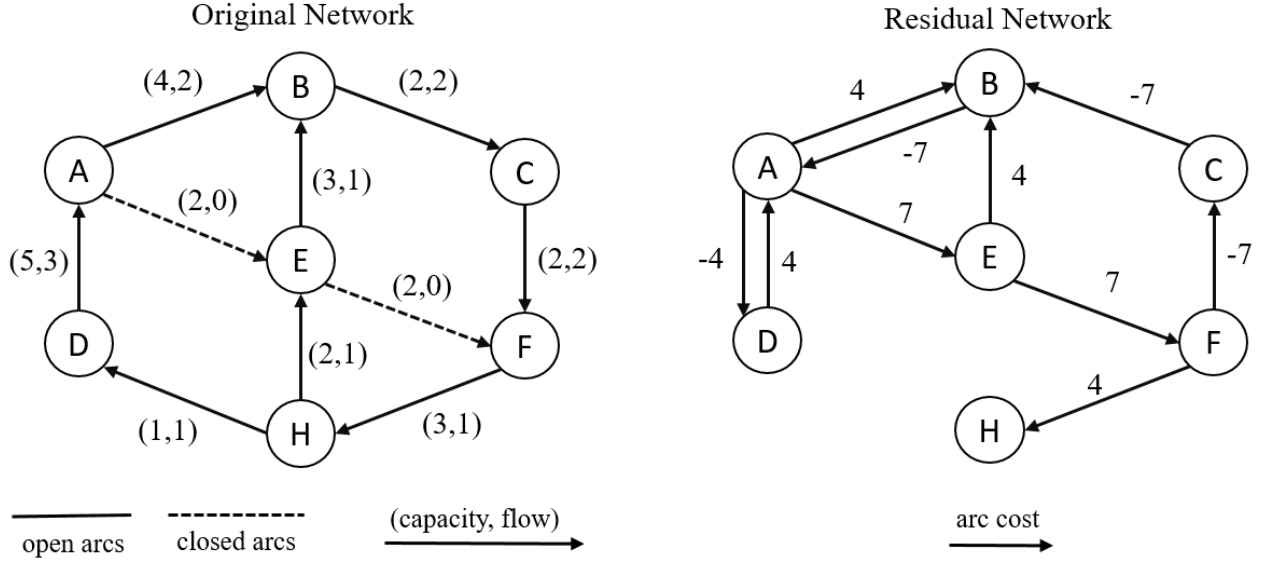


Figure 3: Network example associated with (A, B)

4.2.4 Main local search loop

Since each new neighborhood is associated with a candidate arc, let A^+ denote the candidate arcs, which equals $\{(i, j) \in A : \sum_{k \in K} x_{ij}^k > 0\}$. With respect to each arc (i^*, j^*) in A^+ , we build a residual network and find the shortest path from i^* to j^* to complete the cycle. We still employ the label-correcting algorithm implemented by [Ahuja et al. \(1993\)](#) to identify the shortest path.

Note that the closure of some arcs may lead to infeasible neighboring solutions. Therefore it is necessary to check whether at least one path links the origin and destination of each commodity before comparing the cycle cost. Although it can not ensure the feasibility of the first stage when solving the flow distribution problem, it still decreases the appearance of infeasible local search moves.

Once the lowest cost cycle is found, A_1 is updated by the move including opening and closure of some arcs. Then we solve the flow distribution problem and update A_1 again by closing empty arcs. Before solving the fleet assignment problem, A_2 is updated by $A_1 \cup A_2$. If the two stages are both feasible, we obtain a feasible solution of the original formulation. Furthermore, A_1 and A_2 in the next iteration are determined by this feasible solution. To explore more extensive search space, we assign a tabu status to the arc associated with the feasible move. The local search approach terminates if either of the two stages is infeasible or the maximum iteration is reached. The sketch of the local search

algorithm is as follows.

Algorithm 3: Local search algorithm

Input: Linear solution obtained by CG

```
1  while termination criterion not met do
2      determine  $A_1, A_2$  and  $A^+$ ;
3      foreach  $(i, j) \in A^+$  do
4          | create residual network, and determine the best cycle;
5      end
6      find the cycle with minimum cost;
7      update  $A_1$  then solve flow distribution problem, stop if infeasible;
8      update  $A_1$  and  $A_2$ ;
9      solve vehicle assignment problem and stop if infeasible;
10     update the current solution;
11     assign a tabu status to  $(i, j)$ ;
12     if find better solution then
13         | update best feasible solution;
14     end
15 end
```

Output: best feasible solution

5 Numerical experiments and analyses

The computation study is implemented in C++, using CPLEX 12.63 as the linear programming solver. All experiments are performed using a computer with four 64-bit 2.4 GHz Intel Core processors and 4 GB of RAM, operating under Windows 10. Computing time are reported in seconds.

6 Conclusion

Acknowledgments

This work was supported by the National Natural Science Foundation of China

References

- Ahuja, RK, TL Magnanti, P Mirchandani. 1993. Network flow-theory, algorithm, and applications. *Prentice-Hall, Englewood Cliffs, NJ* .
- Andersen, J., Christiansen M., Crainic T. G., Nhaug R. 2011. Branch and price for service network design with asset management constraints. *Transportation Science* **45**(1) 33–49.
- Andersen, J., Crainic T.G., Christiansen M. 2009b. Service network design with asset management: Formulations and comparative analyses. *Transportation Research Part C Emerging Technologies* **17**(2) 197–207.
- Chouman, M., T.G. Crainic. 2015. Cutting-plane matheuristic for service network design with design-balanced requirements. *Transportation Science* **49**(1) 99–113.
- Chouman, Mervat, Teodor Gabriel Crainic, Bernard Gendron. 2009. A cutting-plane algorithm for multicommodity capacitated fixed charge network design. *Technical report, Center for Research on Transportation, Universit de Montral, Montral* .
- Chouman, Mervat, Teodor Gabriel Crainic, Bernard Gendron. 2011. Commodity representations and cut-set-based inequalities for multicommodity capacitated fixed-charge network design. *Technical report CIRRELT-2011-56, Centre Interuniversitaire de Recherche sur Les Rseaux dEntreprise, La Logistique et Les Transports, Universit de Montral, Montral* .
- Chouman, Mervat, Teodor Gabriel Crainic, Bernard Gendron. 2016. Commodity representations and cut-set-based inequalities for multicommodity capacitated fixed-charge network design. *Transportation Science* <https://doi.org/10.1287/trsc.2015.0665>.
- Crainic, Teodor Gabriel, Mike Hewitt, Michel Toulouse, Duc Minh Vu. 2016. Service network design with resource constraints. *Transportation Science* **50**(4).
- Crainic, T.G. 2000. Service network design in freight transportation. *European Journal of Operational Research* **122**(2) 272–288.
- Crainic, T.G. 2003. *Long-haul freight transportation*. Springer.
- Gendron, Bernard, Teodor Gabriel Crainic. 1994. Relaxations for multicommodity capacitated network design problems. *Technical report, Publication CRT-945, Centre de Recherche sur Les Transports, Universit de Montral, Montral* .
- Gendron, Bernard, Teodor Gabriel Crainic, Antonio Frangioni. 1999. *Multicommodity capacitated network design*. Springer US.
- Ghamlouche, I., Crainic T.G., Gendreau M. 2003. Cycle-based neighbourhoods for fixed-charge capacitated multicommodity network design. *Operations Research* **51**(4) 655–667.
- Ghamlouche, Ilfat, Teodor Gabriel Crainic, Michel Gendreau. 2004. Path relinking, cycle-based neighbourhoods and capacitated multicommodity network design. *Annals of Operations Research* **131**(1) 109–133.

- Kim, D., Barnhart C., Ware K., Reinhardt G. 1999. Multimodal express package delivery: a service network design application. *Transportation Science* **33**(4) 391–407.
- Li, Xiangyong, Kai Wei, Y. P. Aneja, Peng Tian. 2016. Design-balanced capacitated multicommodity network design with heterogeneous assets. *Omega* .
- Magnanti, T. L., R.T. Wong. 1984. Network design and transportation planning: models and algorithms. *Transportation Science* **18**(1) 1–55.
- Minh, V., Crainic T., Toulouse M. 2013. A three-stage matheuristic for the capacitated multi-commodity fixed-cost network design with design-balance constraints. *Journal of heuristics* **19** 757–795.
- Minoux, M. 1989. Networks synthesis and optimum network design problems: Models, solution methods and applications. *Networks* **19**(3) 313360.
- Pedersen, M.B., Crainic T.G., Madsen O. B. G. 2009. Models and tabu search metaheuristics for service network design with asset-balance requirements. *Transportation Science* **43**(2) 158–177.
- Smilowitz, K. R., Atamtrk A., Daganzo C.F. 2002. Deferred item and vehicle routing within integrated networks. *Transportation Research Part E: Logistics and Transportation Review* **39**(4) 305–323.
- Teypez, N., Schrenk S., Cung V.D. 2010. A decomposition scheme for large-scale service network design with asset management. *Transportation Research Part E: Logistics and Transportation Review* **46**(1) 156–170.