

Avoiding unnecessary demerging and remerging of multi-commodity integer flows

Zhiyuan Lin¹  | Raymond S. K. Kwan²

¹Institute for Transport Studies, University of Leeds, Leeds, UK

²School of Computing, University of Leeds, Leeds, UK

Correspondence

Zhiyuan Lin, Institute for Transport Studies, University of Leeds, Leeds, LS2 9JT, UK.
Email: z.lin@leeds.ac.uk

Funding information

This research was supported by the Engineering and Physical Sciences Research Council, Grant/Award Number: EP/M007243/1

Abstract

Resource flows may merge and demerge at a network node. Sometimes several demerged flows may be immediately merged again, but in different combinations compared to before they were demerged. However, the demerging is unnecessary in the first place if the total resources at each of the network nodes involved remains unchanged. We describe this situation as “unnecessary demerging and remerging (UDR)” of flows, which would incur unnecessary operations and costs in practice. Multi-commodity integer flows in particular will be considered in this paper. This deficiency could be theoretically overcome by means of fixed-charge variables, but the practicality of this approach is restricted by the difficulty in solving the corresponding integer linear program (ILP). Moreover, in a problem where the objective function has many cost elements, it would be helpful if such operational costs are optimized implicitly. This paper presents a heuristic branching method within an ILP solver for removing UDR without the use of fixed-charge variables. We use the concept of “flow potentials” (different from “flow residuals” for max-flows) guided by which underutilized arcs are heuristically banned, thus reducing occurrences of UDR. Flow connection bigraphs and flow connection groups (FCGs) are introduced. We prove that if certain conditions are met, fully utilizing an arc will guarantee an improvement within an FCG. Moreover, a location sub-model is given when the former cannot guarantee an improvement. More importantly, the heuristic approach can significantly enhance the full fixed-charge model by warm-starting. Computational experiments based on real-world instances have shown the usefulness of the proposed methods.

KEYWORDS

fixed-charge multi-commodity flow, flows on bipartite graphs, heuristic branching, train unit scheduling, unnecessary flow demerging/remerging

1 | INTRODUCTION

Many routing and logistics optimization problems can be modeled by multi-commodity flows. When flows *merge*, peak demands could be satisfied; and *demerging* allows more efficient use of resources in periods of lower demands. Optimization models may need additional controls over how multi-commodity flows merge/demerge. Examples (A) and (B) of Figure 1 illustrate what flow merging and demerging are.

In particular, an unacceptable situation in practice is the *remerging* of some *demerged* flows over a group of nodes. The flows could be recombined into combinations that are equivalent to the case without any demerging/remerging. We call this

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. *Networks* published by Wiley Periodicals LLC.

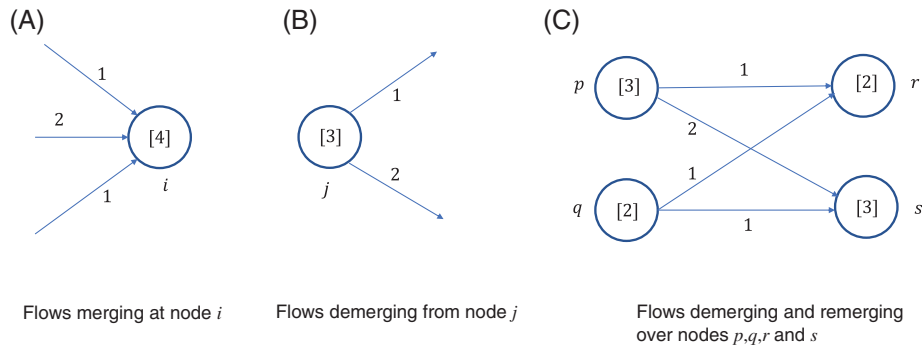


FIGURE 1 Examples of flow merging (A), demerging (B), and demerging/remerging (C). Arc flow amounts are marked along the arcs and node flow amounts are marked by [] inside the nodes [Color figure can be viewed at wileyonlinelibrary.com]

phenomenon unnecessary demerging and remerging (UDR). Figure 1C gives such an example of flow demerging/remerging over four nodes $p, q, r,$ and s . The additional controls may be modeled by means of binary fixed-charge variables, but they would increase the computational burden tremendously. Examples of such multi-commodity flow problems with potential UDR include (with different characteristics in how flows merge/demerge) Train Unit Assignment [5], Train Unit Shunting [11,17], Rolling Stock Rescheduling during Disruptions [24], and there are some closely related classic problems such as the Multivehicle Tanker Scheduling Problem [4] and the Fixed-charge Transportation Problem [25]. This paper is inspired from the Train Unit Scheduling Problem (TUSP) [21], where unit coupling/decoupling operations correspond to flow merging/demerging. Typically, we focus on a kind of integer multi-commodity flow problem based on a directed acyclic graph (DAG) $G = (N, A)$ where for some nodes $j \in N$, there is a demand r_j to be satisfied by the flows, and a capacity u_j . Moreover:

- i. Each arc $a \in A$ has an attribute $l_a \in L_G$ derived from G , which can be a location, a time slot, or a person in charge, and so on. We use *location* to refer to the attribute hereafter.
- ii. For all arcs associated with location $l \in L_G$, denoted by $a \in A_l$, we can collect the nodes of the arcs in set $N_l = \{j_1, j_2, \dots, j_n\}$, which is partitioned into two groups I_l and J_l where the former contains the tails of all arcs in A_l and the latter the heads, that is, $A_l \subseteq I_l \times J_l$.
- iii. I_l, J_l and A_l form a bipartite graph (“bigraph” hereafter), denoted by $B_l = (I_l, J_l, A_l)$. From the perspective of a bigraph, resources flow from I_l to J_l in a single direction.
- iv. Often $|L_G| > 1$. When there is only one location, the problem is reduced to a variant of the transportation problem (with possible forbidden arcs).

More than one flow can be merged together to satisfy the demand of a node, and the reverse process is the demerging of flows from a node. The term “merging/demerging” will sometimes be shortened as “M/D” in this paper. Flow M/D activities are often essential for satisfying high demands at nodes by providing more flows there. In addition, flow M/D can also be used as a way of redistributing resources across the overall network regardless of demands.

The control of M/D activities is not trivial in producing a realistic and operable flow solution from a mathematical programming based model. UDR in a flow solution occurs when an alternative solution exists that can achieve the same or similar cost and effect while having a fewer number of M/D activities. Theoretically by using “fixed-charge” variables, the costs due to M/D operations can be minimized explicitly in the objective, for example, in [22] (where M/D corresponds to train unit coupling/decoupling). However, in practice, the difficulty of solving the model with fixed-charge variables prevents its application to medium/large instances. Processing UDR at a later stage is an alternative approach that is shown to be workable and effective for real-world instances [20]. However, it cannot guarantee global optimality.

This paper proposes a heuristic branch-and-bound (BB) method for removing UDR without using fixed-charge variables. We also propose a warm-start framework that will significantly improve the performance of the exact fixed-charge flow model. This method partly relies on the fact that the number of M/D operations can be expressed in terms of flow potentials of a feasible solution, and by favoring full arc utilization heuristically on the BB tree, the occurrences of UDR would be reduced to near-optimal. It should be noted that here “flow potential” is a new concept and should not be confused with “flow residuals” used for finding maximum flows, and the calculation processes of the two are different. It will be shown that the exact method based on the fixed-charge flow model would significantly benefit from this heuristic by warm-starting strategies and it even enables the exact fixed-charge model to solve instances that were once intractable due to their sizes.

To the best knowledge of the authors, the problem investigated in this paper is almost an uncharted area and little existing literature can be found. A first attempt is tried in [21] to remove UDR (as train unit coupling/decoupling operations), where the concept of flow potential and flow connection groups (FCGs) are introduced and implemented into the BB process to

heuristically remove UDR. A sufficient condition for guaranteeing an improvement on the M/D count is also proposed, which is used for guiding heuristic branching. Nonetheless, we point out certain limitations of [21]:

- The sufficient condition is restricted to cases with interchangeable commodity types.
- The sufficient condition and branching scheme only consider fully utilizing balanced arcs, while unbalanced underutilized arcs are ignored.
- FCGs, due to their local nature, are unable to handle more complex UDR involving several connection groups at a location.

In this paper, to deal with the above three limitations, we give a sufficient condition guaranteeing an improvement for non-interchangeable commodity types, a generalized sufficient condition for unbalanced arcs, and a location sub-model as a pure fixed-charge integer multi-commodity transportation problem that will consider the entire location. We also provide theoretical background about the difficulty in determining the existence and possibility of improvement for UDR in general, and thus justify the rationale of the location sub-model. More importantly, we propose a warm-start framework using sub-optimal solutions given by the heuristic model that enables the fixed-charge model to solve medium instances exactly with high-quality solutions. Finally, the problem described in [21] is limited within train unit scheduling. We realize that it can be generalized into a broader scope of application areas where multi-commodity flows are used. Therefore, this paper presents the problem and its solution approaches in a generic multi-commodity flow perspective and will only come to real-world instances in the computational experiment section with results from the TUSP.

This paper is organized as follows. Section 2 surveys relevant literature. Section 3 describes the issue of UDR yielded from certain network flow formulations. Section 4 introduces an enhanced formulation with “fixed-charge” variables included, and discusses its pros and cons. Section 5 proposes a significantly upgraded heuristic branching method (compared with [21]) for removing UDR, including the concept of flow potentials and their relations to the number of M/D operations, the idea of FCGs, arc full utilization, and two alternative ways for validity checking of arc full utilization. The details of the heuristic branching scheme, Arc Full Utilization Branching (AFUB), will also be presented. Section 6 reports experiments conducted based on real-world instances of some UK rail networks. Finally, Section 7 concludes this paper and discusses future research.

2 | LITERATURE REVIEW

2.1 | Multi-commodity flow problems and their fixed-charge variants

The multi-commodity flow problem can be traced back to the pioneering work by Ford and Fulkerson [10]. In spite of many researches on the continuous and binary versions of the problem, general integer multi-commodity flow problems where variables can take general integral values have received relatively less attention, as reported in [2]. In Barnhart et al. [3], an origin-destination integer multi-commodity flow problem is solved by branch-and-price-and-cut, where the path variables are binary and each commodity can only use one path. In Alvelos [2], an approach based on branch-and-price for general integer multi-commodity flow problems is proposed, where invalid columns from subproblems are prevented explicitly by adding branching constraints, and customized branching techniques are used taking advantage of the problem structure.

The fixed-charge single-/multi-commodity flow problem is characterized by adding extra fixed-charge variables. For instance, a description of a single commodity fixed-charge flow problem can be found in [16]. Different terms referring to the same or similar problems with “fixed-charge” and “multi-commodity” features exist, for instance, the Network Design Problem [1,12,19]. Similar to multi-commodity flows, a mixed integer version and an all-integer version can be further categorized for the fixed-charge variants [15]. The fixed-charge multi-commodity flow (FCMF) problems are more challenging compared with a standard multi-commodity flow problem, even for the mixed integer version. Most of the current solution methods for FCMF problems are (meta)heuristic-based. Moreover, while there are plenty of researches on the mixed integer FCMF, the all-integer FCMF version has received less attention. In [15], Hewitt et al. propose a hybridization method combining exact algorithms with heuristic search techniques that produces provably high-quality solutions quickly for the all-integer FCMF problem with binary flow variables. The method searches over neighborhoods involving carefully chosen integer programming models derived from the arc formulation of the FCMF. To get lower bounds, the linear programming relaxation of the path formulation of FCMF is used and strengthened with cuts derived the neighborhood search. Later in Hewitt et al. [14], an algorithm for fixed-charge FCMF with binary flow variables uses restrictions of the problem to produce high-quality solutions quickly. Column generation is used both for generating problem restrictions and for producing bounds on the value of the optimal solution.

The focus of this paper is not on solving a generic all-integer FCMF problem, but only on integer multi-commodity flow problems with potential unnecessary flow demerging/remerging issues, which could be formulated as a special kind of the integer FCMF problem with the weights on the fixed-charge variables uniquely determined. (see Section 4).

2.2 | Unnecessary flow demerging/remerging problem

Previous literature solely dedicated to UDR is very scarce, and [21] is likely to be the only one. There are relevant researches where similar observations are reported, and all of them are from the field of rolling stock scheduling. Coupling/decoupling (C/D) operations among train multiple units precisely correspond to flow merging/demerging when train unit scheduling is formulated by multi-commodity flow models. As far as the authors are aware, the only closely relevant discussions are found in [11,17] in train unit shunting at the station level, and in [24] in rolling stock rescheduling.

Coupling/decoupling costs are considered in shunting of train units overnight or during long idling time in the day. Freling et al. [11] propose a two-stage integer linear programming approach for the Train Unit Shunting Problem at a station within 24 hours (mainly overnight). The main focus is on shunting train units that are temporarily unused between two trips to sidings (shunt yard). Instances for the station with up to 80 train units to be parked at 19 sidings have been tested where near-optimal solutions can be found. The definition of a unit block as “an entity of one or more train units that remain together for the entire planning period” is given in [11], and this idea will be also used in this paper. At the first stage of matching supply to demand, the issue of “keeping train units together as much as possible...the number of blocks has to be minimized” is discussed. This is essentially a special case in dealing with unnecessary coupling/decoupling for train units with long idling time and shunting movements. At the first stage, each shunting movement from an arrival trip to a departure trip is assigned with a weight to be minimized in the objective, thus realizing the goal of keeping units together as much as possible. Kroon et al. [17] later propose an integrated approach for the same Train Unit Shunting Problem as in [11] where unit matching and shunting assignment are processed together, hence global optimality is guaranteed. In both papers, since the problem studied are different from our work, it requires only binary variables to formulate, and thus unnecessary coupling/decoupling operations can be removed by directly adding relevant binary variables to the objective.

How to best reschedule a fleet of rolling stock units during a disruption is an optimization problem regularly faced by railway operators. Lusby et al. [24] propose a branch-and-price method based on a path formulation over a time-space network. Near-optimal solutions can be found within a few seconds on real-world instances. Coupling/decoupling is included as a kind of cost in the objective. Instead of the exact number of coupling/decoupling operations, an approximated number is used that the cost due to C/D for each unit's workload (represented by a path) is estimated in advance by the number of times a unit enters and leaves the (sub)trips it covers (or equivalently the times a unit returns to depots implied by its path).

The unnecessary C/D investigated in this paper's experiment section for real-world TUSP covers a broader range of trip connections compared with [11,17]. In their papers, only the C/D operations from long dwelling connections requiring shunting activities are considered, while in this research C/D operations from all kinds of connections in a train unit schedule are considered. Note that in the UK, C/D operations can be done at platforms straightforwardly without extra shunting activities in a relatively short time duration, for example, 5 minutes. In contrast to [24], the exact number of C/D operations is used in this paper, and we do not assume that depot-return activities are necessarily associated with C/D operations.

An earlier short conference paper [21] by the authors firstly discussed the unnecessary coupling/decoupling problem as a new kind of network flow problem. Nonetheless, as is already mentioned in Section 1, [21] has several limitations to be overcome by this paper. It also has not got the warm-start framework that is shown to be pivotal in getting high-quality solutions compared with the manual results in the TUSP.

3 | PROBLEM DESCRIPTION

Assume that an integer multi-commodity flow problem is based on a DAG $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. The commodities are denoted by $k \in K$. $\mathcal{N} = N \cup \{s, t\}$ represents the nodes where s, t are the source and sink with a flow of amount b^k to be routed from s to t for each commodity $k \in K$.¹ N is the set of intermediate nodes j each with a demand r_j and a capacity u_j . $\mathcal{A} = A \cup A_0$ is the arc set where $A = \{(i, j) \mid i, j \in N\}$ contains connection arcs and $A_0 = \{(s, j) \mid j \in N\} \cup \{(j, t) \mid j \in N\}$ contains source/sink arcs. A path is defined restrictively as an s - t path from the source to the sink. Let P^k be the set of paths associated with commodity k (i.e., the paths can be used to route commodity k) and x_p be the flow amount in $p \in P^k$. An integer multi-commodity flow model based on path variables can be formulated as:

$$(M_x) \quad \min \quad \sum_{k \in K} \sum_{p \in P^k} c_p x_p \quad (1)$$

$$\text{s.t.} \quad \sum_{p \in P^k} x_p \leq b_k, \quad \forall k \in K; \quad (2)$$

¹Our proposed approaches also work for integer multi-commodity flow problems with multiple sources and sinks. Here without the loss of generality, we assume that there is a single source s and a single sink t .

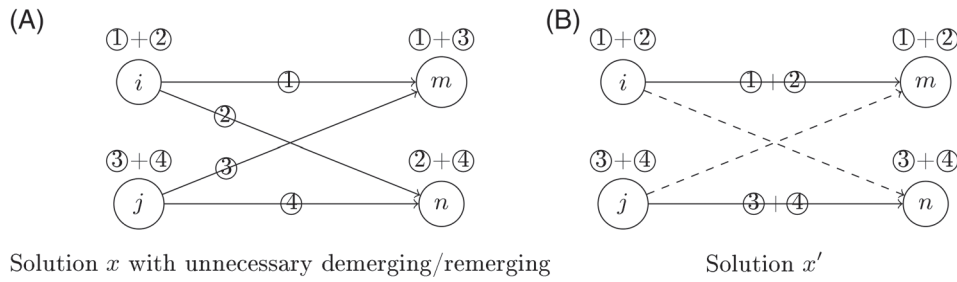


FIGURE 2 Solutions x and x' over the same nodes and arcs

$$\sum_{k \in K_j} \sum_{p \in P_j^k} q_k x_p \geq r_j, \quad \forall j \in N; \tag{3}$$

$$\sum_{k \in K_j} \sum_{p \in P_j^k} v_k x_p \leq u_j, \quad \forall j \in N; \tag{4}$$

$$x_p \in \mathbb{Z}_+, \quad \forall p \in P^k, \forall k \in K. \tag{5}$$

In Objective (1), x_p gives the flow amount along path p and c_p is the cost of using a unit of the flow represented by p . Constraints (2) require the used flows to be no more than the total amount b_k for each commodity k . Constraints (3) ask the flow provision (q_k being the unit provision contribution of commodity k) for each node to be no less than the demand r_j . Constraints (4) limit the used flow resource at j to be no more the capacity u_j , where v_k is the flow resource consumed by a unit flow of commodity k . Finally Constraints (5) give the variable domain.

The above formulation can have flow demerging/remerging at nodes. If no special control is taken into account, UDR may incur additional costs and result in impractical solutions unacceptable for real-world operations.

Let x be the overall flow solution over \mathcal{G} given by model (M_x) and $x(\bar{\mathcal{G}})$ be the solution over a subgraph $\bar{\mathcal{G}} \subset \mathcal{G}$. Let $z(x)$ be the objective value of solution x and $\#[x]$ be the number of M/D operations from x . Assume that the cost on a single merging or demerging operation can be calculated and is comparable with other costs in the objective. Let $\zeta(x)$ denote the cost from the M/D operations in solution x .

Definition 1. A feasible solution x to (M_x) is said to have unnecessary demerging/remerging operations if there exists at least another feasible solution x' such that $|z(x) - z(x')| \leq \epsilon_Q$, but $z(x) + \zeta(x) > z(x') + \zeta(x')$. $\epsilon_Q \geq 0$ is an objective quality tolerance value defined according to the relations between the costs on M/D operations and other costs.

Figure 2 gives an example of UDR between solutions x (left) and x' (right) over the same subgraph of \mathcal{G} consisted of four intermediate nodes $\{i, j, m, n\}$ and four connection arcs $\{(i, m), (i, n), (j, m), (j, n)\} \in A$ (without commodity attributes). Suppose x and x' only differ in the flows through the four arcs. Four flows of amount one—①, ②, ③, and ④ of the same commodity compatible with all the nodes—are involved, and every node needs a flow quantity of two. The four arcs have the same cost, and the objective is to minimize a weighted sum of the amount of routed flows and the costs of used arcs multiplied by the arc flow amount. In x the two arrival flow combinations ① + ② and ③ + ④ both demerge and remerge into two new flow combinations ① + ③ and ② + ④. While in x' , the merged flow combinations covering nodes i and j continue to serve node m and n . Now we have $z(x) = z(x')$ but $\#[x] = \#[x'] + 4$. Therefore $z(x) + \zeta(x) > z(x') + \zeta(x')$, and x has UDR.

When planning is done by manual practitioners and M/D activities incur costs, the UDR as in Figure 2A is unlikely to appear, as it is simply too clumsy. However, when an optimization model such as (M_x) is used, without special control, UDR is possible to appear. This is because the path flow variables x_p (or equivalently, arc flow variables x_a if an arc based formulation is used) alone are difficult to measure the exact number of M/D operations. In real-world cases, UDR in optimization models such as (M_x) may raise serious problems on the practicality of the results. For example, Figure 3 shows a scheduling result of how trains are connected at Leeds Station given by a TUSP model similar to (M_x) that does not consider UDR. The left side shows the arrival trains, and the right shows the departure trains. Each link from an arrival train to a departure train corresponds to an arc in A , and the used arcs are marked in yellow. Extensive UDR operations can be observed. Figure 3 also illustrates the five characteristics mentioned in Section 1, where the location l is a train station and the nodes are trips to be covered by train unit vehicles as flows. It can be observed that a bigraph (I_l, J_l, A_l) is formed by the arrival trains as I_l and the departure trains as J_l , with single directional arcs from the former to the latter.

To solve the above problem of UDR, we discuss two alternative remedies. The first one is to introduce additional fixed-charge variables; the second one is to use heuristic branching based on the concept of *flow potential* at the BB stage. Finally, we

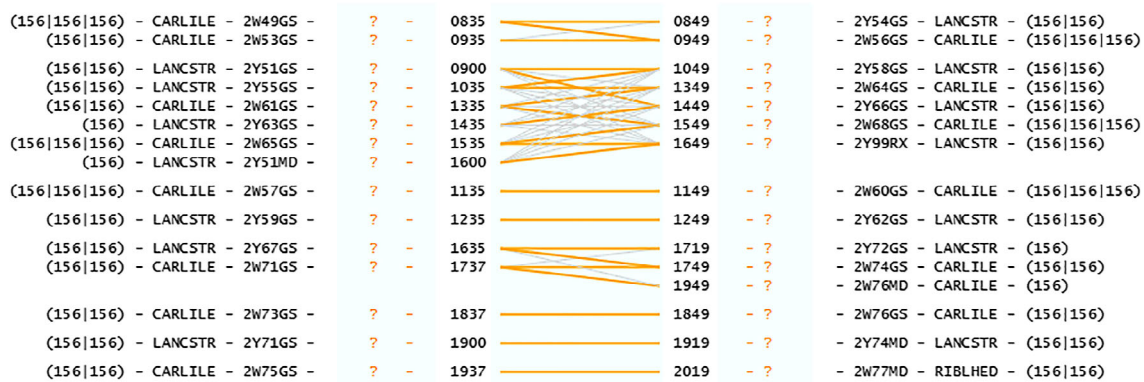


FIGURE 3 Example of UDR at Leeds station (Northern Class 156 dataset) [Color figure can be viewed at wileyonlinelibrary.com]

propose a newly developed framework by warm-starting the fixed-charge model with an initial feasible solution produced from the single variable model with branching heuristics.

4 | FIXED-CHARGE VARIABLE MODEL

This section briefly discusses the integer FCMF model proposed in [22]. We call the arcs in \mathcal{A} having no commodity attribute *omni-arcs*² and the arcs in $\mathcal{A}^k, k \in K$ having commodity attribute *commodity-arcs*. To minimize UDR, based on the original integer multi-commodity flow model (M_x) , fixed-charge variables $y_a \in \{0, 1\}, a \in \mathcal{A}$ are added to indicate whether an omni-arc a is used ($y_a = 1$) or not ($y_a = 0$). The number of M/D operations can be straightforwardly calculated from the fixed-charge variables, and the number of M/D operations can be minimized as an additional term in the objective.

Let x_a^k be the flow amount over the commodity-arc with commodity k at arc a , and let x_a be the flow amount over omni-arc a , that is, $x_a = \sum_{k \in K} x_a^k$. Note that the fixed-charge variables are defined such that $y_a = 1$ if $x_a > 0$ and $y_a = 0$ if $x_a = 0$. Let $\delta_-(j)$ and $\delta_+(j)$ be the omni-arcs terminated at and originated from of j , respectively. For a node $j \in N$, the number of merging operations can be calculated by $\#_j^C = \sum_{a \in \delta_-(j)} y_a - 1$, and similarly the number of demerging operations is $\#_j^D = \sum_{a \in \delta_+(j)} y_a - 1$. Therefore, the total number of M/D operations over the entire DAG is

$$\begin{aligned} \Sigma\# &= \sum_{j \in N} (\#_j^C + \#_j^D) \\ &= \sum_{j \in N} \left[\left(\sum_{a \in \delta_-(j)} y_a - 1 \right) + \left(\sum_{a \in \delta_+(j)} y_a - 1 \right) \right] \\ &= \sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a - 2 |N|. \end{aligned} \tag{6}$$

When used as a term in the objective, to avoid possible negative values, it is sufficient to only minimize $\sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a$. Then the integer FCMF formulation as an enhanced version of model (M_x) is given as,

$$(M_{xy}) \quad \min \quad \sum_{k \in K} \sum_{p \in P^k} c_p x_p + W \left(\sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a \right) \tag{7}$$

$$\text{s.t.} \quad \sum_{p \in P^k} x_p \leq b_k, \quad \forall k \in K; \tag{8}$$

$$\sum_{k \in K} \sum_{p \in P_j^k} q_k x_p \geq r_j, \quad \forall j \in N; \tag{9}$$

$$\sum_{k \in K} \sum_{p \in P_j^k} v_k x_p \leq u_j, \quad \forall j \in N; \tag{10}$$

$$\sum_{k \in K} \sum_{p \in P_a^k} x_p \leq u_a y_a, \quad \forall a \in \mathcal{A}; \tag{11}$$

²Omni-arcs are as known as “block-arcs” in the TUSP, where a connection arc represents a block of coupled units, that is, a unit block, during a turnaround time between two trains, regardless of the details of the block. The definition of a unit block is first introduced in [11].

$$x_p \in \mathbb{Z}_+, \quad \forall p \in P^k, \forall k \in K; \tag{12}$$

$$y_a \in \{0, 1\}, \quad \forall a \in \mathcal{A}. \tag{13}$$

The newly added Constraints (11) are used to force y_a to be binary where u_a is the largest value the flow amount over arc a can take as implied by $x_a = \sum_{k \in K} \sum_{p \in P_a^k} x_p$, where P_a^k is the set of paths associated with commodity k containing arc a .

The term $\sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a$ can accurately capture the total number of M/D operations in a feasible solution, making the optimal solution in (M_{xy}) exact by minimizing both the original operational cost in the first term and the cost due to M/D operations in the second term. By setting $z(x) = \sum_{k \in K} \sum_{p \in P^k} c_p x_p$ and by slightly violating the notational convention and setting $\zeta(x) = W \left(\sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a \right)$ where y is determined afterwards based on x , model (M_{xy}) is able to remove UDR operations as in Definition 1.

Nonetheless, the additional variables/constraints with respect to y_a in (M_{xy}) can significantly increase the difficulty in solving the model. For instance, high-quality solutions are difficult to find using variants of (M_{xy}) proposed in [22] when the number of nodes is over 200.

5 | A HEURISTIC BRANCHING SCHEME FOR REMOVING UNNECESSARY DEMERGING/REMERGING FOR MODEL (M_X)

Recall Definition 1 that UDR in a feasible x happens when there exists another feasible x' such that $|z(x) - z(x')| \leq \epsilon_Q$ but $z(x) + \zeta(x) > z(x') + \zeta(x')$. In addition, each node is associated with a location, and a bigraph can be formed at each location. We show that arcs with “positive flow potentials” have strong correlations with UDR. Given a feasible solution, by heuristically detecting such arcs with positive flow potentials and by enforcing full utilization where appropriate, UDR operations can be significantly reduced. Verification on whether fully utilizing an arc leads to a reduction of M/D is important and two approaches are thus proposed: (i) Conducting a verification within a subset of nodes of the location bigraph, that is, a “FCG,” and (ii) Conducting a verification within all nodes of a location using a location sub-model. Being heuristic though, this process of arc full utilization is embedded into the BB tree of the customized branch-and-price solver designed for models similar to (M_X) to produce high-quality solutions by taking advantage of BB.

5.1 | Flow potential and total number of M/D operations

The concept of flow potential is proposed in [21]. Let w_j be the flow amount on $j \in N$ in a feasible solution x . For each intermediate arc $a = (i, j) \in A$, let $\mu_a = \min(w_i, w_j)$ and $M_a = \max(w_i, w_j)$. For each source/sink arc $a \in (s, j)$ or $a \in (j, t)$, $j \in N$, define $\mu_a = w_j$. M_a is not needed for source/sink arcs.

Definition 2. Given a feasible solution x over $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, the flow potential over arc $a \in \mathcal{A}$ is defined as $\pi_a = \begin{cases} \mu_a - x_a, & x_a > 0 \\ 0, & x_a = 0 \end{cases}$.

Figure 4 gives an example of the flow potential over arc (i, j) . The flow amount over (i, j) is $x_{ij} = 1$. The flow amounts on the two endpoints are $w_i = \mu_{ij} = 2$, and $w_j = M_{ij} = 3$. This means that one additional flow over (i, j) could have been exploited, such that the flow potential on (i, j) is $\pi_{ij} = \mu_{ij} - x_{ij} = 2 - 1 = 1$. The definition of “flow potential,” although sharing some similarity, should not be confused with the commonly used “flow residual” as in Augmenting Path Algorithm for the max-flow problem [1]. The two have different definitions and purposes where regarding flow potential, we only focus on unused capacity over used arcs, not all arcs (as unused arcs all have zero potentials), and we only use flow potential for identifying UDR operations, rather than finding a max-flow.

A used arc with a positive flow potential is called *underutilized* and a used arc with a zero flow potential is called *fully utilized*. Intuitively, a positive flow potential over an arc indicates that the arc’s capacity is not fully exploited, and the flows that cannot use the residual capacity (i.e., the potential π_a) may have to use other arcs, implying a possible increase in the number

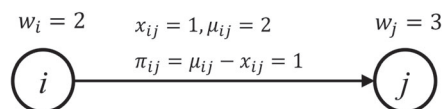


FIGURE 4 An example of flow potential $\pi_{ij} = 1$ over arc (i, j)

TABLE 1 Values regarding flow potentials on Figure 2
($c_a = 2, \forall a$)

	$a:$	(i, m)	(i, n)	(j, m)	(j, n)
(a)	x_a	1	1	1	1
Unnecessary	π_a	1	1	1	1
(b)	x_a	2	0	0	2
Non-unnecessary	π_a	0	0	0	0

of M/D operations. To illustrate this, relevant flow values in Figure 2 are summarized in Table 1, where a potential relationship between UDR and positive flow potential can be observed. Based on Definition 2, we can represent fixed-charge variables in terms of μ_a, x_a and π_a , that is $y_a = \frac{\pi_a + x_a}{\mu_a}$. Therefore, by employing (6), the total number of M/D operations, $\Sigma\#$, can be expressed in terms of flow potentials:

$$\begin{aligned}
 \Sigma\# &= \sum_{a \in A_0} y_a + 2 \sum_{a \in A} y_a - 2 |N| \\
 &= \sum_{a \in A_0} \frac{\pi_a + x_a}{\mu_a} + 2 \sum_{a \in A} \frac{\pi_a + x_a}{\mu_a} - \sum_{j \in N} \sum_{a \in \delta_{\pm}(j)} \frac{x_a}{w_j} \\
 &= \sum_{a \in A_0} \frac{\pi_a + x_a}{\mu_a} + 2 \sum_{a \in A} \frac{\pi_a + x_a}{\mu_a} - \sum_{j \in N} \left(\sum_{a \in \delta_{\pm}(j) \cap A_0} \frac{x_a}{\mu_a} \right. \\
 &\quad \left. + \sum_{a \in \delta_{\pm}(j) \cap A: w_j = \mu_a \neq M_a} \frac{x_a}{\mu_a} + \sum_{a \in \delta_{\pm}(j) \cap A: w_j = M_a \neq \mu_a} \frac{x_a}{M_a} + \sum_{a \in \delta_{\pm}(j) \cap A: w_j = \mu_a = M_a} \frac{x_a}{\mu_a} \right) \\
 &= \sum_{a \in A_0} \frac{\pi_a + x_a}{\mu_a} + 2 \sum_{a \in A} \frac{\pi_a + x_a}{\mu_a} - \left[\sum_{a \in A_0} \frac{x_a}{\mu_a} + \sum_{a \in A: \mu_a \neq M_a} \left(\frac{1}{\mu_a} + \frac{1}{M_a} \right) x_a + \sum_{a \in A: \mu_a = M_a} \frac{2x_a}{\mu_a} \right] \\
 &= \sum_{a \in A_0} \frac{\pi_a + x_a}{\mu_a} + 2 \sum_{a \in A} \frac{\pi_a + x_a}{\mu_a} - \left[\sum_{a \in A_0} \frac{x_a}{\mu_a} + \sum_{a \in A} \left(\frac{1}{\mu_a} + \frac{1}{M_a} \right) x_a \right] \\
 &= \sum_{a \in A_0} \frac{\pi_a}{\mu_a} + 2 \sum_{a \in A} \frac{\pi_a}{\mu_a} + \sum_{a \in A} \left(\frac{1}{\mu_a} - \frac{1}{M_a} \right) x_a. \tag{14}
 \end{aligned}$$

See the Appendix for a detailed explanation on how node flows (w_j) are replaced by two endpoint flows μ_a and M_a using an illustrative example.

An arc (i, j) in a feasible flow solution is called balanced if $w_i = w_j$ and unbalanced if $w_i \neq w_j$. From (14), it can be seen that the total number of M/D operations consists of two parts. The first part $\sum_{a \in A_0} \frac{x_a}{\mu_a} + 2 \sum_{a \in A} \frac{x_a}{\mu_a}$ relates to underutilized arcs and the second part $\sum_{a \in A} \left(\frac{1}{\mu_a} - \frac{1}{M_a} \right) x_a$ relates to unbalanced arcs. If an arc is unbalanced, M/D operations are inevitable. If we assume that $w_j, \forall j \in N$ is already the best provision for all nodes, then M/D caused by unbalanced arcs may be regarded as “reasonable.” On the other hand, underutilized arcs tend to be more responsible for UDR where better alternatives can be constructed to retain exactly the same flow provision for the relevant nodes. Figure 2 and Table 1 give such an example.

Equation (14) does not guarantee that fully utilizing an arbitrary underutilized arc will definitely reduce the number of M/D operations, since a decrease on the flow amount over one arc may cause an increase on another arc’s flow. Therefore, it is important to further explore under which conditions such an action will ensure a decrease on the number of UDR.

5.2 | Arc full utilization and improvement checking

5.2.1 | Flow connection group

Recall that in our problem, each arc a has an associated location $l_a \in L_G$, and a bigraph B_l can be correspondingly derived at each location l . A *flow connection bigraph* derived from a *used* connection arc $a = (i, j) \in A$ in solution x ($x_a > 0$), denoted by $B(x_a) = (I, J, A_{IJ})$ as a subset of the location bigraph B_{l_a} , is a bipartite graph formed by all arcs and nodes “having relevant flows” with x_a at the location associated with arc a that can be separated from flows of other nodes and arcs. Additionally, information on node flow amount of the relevant nodes $w_i, w_j, i \in I, j \in J$ is also included in a flow connection bigraph. I, J are called sending nodes and receiving nodes, respectively, and $A_{IJ} := \{(i, j) \in A \mid i \in I, j \in J\}$ is the set of arcs between the sending and receiving nodes. The precise definition of a flow connection bigraph $B(x_e), e = (i, j)$ is algorithmically given in Algorithm 1. Note that the source and sink s, t are treated differently from intermediate nodes $j \in N$. Within the same flow connection bigraph

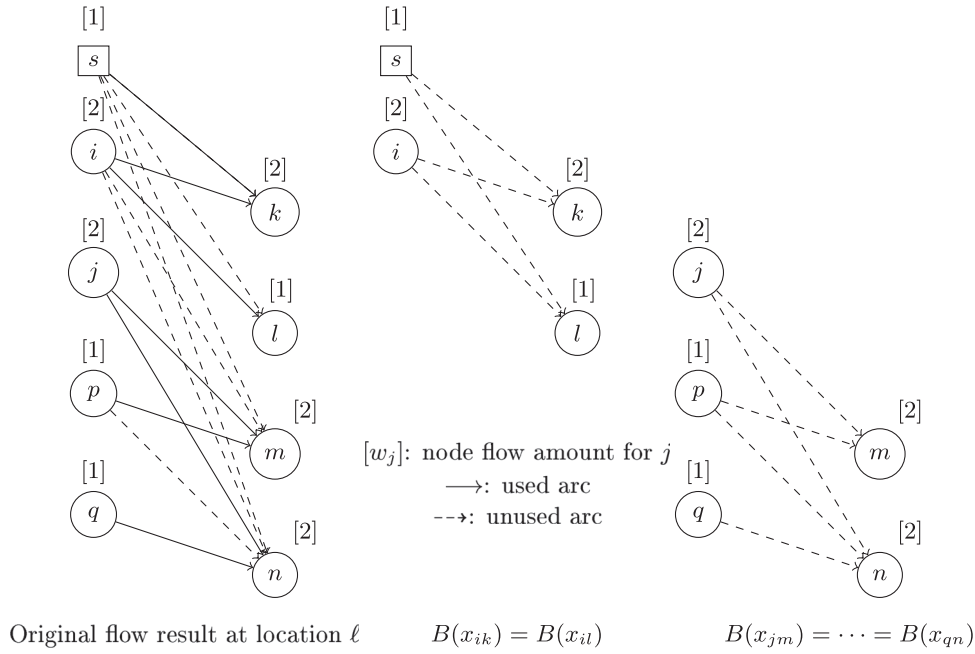


FIGURE 5 Example of two flow connection bigraphs derived from the same location ℓ that is associated with all arcs above

$B(x_a) = (I, J, A_{IJ})$, all flowed arcs in A_{IJ} share the same flow connection bigraph, that is, $B(x_{a_1}) = B(x_{a_2}) = \dots = B(x_{a_n}), \forall a_r \in A_{IJ}, x_{a_r} > 0$. Finally from Algorithm 1, it can be verified that for any flow connection bigraph $(I, J, A_{IJ}), \sum_{i \in I} w_i = \sum_{j \in J} w_j$. Figure 5 gives an example of how two flow connection bigraphs are derived from a flow solution at a location ℓ associated with eight nodes i, j, k, l, m, n, p, q , and the arcs among them. Note that since a source arc (s, k) is used ($x_{sk} = 1$), s should be included into the sending nodes of $B(x_{ik})$ with $w_s = 1$.

Algorithm 1. Creating $B(x_e), e = (i, j) \in A$ based on flow solution $x (x_e > 0)$

Let $V^-(e), V^+(e)$ be the start and end node of arc e .

$I := \{i\}, J := \{j\}, w_i := x_{ij}; w_j := x_{ij}$.

repeat

for all $u \in I, e \in \delta_+(u)$

if $x_e > 0$ and $V^+(e) \notin J$ and $V^+(e) \notin \{s, t\}$

$J + = \{V^+(e)\}, w_u + = x_e$

for all $v \in J, e \in \delta_-(v)$

if $x_e > 0$ and $V^-(e) \notin I$ and $V^-(e) \notin \{s, t\}$

$I + = \{V^-(e)\}, w_v + = x_e$

until No new nodes can be added to I, J

$w_s := 0, w_t := 0$

for all $u \in I$

if $x_{u,t} > 0$

$J + = \{t\}, w_u + = x_{ut}, w_t + = x_{ut}$

for all $v \in J$

if $x_{s,v} > 0$

$I + = \{s\}, w_v + = x_{sv}, w_s + = x_{sv}$

for all $u \in I, v \in J$

if $\exists (u, v) \in \mathcal{A}$

$A_{IJ} + = \{(u, v)\}$

A flow solution over a flow connection bigraph is called an FCG. More precisely:

Definition 3. An FCG over a flow connection bigraph $B = (I, J, A_{IJ})$, denoted by $x(B) = (x_a)_{a \in A_{IJ}}$, is a subset of a flow solution based on B with the same flow provision $w_j, \forall i \in I \cup J$ as prescribed in B .

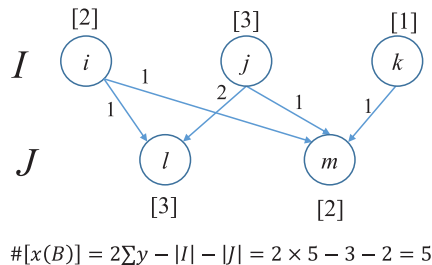


FIGURE 6 An example of Equation (16) [Color figure can be viewed at wileyonlinelibrary.com]

The number of M/D operations within an FCG $x(B)$, $B = (I, J, A_{IJ})$, denoted by $\#[x(B)]$, can be calculated by

$$\begin{aligned} \#[x(B)] &= \sum_{i \in \Lambda\{s\}} \left(\sum_{a \in \delta_+(i)} y_a - 1 \right) + \sum_{j \in \Lambda\{t\}} \left(\sum_{a \in \delta_-(j)} y_a - 1 \right) \\ &= \sum_{a \in A_{IJ} \cap A_0} y_a + 2 \sum_{a \in A_{IJ} \cap A} y_a - |\Lambda\{s\}| - |\Lambda\{t\}|. \end{aligned} \tag{15}$$

When there is no s and t contained in an FCG, the above can be further simplified into

$$\begin{aligned} \#[x(B)] &= \sum_{i \in I} \left(\sum_{a \in \delta_+(i)} y_a - 1 \right) + \sum_{j \in J} \left(\sum_{a \in \delta_-(j)} y_a - 1 \right) \\ &= 2 \sum_{a \in A_{IJ}} y_a - |I| - |J|. \end{aligned} \tag{16}$$

Figure 6 gives an example of Equation (16). Note that in (15) and (16), only M/D operations within the current location, that is, those yielded by using the arcs in A_{IJ} , are concerned in the above definition. There can be M/D operations prior to I or beyond J in x , but they are not included in calculating $\#[x(B)]$. We call an FCG without s and t as the case in (16) an *interior FCG*.

In a feasible flow solution x , let w_j^k be the flow amount achieved by commodity k at j and let w_j be the flow amount of all commodities achieved at node j , that is, $w_j = \sum_{k \in K} w_j^k$. The flow contribution in satisfying the demand at node j achieved by x is denoted by $\rho_j := \sum_{k \in K} q^k w_j^k$, where q^k is the unit contribution of commodity k . Two flow solutions x and x' are said to have *equivalent flow contribution* on j , denoted by $w_j \cong w_{j'}$, if (i) $w_j = w_{j'}$, (ii) $\rho_j \geq r_j$ and $\rho_{j'} \geq r_{j'}$, where r_j is the flow demand on j as defined in (M_x) . A special case of equivalent flow contribution is that all the commodities involved are “interchangeable” within an FCG in providing enough flow contribution (i.e., flow amount multiplied by q^k) while keeping the flow amount per node unchanged. In other words, condition $w_j \cong w_{j'}$ can be simplified to $w_j = w_{j'}$ if commodities are interchangeable. This special case is referred to as “commodity interchangeability condition.” If further requirement $w_j^k = (w_{j'}^k)'$, $\forall k \in K$ is imposed, then x and x' are said to have *identical flow contribution* on j , denoted by $w_j \equiv w_{j'}$. Indeed this is a stronger condition since $w_j \cong w_{j'}$ holds as long as $w_j \equiv w_{j'}$.

Let $B = (I, J, A_{IJ})$. We use $w[x(B)] \cong w[x'(B)]$ ($w[x(B)] = w[x'(B)]$) if all commodities are interchangeable) to denote the case where $w_j \cong w_{j'}, \forall j \in I \cup J$ ($w_j = w_{j'}, \forall j \in I \cup J$ if all commodities are interchangeable). Similarly, $w[x(B)] \equiv w[x'(B)]$ is used to denote the case where $w_j \equiv w_{j'}, \forall j \in I \cup J$.

Recall from (14) that arcs with positive flow potentials are a major source of UDR in terms of excessive M/D. If appropriate underutilized arcs ($a, \pi_a > 0$) in a feasible integer solution of (M_x) are identified and modified to be fully utilized ($x_a = \mu_a$), UDR is likely to be reduced. FCGs are useful for the above process, where in each iteration we focus on an individual FCG and fully utilize one of its underutilized arcs while keeping all the other nodes’ flow contribution still equivalently provided. The advantage of restricting the scope to an FCG is that it is possible to obtain some sufficient conditions for guaranteeing an improvement. Before going into details of such sufficient conditions, we first show that an improvement cannot be always guaranteed by randomly forcing one arc to be fully utilized within an FCG.

Let us consider a case where the “commodity interchangeability condition” is met, such that it is sufficient to only keep the flow amount over the nodes unchanged. Given an FCG $x(B)$ based on $B = (I, J, A_{IJ})$ with $\pi_a > 0$, and let $x'(B)$ be a resulting FCG by fully utilizing a , that is, let $x'_a = \mu_a$. When comparing the two FCGs, we assume that the parts outside B are not affected since the node flow amount $w_j, \forall j \in I \cup J$ are kept unchanged. We have the following observations that a single operation by fully utilizing a may not guarantee: (i) $x'(B)$ is feasible, (ii) $\#[x'(B)] < \#[x(B)]$.

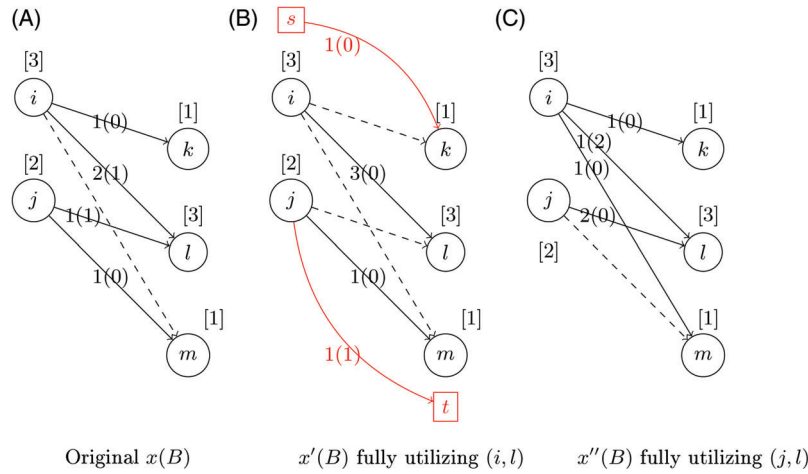


FIGURE 7 Failure examples in arc full utilization. Used arcs are in solid lines and unused are in dashed lines. A flowed arc a is labeled with “ $x_a(\pi_a)$,” and a node j is labeled with “ $[w_j]$ ” [Color figure can be viewed at wileyonlinelibrary.com]

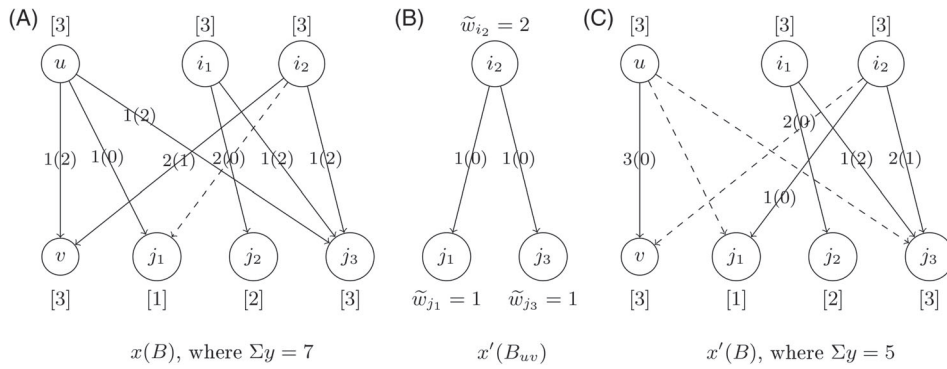


FIGURE 8 Example on Theorem 1 where arc full utilization on arc (u, v) is guaranteed by complete potential graph B_{uv}

Figure 7 gives two examples corresponding to the above two facts. Originally $x(B)$ has two underutilized arcs (i, l) and (j, l) with $\#[x(B)] = 3$ (see (a)). Fully utilizing (i, l) leads to $x'(B)$, which is infeasible within B itself³ (see (b)). On the other hand, fully utilizing (j, l) leads to $x''(B)$, where no improvement is given as $\#[x''(B)] = \#[x(B)] = 3$ (see (c)).

5.2.2 | Sufficient conditions of guaranteed improvements by full utilization within an interior FCG

A bigraph $B = (I, J, A_{IJ})$ is complete if $\exists(i, j), \forall i \in I, \forall j \in J$. For denotational convenience, in this section w_j can be written as $w(j), j \in N$, and x_{ij} can be written as $x(i, j), (i, j) \in A$. Recall the classic result from the Transportation Problem with m sources and n sinks [8], we have the following Lemma 1.

Lemma 1. Assume that the “commodity interchangeability condition” is met. A basic feasible flow solution using at most $m + n - 1$ arcs can always be found in a complete flow connection bigraph $B = (I, J, A_{IJ}), I = \{i_1, i_2, \dots, i_m\}, J = \{j_1, j_2, \dots, j_n\}$, if and only if $\sum_{p=1}^m w(i_p) = \sum_{q=1}^n w(j_q)$.

Given an underutilized balanced arc $(u, v) \in A_{IJ}$ in an FCG $x(B), B = (I, J, A_{IJ})$, let $I_{uv} := \{i \in I \setminus \{u\} | x_{iv} > 0\}, J_{uv} := \{j \in J \setminus \{v\} | x_{uj} > 0\}$ and $A_{uv} := \{(i, j) \in A_{IJ} \setminus \{(u, v)\} | i \in I_{uv}, j \in J_{uv}\}$. Let $\tilde{w}_i := x_{iv}, \forall i \in I_{uv}$ and $\tilde{w}_j := x_{uj}, \forall j \in J_{uv}$. The resulting bipartite subgraph $B_{uv} := (I_{uv}, J_{uv}, A_{uv})$ with the adjusted node demand \tilde{w}_i, \tilde{w}_j is called the potential graph of $x(B)$ derived from (u, v) . Figure 8 gives an example where B_{uv} (middle) is the potential graph of $x(B)$ derived from (u, v) (left).

³To make $x'(B)$ feasible without affecting the flow contributions on the nodes outside B , source arc (s, k) and sink arc (j, t) can be used as depicted, which is generally undesirable as they indicate an increase in the number of used commodity flows by one.

Fully utilizing balanced arcs

The following Theorem 1 is proposed in [21] as a sufficient condition for commodities that are interchangeable and for an underutilized arc that is balanced.

Theorem 1. *Let $x(B)$ be an interior FCG with a balanced underutilized arc (u,v) . Assume that the “commodity interchangeability condition” is satisfied. There exists at least one $x'(B)$ by fully utilizing (u,v) such that $x'(B)$ is feasible, $w[x(B)] = w[x'(B)]$, and $\#[x'(B)] < \#[x(B)]$, if the potential graph $B_{uv} = (I_{uv}, J_{uv}, A_{uv})$ is complete.*

Figure 8 gives an example of Theorem 1. The original solution $x(B)$ over $B = (I, J, A_{IJ})$ is shown in (a), where

$$I = \{u, i_1, i_2\}, J = \{v, j_1, j_2, j_3\},$$

$$A_{IJ} = \{(u, v), (u, j_1), (u, j_3), (i_1, j_2), (i_1, j_3), (i_2, v), (i_2, j_1), (i_2, j_3)\}.$$

Figure 8B gives the potential graph derived from (u,v) , where $I_{uv} = \{i_2\}, J_{uv} = \{j_1, j_3\}, A_{uv} = \{(i_2, j_1), (i_2, j_3)\}$ with the adjusted node demands. It is complete and has a feasible flow solution as shown. Figure 8C shows $x'(B)$ as a result of fully utilizing arc (u,v) and incorporating $x'(B_{uv})$ with $x(B) \setminus x(\delta_+(u) \cup \delta_-(v))$. Note that arc $(i_2, j_3) \in B_{uv}$ has already been used in $x(B)$. In $x'(B)$, arc (u,v) are successfully separated from B . Additionally, it can be verified that further fully utilizing arc (i_2, j_3) will not give an improvement, since the potential graph $B_{i_2, j_3} = (\{i_1\}, \{j_1\}, \emptyset)$ is not complete. Similarly, further fully utilizing (i_1, j_3) will not be successful, since $B_{i_1, j_3} = (\{i_2\}, \{j_2\}, \emptyset)$ is not complete either.

When an interior FCG does not satisfy the commodity interchangeability condition, which is more often seen in real practices, Theorem 1 cannot be applied. Here we give a sufficient condition that can be used in the FCGs where (i) commodities are not necessarily interchangeable, (ii) the component of each node strictly remains the same in the new FCG. An underutilized arc (i,j) that is not only balanced ($w_i = w_j$) but also has exactly the same component on its two end points (i.e., $w_i \equiv w_j$) is called *strongly balanced*. Suppose we want to fully utilize a strongly balanced arc within an FCG while strictly retaining the component of each node. Let C_{\max} denote the maximum flow amount allowed at nodes (which is implicitly included in Constraints (4)). We have the following conclusions.

Theorem 2. *Let $x(B)$ be an interior FCG with a strongly balanced underutilized arc (u,v) . Let C_{\max} be the maximum integer flow amount that can be used at the nodes.*

1. *When $C_{\max} \in \{2,3,4\}$, any feasible solution $x'(B)$ that satisfies $w[x'(B)] \equiv w[x(B)]$ by fully utilizing (u,v) has a fewer number of M/D operations, that is, $\#[x'(B)] < \#[x(B)]$, if the potential graph $B_{uv} = (I_{uv}, J_{uv}, A_{uv})$ is complete.*
2. *When $C_{\max} \geq 5$, the above property (1) no longer holds.*

Proof. Similar reasoning as in the proof for Theorem 1 [21] can be used, such that an improvement can be guaranteed if there exists an arc assignment $y'_a, a \in A_{uv}$ where the number of used omni-arcs $\sum_{a \in A_{uv}} y'_a$ in the potential graph is less than the number of removed omni-arcs as a result of separating (u,v) from the rest parts of $x(B)$. This is equivalent to $\sum_{a \in A_{uv}} y'_a < |I_{uv}| + |J_{uv}|$. Let $\tilde{W} = \sum_{i \in I_{uv}} \tilde{w}_i = \sum_{j \in J_{uv}} \tilde{w}_j$ be the total number of flows in I_{uv} (or J_{uv}). Note that $1 \leq |I_{uv}| = |J_{uv}| \leq \tilde{W}, x_{uv} \geq 1$, and it is sufficient to always assume $w_u = w_v = x_{uv} + \tilde{W} = C_{\max}$.

1. The first conclusion can be verified by enumerating all possible scenarios. Note that since $w_u \equiv w_v$ and the potential graph B_{uv} is complete, for any flow in I_{uv} with its own commodity type, it can always be matched to at least one unit in J_{uv} .
 - i. When $C_{\max} = 2, \tilde{W} = 1, |I_{uv}| = |J_{uv}| = 1, \sum_{a \in A_{uv}} y'_a = 1$ is the only possibility, which satisfies $\sum_{a \in A_{uv}} y'_a < |I_{uv}| + |J_{uv}|$.
 - ii. When $C_{\max} = 3$, it only needs to consider $\tilde{W} = 2$ ($\tilde{W} = 1$ is similar to (i)), such that $1 \leq |I_{uv}| = |J_{uv}| \leq 2$. When at least one of $|I_{uv}|$ or $|J_{uv}|$ is 1 (without loss of generality, assume $|I_{uv}| = 1$), the only possible flow assignment on y is to use exactly $|J_{uv}|$ omni-arcs, such that $\sum_{a \in A_{uv}} y'_a = |I_{uv}| + |J_{uv}| - 1$. The last option is $|I_{uv}| = |J_{uv}| = 2$, giving $\sum_{a \in A_{uv}} y'_a = 2$.
 - iii. When $C_{\max} = 4$, it only needs to consider $\tilde{W} = 3$, such that $1 \leq |I_{uv}| = |J_{uv}| \leq 3$. When at least one of $|I_{uv}|$ or $|J_{uv}|$ is 1, similarly we have $\sum_{a \in A_{uv}} y'_a = |I_{uv}| + |J_{uv}| - 1$. When at least one of $|I_{uv}|$ or $|J_{uv}|$ is 3, then we have $\sum_{a \in A_{uv}} y'_a = 3$ such that $\sum_{a \in A_{uv}} y'_a < |I_{uv}| + |J_{uv}|$. The last option is $|I_{uv}| = |J_{uv}| = 2$. In this case the number of used omni-arcs must satisfy $\sum_{a \in A_{uv}} y'_a \leq 3$ since it is bounded by the fact that $\tilde{W} = 3$.

Therefore, for all possibilities when $C_{\max} \in \{2,3,4\}$, $\sum_{a \in A_{uv}} y'_a < |I_{uv}| + |J_{uv}|$ holds as long as $x'(B)$ is feasible and $w[x(B)] \equiv w[x'(B)]$.

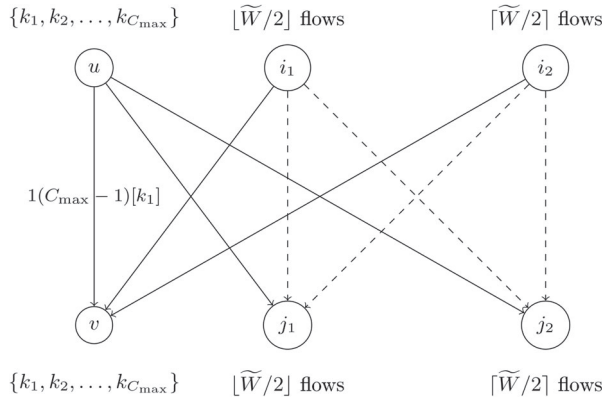


FIGURE 9 The FCG constructed as in the proof when $C_{\max} \geq 5$

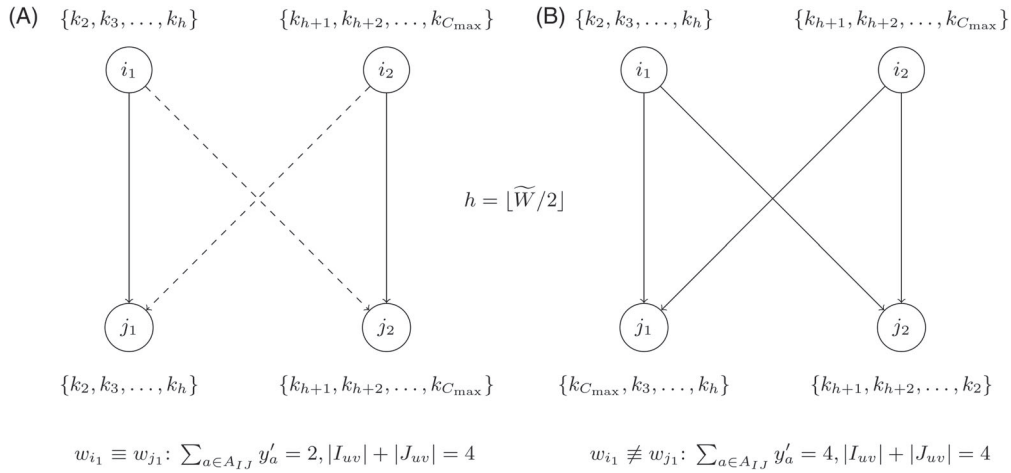


FIGURE 10 (A). A case giving an improvement. (B). A case without an improvement

2. For $C_{\max} \geq 5$, we prove that a counter example giving no improvement can always be found. Consider the special case of an $x(B)$ where u and v are consisted of C_{\max} different non-interchangeable commodities $k_1, k_2, \dots, k_{C_{\max}}$, each having a flow amount of one. Underutilized arc (u, v) is filled with one flow amount of commodity k_1 where $\pi_{uv} = C_{\max} - 1$. Therefore in B_{uv} , both I_{uv} and J_{uv} have a total flow amount of $\tilde{W} = C_{\max} - 1$, made up of commodities from $k_2, k_3, \dots, k_{C_{\max}}$ each occurring once and only once in I_{uv} and J_{uv} , respectively. To find a feasible flow in B_{uv} constructed as above, since all commodities are non-interchangeable, the only way to allocate one flow of commodity $k_r, r = 2, \dots, C_{\max}$ from I_{uv} is to match it to the corresponding flow of commodity k_r in J_{uv} .

Additionally we require that $|I_{uv}| = |J_{uv}| = 2$ where $I = \{i_1, i_2\}, J_{uv} = \{j_1, j_2\}$ in a way that $w_{i_1} = \lfloor \frac{\tilde{W}}{2} \rfloor, w_{i_2} = \lceil \frac{\tilde{W}}{2} \rceil$ and $w_{j_1} = \lfloor \frac{\tilde{W}}{2} \rfloor, w_{j_2} = \lceil \frac{\tilde{W}}{2} \rceil$, that is, the flow amounts in w_{i_1} and w_{i_2} (w_{j_1} and w_{j_2}) are as close as possible. Since $\tilde{W} \geq 4, \lfloor \frac{\tilde{W}}{2} \rfloor \geq 2$, meaning any node in B_{uv} should have a flow amount of at least two. Figure 9 gives an example of such an FCG.

Since $|I_{uv}| + |J_{uv}| = 4$, only a flow assignment with $\sum_{a \in A_{IJ}} y'_a < 4$ can guarantee an improvement. However this is only possible if $w_{i_1} \equiv w_{j_1}$ (which also implies $w_{i_2} \equiv w_{j_2}$), where $\sum_{a \in A_{IJ}} y'_a = 2$ can be achieved as shown on the left of Figure 10. As long as the condition $w_{i_1} \equiv w_{j_1}$ ($w_{i_2} \equiv w_{j_2}$) is not satisfied, since each node contains at least two unit flows ($\lfloor \frac{\tilde{W}}{2} \rfloor \geq 2$), all the four arcs in A_{IJ} will have to be used since any node needs to be connected with both the two nodes in the opposite, giving $\sum_{a \in A_{IJ}} y'_a = |I_{uv}| + |J_{uv}| = 4$. Such an example is given on the right of Figure 10, where k_2 and $k_{C_{\max}}$ are swapped between j_1 and j_2 compared with the potential graph on the left. Therefore, when $C_{\max} \geq 5$, the property as in (1) for $C_{\max} \in \{2, 3, 4\}$ will no longer hold. ■

In the TUSP, C_{\max} is the maximum number of units that can be coupled at a trip. As far as the authors are aware, $C_{\max} = 3$ is set for all the train operators we have collaborated with so far. Therefore, for real-world TUSPs, it generally only needs to consider Property (1) in Theorem 2 when unit types are not interchangeable within an interior FCG.

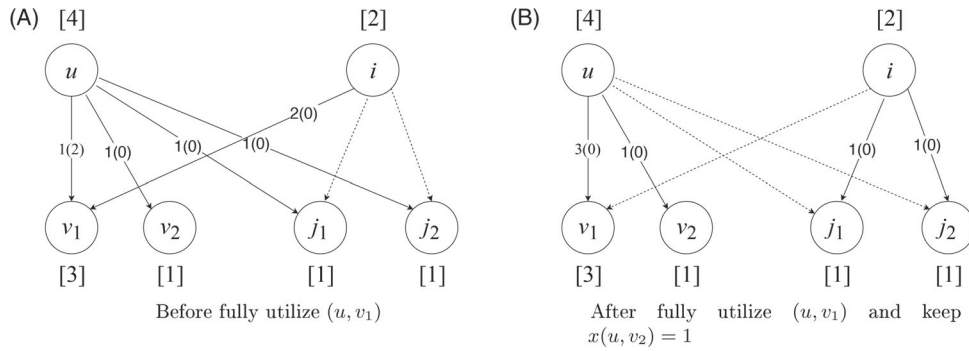


FIGURE 11 Fully utilization on unbalanced arc (u, v_1)

Fully utilizing unbalanced arcs

Both Theorems 1 and 2 only consider cases where the focused arc is balanced. Next we further extend the conclusions from the two theorems into general cases such that unbalanced arcs can also be considered appropriately. First let us observe an example as illustrated in Figure 11. In this FCG made with two sending nodes u, v and four receiving nodes v_1, v_2, j_1, j_2 , all arcs involved are unbalanced, and in the original solution on the left, arc (u, v_1) is underutilized with a potential of two. If we fully utilize (u, v_1) and keep $x(u, v_2) = 1$ unchanged, we will get a new FCG given on the right where the M/D number is reduced from four to two.

In fact, we can generalize the results in Theorem 1 into the following Theorem 3, which provides a useful way to detect such cases as in Figure 11 where fully utilizing an unbalanced arc will give an improvement. To formally state this theorem, we need to introduce the definition of a *balanced underutilized arc set solution*. Let $x(B)$ be an interior FCG based on $B = (I, J, A_{IJ})$. Suppose there is an arc set $U \times V \subset A_{IJ}$ ($U \subset I, V \subset J$) with $U = \{u_1, u_2, \dots, u_m\} \subset I$, where $0 \leq |U| < |I|$ and a set of receiving nodes $V = \{v_1, v_2, \dots, v_n\} \subset J$, where $0 \leq |V| < |J|$, and $|U| + |V| > 0$. $x(U \times V)$, the solution over arc set $U \times V$, is called balanced if $\sum_{p=1}^m w(u_p) = \sum_{q=1}^n w(v_q) := \mu_{UV}$, and is called underutilized if $\sum_{a \in U \times V} x_a < \mu_{UV}$, that is, at least one arc in $U \times V$ is underutilized (which can be either balanced or unbalanced). A flow potential for the arc set $U \times V$ can also be defined

$$\text{accordingly as } \pi_{UV} = \begin{cases} \mu_{UV} - \sum_{a \in U \times V} x_a, & \sum_{a \in U \times V} x_a > 0 \\ 0, & \sum_{a \in U \times V} x_a = 0. \end{cases}$$

We can also define the potential graph of $x(B)$ derived from $x(U \times V)$. Let $I_{UV} := \{i \in I \setminus \{U\} \mid x_{iv_q} > 0, q = 1, \dots, n\}, J_{UV} := \{j \in J \setminus \{V\} \mid x_{u_pj} > 0, p = 1, \dots, m\}$ and $A_{UV} := \{(i, j) \in A_{IJ} \setminus \{U \times V\} \mid i \in I_{UV}, j \in J_{UV}\}$. Let $\tilde{w}_i := x_{iv_q}, \forall i \in I_{UV}$ and $\tilde{w}_j := x_{u_pj}, \forall j \in J_{UV}$. The resulting bipartite subgraph $B_{UV} := (I_{UV}, J_{UV}, A_{UV})$ with the adjusted node demand \tilde{w}_i, \tilde{w}_j is called the *potential graph* of $x(B)$ derived from $U \times V$. For example, $B_{\{u\}\{v_1, v_2\}} = (\{i\}, \{j_1, j_2\}, \{(i, j_1), (i, j_2)\})$ is the potential graph of the FCG on the left of Figure 11 derived from the balanced underutilized arc set solution $x(\{u\} \times \{v_1, v_2\})$, with adjusted node demands $\tilde{w}_i = 2, \tilde{w}_{j_1} = \tilde{w}_{j_2} = 1$.

Theorem 3. *Let $x(B)$ be an interior FCG with a balanced underutilized arc set solution $x(U \times V)$. Assume that the “commodity interchangeability condition” is satisfied. There exists at least one $x'(B)$ by fully utilizing all underutilized arcs in $U \times V$ such that $x'(B)$ is feasible, $w[x(B)] = w[x'(B)]$ and $\#[x'(B)] < \#[x(B)]$, if the potential graph $B_{UV} = (I_{UV}, J_{UV}, A_{UV})$ is complete.*

Proof. The proof is similar to the proof for Theorem 1 in [21], by noticing that the number of removed omni-arcs, $|I_{UV}| + |J_{UV}|$, is always greater than the possibly largest number of newly used arcs in $B_{UV} = (I_{UV}, J_{UV}, A_{UV})$, which is $|I_{UV}| + |J_{UV}| - 1$, since the potential graph is complete and the result given by Lemma 1. ■

Remark. Note that if an underutilized arc in $x(U \times V)$ is balanced, then the arc itself can make an independent balanced underutilized arc set and can be excluded from $x(U \times V)$. Therefore, eventually we can exclude all balanced underutilized arcs and only keep the unbalanced arcs in a balanced underutilized arc set.

An example of the application of Theorem 3 has been already given in Figure 11, where we have a balanced arc set solution $x(\{u\} \times \{v_1, v_2\})$ with a potential graph

$$B_{\{u\}\{v_1, v_2\}}(\{i\}, \{j_1, j_2\}, \{(i, j_1), (i, j_2)\})$$

that is complete, and therefore an improvement can be made by fully utilizing arc (u, v_1) .

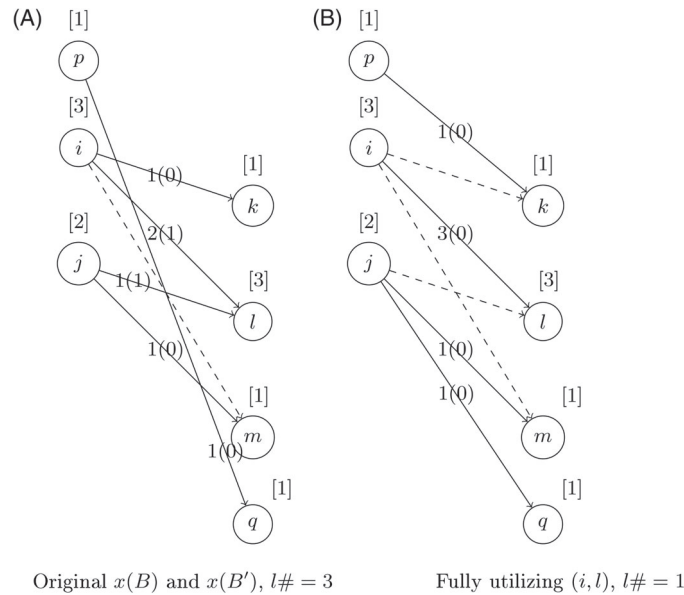


FIGURE 12 Local model example

We will not present a theorem analogous to Theorem 2 for general cases (including unbalanced underutilized arcs) with non-interchangeable unit types, since the possible scenarios will be rather problem-specific and it is more appropriate to remove UDR by the location pure fixed-charge model proposed in Section 5.2.3, which takes a holistic view over the location.

5.2.3 | A pure fixed-charge model for checking arc full utilization at a location

The sufficient conditions given in Theorems 1 to 3 can only guarantee an improvement within an interior FCG itself as a result of fully utilizing an underutilized arc. Nonetheless, from the real-world instances of the TUSP tested by us, cases corresponding to the above conditions are responsible for a large proportion of UDR. Despite of this, Theorems 1 to 3 have certain limitations in dealing with other scenarios. First, they are all sufficient conditions only for certain scenarios. Second, they can only handle interior FCGs, that is, without the source, sink and any source/sink arcs. Finally, even for interior FCGs, there are still cases where these conditions are not satisfied within an individual FCG, but a fewer number of M/D operations can still be achieved if more than one FCGs are considered at the same time.

Such an example is given in Figure 12. Consider the case given in Figure 7, where no resolution can be found for $x(B)$. If another FCG $x(B')$, $B' = (\{p\}, \{q\}, \{(p, q)\})$ within the same location is available (see Figure 12A), then by fully utilizing (i, l) , a reduction on M/D number can still be achieved as illustrated in Figure 12B.

The above example indicates that even fully utilizing an arc within an FCG that is not guaranteed by Theorems 1 to 3, by incorporating other FCGs, an improvement is still possible. However due to the “online” nature of the problem, it is unknown beforehand, among all the nodes at the same location, which other FCGs should be considered together. The remedy is to directly use a multi-commodity variant of the pure fixed-charge transportation problem based on an underutilized arc e and the location $l(e)$ associated with e , denoted by (MS_e) , and solved directly by a generic integer solver embedded into the BB process. It takes account of all nodes associated with $l(e)$ to determine whether an improvement can be made by fully utilizing arc e .

Before we go to the details of this model based on individual locations, we first briefly give a theoretical result on the computational complexity of both detecting the existence of improvement and minimizing the M/D without changing the flow amount and node components in an interior FCG, which justifies the use of (MS_e) , as no “efficient” method exists.

It is worth mentioning the results given by [26] about the constant pure fixed-charge transportation problem, where based on a standard transportation problem [8] with m supplies and n demands, fixed-charge variables y_{ij} , $i = 1, \dots, m, j = 1, \dots, n$ are also introduced to indicate whether an arc is used ($= 1$) or not ($= 0$). In the objective, only the fixed-charge variables with the same cost are considered (e.g., $\min \sum_{i=1}^m \sum_{j=1}^n f \cdot y_{ij}$). The authors have proved that the decision problem of the existence of degeneracy (i.e., if a fewer number of arcs than the size of standard basis, $m + n - 1$, take positive values in the solution of the transportation problem) is NP-complete, and further have shown that the pure fixed-charge transportation problem in minimizing the number of used arcs is NP-hard.

The problem of minimizing the number of used arcs in an interior FCG, as what Theorems 1 to 3 try to achieve, is not strictly a pure constant fixed-charge transportation problem, since certain arcs are forbidden to be used (not all trips can be

connected in the DAG \mathcal{G}). Little knowledge about the transportation problems with forbidden arcs is known, even the necessary and sufficient condition for the existence of a feasible solution [9,27]. Therefore, we need to regard the problem within an interior FCG as a generic fixed-charge network flow problem over an incomplete bipartite graph. The objective in this case is concave (with only the fixed-charge variables) and according to the result from [13], this problem with a single commodity is already NP-hard, mainly due to the concavity of the objective, and thus the NP-hardness of our problem, which is with multiple commodities. Therefore, no “easy” approach exists for finding the minimum number of used arcs in an interior FCG, and here we choose to use a generic integer linear program model (MS_e) over the entire location to deal with the more difficult cases where Theorems 1 to 3 cannot be applied to.

Let \mathcal{I}, \mathcal{J} be the set of origin and destination nodes on the bigraph implied by the location with s and t also included in \mathcal{I} and \mathcal{J} , respectively. $A_{\mathcal{I}\mathcal{J}} := \{(i, j) \in \mathcal{A} \mid i \in \mathcal{I}, j \in \mathcal{J}\} \cup \{(s, j) \in A_0 \mid j \in \mathcal{J}\} \cup \{(i, t) \in A_0 \mid i \in \mathcal{I}\}$ denotes the involved arcs. Two alternatives can be considered when designing model (MS_e): whether to equivalently keep the flow contributions (i.e., $w_j \cong w'_j, \forall j \in \mathcal{I} \cup \mathcal{J}$) or to identically keep the them (i.e., $w_j \equiv w'_j, \forall j \in \mathcal{I} \cup \mathcal{J}$). We choose the latter to ensure the nodes' compositions are retained as before, such that theoretically it is possible to keep the flow distribution in other parts of \mathcal{G} unaffected. Model (MS_e) is given in the following:

$$(MS_e) \quad \min_{x,y} \quad \#[l(e)] = \sum_{a \in A_{\mathcal{I}\mathcal{J}} \cap A_0} y_a + 2 \sum_{a \in A_{\mathcal{I}\mathcal{J}} \cap \mathcal{A}} y_a \quad (17)$$

$$\text{s.t.} \quad \sum_{a \in \delta_+(i) \cup A_{\mathcal{I}\mathcal{J}}} x_a^k = w_i^k, \quad \forall k \in K, \forall i \in \mathcal{I}; \quad (18)$$

$$\sum_{a \in \delta_-(j) \cup A_{\mathcal{I}\mathcal{J}}} x_a^k = w_j^k, \quad \forall k \in K, \forall j \in \mathcal{J}; \quad (19)$$

$$x_a^k \leq u_a y_a, \quad \forall a \in A_{\mathcal{I}\mathcal{J}}, \forall k \in K; \quad (20)$$

$$\sum_{k \in K} x_e^k = \mu_e; \quad (21)$$

$$\sum_{k \in K} \sum_{a \in A_0^k} x_a^k \leq \bar{f}[l(e)]; \quad (22)$$

$$x_a^k \in \mathbb{Z}_+, \quad \forall a \in A_{\mathcal{I}\mathcal{J}}, \forall k \in K; \quad (23)$$

$$y_a \in \{0, 1\}, \quad \forall a \in A_{\mathcal{I}\mathcal{J}}. \quad (24)$$

Two values derived from the current feasible solution \bar{x} (also the omni-arc variables \bar{y} derived a posteriori) with underutilized e over location $l(e)$ are recorded before running model (MS_e). They are the location M/D number indicator $\bar{\#}[l(e)] = \sum_{a \in A_{\mathcal{I}\mathcal{J}} \cap A_0} \bar{y}_a + 2 \sum_{a \in A_{\mathcal{I}\mathcal{J}} \cap \mathcal{A}} \bar{y}_a$ and the location total flow amount indicator $\bar{f}[l(e)] = \sum_{k \in K} \sum_{a \in A_0^k} \bar{x}_a^k$. In (MS_e), the Objective (17) is to minimize the number of M/D operations within location $l(e)$. Constraints (18) and (19) ensure the flow amounts per commodity on the nodes are unchanged in the new solution. Constraints (20) give correct values on y such that $y_a = 1$ if $x_a > 0$ and $y_a = 0$ if $x_a = 0$. Constraint (21) realizes the full utilization on arc e . Constraint (22) ensures the total flow amount at location $l(e)$ will not increase compared with the current solution. Finally Constraints (23) and (24) give the domains on the decision variables, where x_a gives the flow amount of commodity k over arc a and y_a is the omni-arc variable over arc a . If it is shown by (MS_e) that fully utilizing e within $l(e)$ will give a new feasible station solution with a fewer number of M/D operations (i.e., $\#[l(e)] < \bar{\#}[l(e)]$), then this full utilization is regarded as successful and will be approved. Otherwise, fully utilizing e will be disapproved by (MS_e).

Using model (MS_e) to verify arc full utilization might be less efficient than Theorems 1 to 3. Nonetheless, it verifies the utilization over an arc exactly within the entire location. To reach a trade-off between efficiency and accuracy, model (MS_e) will only be called if the three flow potential graph based methods cannot guarantee an improvement.

5.3 | AFUB for (M_x)

With the above two alternatives based on potential graphs and the local model for verifying arc full utilization, we propose a BB scheme to heuristically remove UDR operations, namely AFUB. It is a significant enhancement over the heuristic branching method in [21]. AFUB is designed for model (M_x) which does not have fixed-charge variables. Just like fractional arcs are to be

converted into integral ones in a standard BB process, here in AFUB, underutilized arcs are regarded as undesirable and will be branched to be fully utilized when there is evidence showing an improvement by doing so. An option is set on whether it is allowed to branch on “non-guaranteed” underutilized arcs, that is, arcs that have been rejected (if attempted) or have not been verified (if not attempted) by Theorems 1 to 3 or model (MS_e). Suppose an underutilized arc a is found and the strategy allows to further fully utilize it, then two branches will be formed under AFUB:

- On the left branch, a will be fully utilized.
- The right is a “back-up” branch, retaining everything as in the parent but with a labeled as “failed” one more time.

For the right branch, an arc can be “failed” for at most $F_{\max} > 0$ times, and after that it will no longer be considered for branching even if it is underutilized. This option is set to limit the BB tree to a reasonable size, and to also avoid the search falling into a repeated pattern or a small local neighborhood. The left full utilization branch always has a higher priority to be visited than the right back-up branch, regardless of the specific node selection strategies used (e.g., best-first, depth-first or hybridized).

AFUB will only be activated in the BB process after all the variables are integral. A solution will not be regarded as feasible, or “operable,” until all variables are integral *and* all underutilized arcs are either fully utilized or failed for more than F_{\max} times. Normal termination criteria for BB are still used, such as the global optimality gap, the maximum number of tree nodes and maximum time.

The final goal of AFUB is not to find a solution with the theoretically minimum number of M/D, which often leads to a poor quality solution in the $\sum c_p x_p$ part (An extreme case would be to serve each node j with flow(s) over the path made by $s \rightarrow j \rightarrow t$, such that overall no M/D exists at all in the graph.). To prevent the objective value from being excessively compromised by AFUB, another upper bound is used to make the optimal solution consistent with Definition 1. Let $\hat{z}_{\text{int}} = \sum c_p \hat{x}_p$ be the incumbent best integral (but not necessarily operable) objective value in terms of flow variables, and let z_a^n be the objective value as a result of branching underutilized a at BB tree node n . Besides the standard cut off by bound criterion based on the global upper bound \hat{z}_{opr} from the incumbent best operable solution, if an extra quality control condition $z_a^n > \hat{z}_{\text{int}} + \epsilon_Q$ is met, tree node n will also be cut off, that is, the upper bound to cut off a node by bound is set as $\min\{\hat{z}_{\text{opr}}, \hat{z}_{\text{int}} + \epsilon_Q\}$. Note that this additional upper bound only applies at the AFUB stage. This extra level of quality control is important for ensuring the balance between $z(x) = \sum c_p x_p$, the original objective and $\zeta(x) = \#(x)$, the number of M/D operations, as controlled by ϵ_Q in Definition 1. Unlike a standard BB, AFUB cannot guarantee the exactness in finding an optimal solution as a balance between $z(x)$ and $\zeta(x)$ with respect to ϵ_Q . However, the BB tree’s structure will provide wide search space, in contrast to a heuristic method that tries to fixed up underutilized arcs in a uni-directional or local search manner.

Similar to selecting an appropriate fractional variable to branch on in standard BB, for AFUB, it is important to decide which underutilized arc to branch on among a large number of candidates. The following lists two verification methods based on the Theorems 1 to 3, model (MS_e), and a third option to allow to branch on “non-guaranteed” arcs:

- i. *PG*: Find guaranteed arcs by complete potential graphs as in Theorems 1 to 3.
- ii. *MS*: Find guaranteed arcs by the local model (MS_e).
- iii. *NG*: Branch on randomly selected underutilized arcs without guaranteed feasibility or improvement.

When designing a branching variable selection strategy for the AFUB, it is possible to choose a combination of the methods among “*PG*,” “*MS*,” and “*NG*.” When more than one of them are in use, the following priority order is enforced: $PG > MS > NG$, that is, first the potential graph will be used, and if it fails, location sub-model will be called, and if it fails too, the arc will be used without any guaranteeing. In fact, this priority order gradually widens the scope to be concerned from a small FCG (*PG*), to a further widened scope, location (*MS*), and finally to the entire DAG (*NG*). The last criterion may be occasionally needed in conjunction with (*PG*) and/or (*MS*), since a locally unproven underutilized arc may still make a global improvement over the DAG.

Algorithm 2 summarizes the major steps of AFUB.

Algorithm 2. Arc full utilization branching

```

if  $x_a \in \mathbb{Z}_+$ ,  $\forall a \in \mathcal{A}$  then
  Switch to AFUB branching
  Set upper bound to:  $\min\{\hat{z}_{\text{opr}}, \hat{z}_{\text{int}} + \epsilon_Q\}$ 
  for all  $a \in \mathcal{A} : x_a > 0$ 
    if  $\pi_a > 0$  AND  $fail\_times(a) < F_{\max}$ 
      Choose an arc proving strategy  $s$  based on  $\{PG, MS, NG\}$ 
      if Strategy  $s$  approves fully utilizing  $a$ 
        Form two branches  $B_1(a)$  and  $B_2(a)$ :

```

- At $B_1(a)$, fully utilizing a
- At $B_2(a)$, make a back-up node and set $fail_times(a) + = 1$

else
 Reset upper bound to: \hat{z}_{opr}
 Switch back to flow branching
end if

Branching on a (strongly) balanced underutilized arc (i, j) does not require adding an explicit constraint such as $x_{ij} = \mu_{ij}$. It can be realized in a more efficient way by removing all arcs in $\delta_+(i) \cup \delta_-(j) \setminus \{(i, j)\}$, leaving (i, j) the only choice for sending flows from i and for receiving flows to (j) . Similarly, branching on an unbalanced arc (i, j) , $w_i > w_j = \mu_{ij}$ can be realized by removing all arcs in $\delta_-(j) \setminus \{(i, j)\}$.

5.4 | A warm-start based solution approach for model (M_{xy})

A warm-start based approach is proposed for solving the integer FCMF model (M_{xy}) , which is known to be difficult to solve for medium to large scaled instances due to its fixed-charge/concavity nature [13], and most literature uses heuristics instead, for example, [6,7,15]. In our problem, since the fixed-charge term in the objective has its unique form solely used for minimizing the number of M/D operations, and we have designed the AFUB to heuristically reduce the M/D number in the “partial” model (M_x) , this local knowledge of our problem is thus used for a two-step warm-start framework introduced here to speed up the process for exactly solving the “full” model (M_{xy}) to satisfactory optimality.

The principle of this approach is to first solve model (M_x) with AFUB heuristics, obtain its (sub-)optimal solution x^\dagger and (sub-)optimal objective z^\dagger . Next, solve model (M_{xy}) with x^\dagger as an initial feasible solution to get the final optimal solution (x^*, y^*) of model (M_{xy}) . A global upper bound $z^\dagger + \zeta(x^\dagger)$ on the optimal objective value of (M_{xy}) will also be derived from x^\dagger to be used at the beginning for the BB process in (M_{xy}) . Note that the quality control tolerance ϵ_Q should be appropriately set in the AFUB to be consistent with W , the weights of M/D operations in (M_{xy}) . Figure 13 illustrates the basic ideas of this warm-start scheme.

In Section 6, experiment results of this warm-starting approach will be discussed following those for the application of AFUB. The contribution of the AFUB method is in particular demonstrated by this warm-start process. As shown by the computational experiments, instances without warm-start were often unable to be solved by the full model (M_{xy}) . On the other hand, with the aid of an initial solution provided by (M_x) with AFUB, these instances were successfully solved by (M_{xy}) with high-quality solutions.

6 | COMPUTATIONAL EXPERIMENTS ON TUSP

We use the branch-and-price approach in [20] to solve an enhanced version of model (M_x) . Note that (M_x) is a much simplified version used to ease the interpretation of the TUSP. In real-world instances, the requirements are more complicated and instead of using (M_x) , the model proposed in [20] (denoted by (M'_x)) is used in our experiments, where Constraints 3 and 4 are replaced by the train convex hull constraints to strengthen LP relaxation bounds and realize the combination-specific coupling upper bounds [20,23]. Moreover, to follow the requirements on train-family compatibility and locations banned for C/D operations, train-family branching and banned location branching are also used besides the conventional branching on variables. Apart from the above three kinds of branching, AFUB is also included. Similarly, the actual model used for the integer FCMF formulation

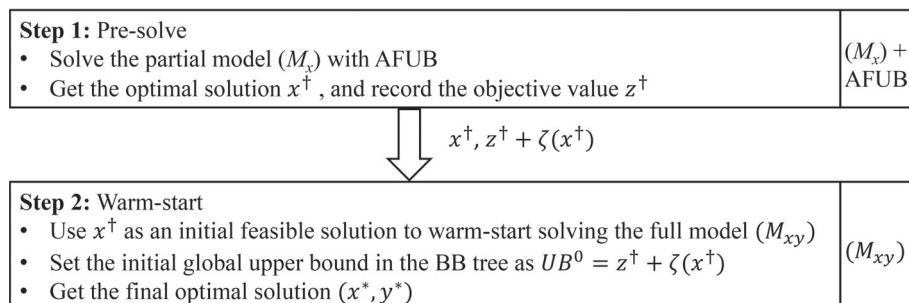


FIGURE 13 Two-step warm-start scheme for speeding up the solution process of the full model (M_{xy})

TABLE 2 Details of the instances

Instance	$ \mathcal{N} $	$ \mathcal{A}^K $	$ K $	Unit types and fleet sizes	Region
GE-360	139	1020	1	Class 360: 20	East Anglia
Northern-156	387	6476	1	Class 156: 56	Northern
TPE	358	8106	3	Class 185: 44 Class 170: 7 Class 350: 8	TPE
TPE2	260	4570	2	Class 185: – Class 170: –	TPE

(M_{xy}) has Constraints 9 and 10 replaced by train convex hull constraints, giving an enhanced version (M'_{xy}) . The solution approach for (M'_{xy}) is inherited from the branch-and-price method for (M'_x) , except that the newly added fixed-charge variables y are also branched after all the flow variables x are integral.

We will compare the experiment results with our previous solver (i.e., the one without AFUB in [20]) and the schedules produced by practitioners. As far as we know, there is no other equivalent existing research for making comparisons.

The experiments were conducted on a 64 bit Xpress-MP suite (latest version 8.0) on a Dell workstation with 8G RAM and an Intel Xeon E31225 CPU. The solvers used are two upgraded versions of the branch-and-price solver used in [20], one for solving (M'_x) and the other for (M'_{xy}) . In both, only the simplex solver from Xpress-MP were applied to solve LP relaxations during the column generation process without employing the default integer programming solver from Xpress-MP itself.

6.1 | Applying AFUB on model (M'_x)

To test how effective the proposed AFUB scheme is for solving model (M'_x) , we conducted experiments based on four real-world instances of the rail network of East Anglia, TransPennine Express (TPE) and Northern. The details of the four instances are summarized in Table 2. The fleet sizes refer to the number of units used in the manual schedules if available. GE-360 and Northern-156 only contain a single type of unit, and TPE and TPE2 contain multiple unit types. Except TPE2, all the other instances are from previous timetables where the corresponding manual unit schedules are available for comparison. TPE2 on the other hand, is derived from a new timetable without any manual schedule produced. Therefore the fleet size information is not available for TPE2.

The parameter settings for model (M'_x) are as follows. The minimum and maximum connection time durations were set to be 4 and 1000 minutes, respectively, plus some manually determined special turnaround rules at certain time and locations. To make fair and clear comparisons with manual solutions focusing on the number of C/D operations, only the fleet size was minimized and therefore $\varepsilon_Q = 0$ (most strict) was set. Also for making fair comparisons, we strictly use the manually scheduled train capacities as the input demand. Note that this restricts the solver from getting better solution qualities such as a fewer number of units used. Only the existing empty-running movements in the manual schedules were allowed. The maximum number of column iterations was set to be 2000 and an iteration will be prematurely stopped if the relative gap between the upper and lower bound is less than 0.1%, and the lower bound will be used as the tree node's LP relaxation value. A maximum total number of 1000 BB tree nodes and 12 hours of running time were set. The branch-and-price was set to terminate as soon as the relative gap between the BB tree's global lower bound and global upper bound is less 0.1%. The maximum failure count for fully utilizing an arc was set as $F_{\max} = 30$.

Different strategies were considered (see Section 5.3) when implementing AFUB. For instance, one can only use potential graph (PG), or use both PG and the local model (MS). When non-guaranteed arcs are allowed to be branched (NG), they are selected randomly. Finally we also would like to compare the results between using PG and/or MS and not using them at all by only branching on arbitrary underutilized arcs. Since unnecessary C/D can rarely happen in manually constructed schedules, we use the C/D counts in manual schedules as the benchmark for accessing the quality given by the AFUB in the experiments. Table 3 gives the results on the four instances with different strategies indicated by Yes (Y) or No (N) regarding PG , MS , and NG . When non-guaranteed arcs are allowed to be branched, because of the randomness in selecting underutilized arcs, the average values over 100 runs are reported. The column “rule%” gives the percentage of other branching rules (i.e., train-family, banned location and fractional-to-integral) and AFUB in the BB process. “BB#” gives the number of nodes in the BB tree. “cd#” gives the number of C/D operations in the solution, followed by its SD if the average value is taken using NG .

The GE-360 instance is a small instance with 137 trips and only a single unit type Class 360. From Table 3, it can be seen that the number of C/D operations is significantly improved from 50 to 28 even if only the potential graph (PG) is used. If PG and MS are both used, this number can be improved to 24, the same as in the manual solution. Moreover, allowing branching

TABLE 3 Results of AFUB on (M'_x)

Instance	Test	AFUB	PG	MS	NG	Unit#	Rule% ^a	BB#	Time (s)	cd# (sd)
GE-360	1	N	-	-	-	20	100/0	13	8	50
	2	Y	Y	N	N	20	70/30	24	11	28
	3	Y	Y	Y	N	20	64/36	25	13	24
	4	Y	Y	Y	Y	20	41/59	147.28	29.84	23 (0)
	5	Y	N	N	Y	20	48/52	430.63	643.04	36.61 (7.37)
	Manual	-	-	-	-	20	-	-	-	24
Northern-156	1	N	-	-	-	56	100/0	330	681	63
	2	Y	Y	N	N	56	94/6	364	800	35
	3	Y	Y	Y	N	56	78/22	424	1096	35
	4	Y	Y	Y	Y	56	68/32	514	1282.05	35 (0)
	5	Y	N	N	Y	56	77/23	1120.09	2665.93	39.10 (2.31)
	Manual	-	-	-	-	56	-	-	-	23
TPE	1	N	-	-	-	59	100/0	66	216	47
	2	Y	Y	N	N	59	94/6	71	242	36
	3	Y	Y	Y	N	59	94/6	71	236	36
	4	Y	Y	Y	Y	59	74/26	121	305.55	36 (0)
	5	Y	N	N	Y	59	56/44	220.22	1243	39.36 (3.329)
	Manual	-	-	-	-	59	-	-	-	34
TPE2	1	N	-	-	-	51	100/0	79	82	31
	2	Y	Y	N	N	51	93/7	85	90	18
	3	Y	Y	Y	N	51	75/25	145	121	18
	4	Y	Y	Y	Y	51	75/25	145	105.15	18 (0)
	5	Y	N	N	Y	51	71/29	192.52	136.66	17.04 (2.019)

^a Others/AFUB.

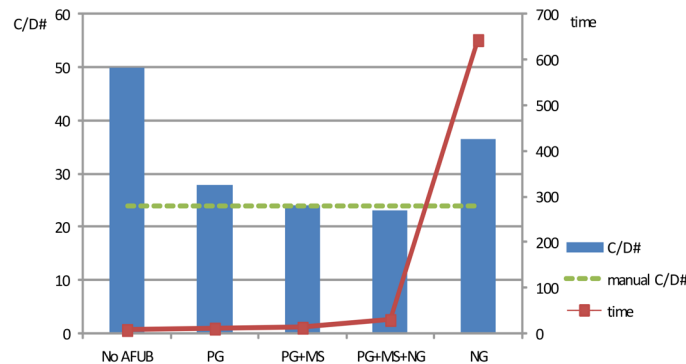


FIGURE 14 GE-360: C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

on non-guaranteed arcs can further reduce the C/D number by one. Note that if both *PG* and *MS* are used, often there would be no further space of improvement for the additional use of *NG*, yielding relatively stable behaviors with a zero deviation. This implies the usefulness of the verification methods *PG* and *MS* such that they are able to cover most underutilized arcs that can guarantee an improvement. Similar phenomena also happened in the experiments for the other instances. If *PG* and *MS* are totally abandoned and AFUB only selects non-guaranteed underutilized arcs randomly, the quality on the number of C/D will decline significantly, not to mention the time consumed and the number of BB nodes will be larger by several magnitudes, as shown in Test 5. Figure 14 further illustrates the C/D number and computational time from different strategies, where the advantage of applying AFUB can be observed.

The Northern-156 instance has 385 trips and a single unit type Class 156. Figure 15 illustrates the C/D number and computational time from different strategies. Without any AFUB, the number of C/D operations is large (65), and by applying *PG* or *PG + MS*, this number is reduced to 35. The earlier Figure 3 is taken from the solution in Test 1 without any AFUB applied. As a comparison, Figure 16 gives the improved results at Leeds Station given by Test 2 (*PG* only). One can see that all the unnecessary C/D operations in Figure 3 have been removed. Unfortunately, the heuristic AFUB still cannot obtain a very close

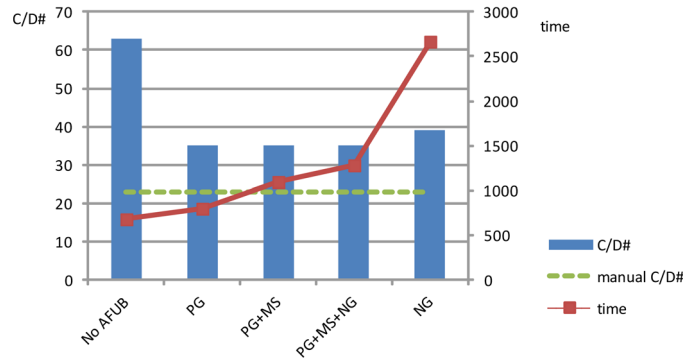


FIGURE 15 Northern-156: C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

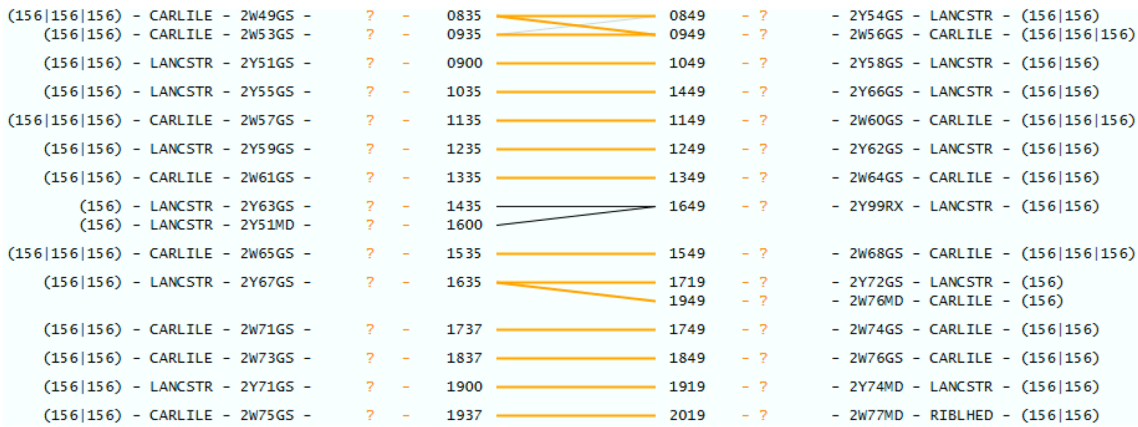


FIGURE 16 The result at Leeds station given by Test 2 in Northern-156 instance [Color figure can be viewed at wileyonlinelibrary.com]

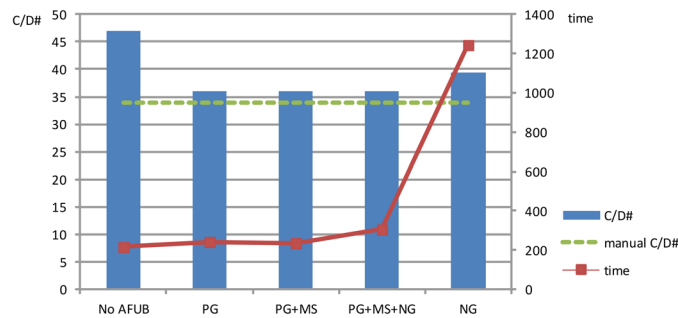


FIGURE 17 TPE: C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

cd# to the manual schedule’s 23. After comparing the schedule in Test 2 and the manual solution, it is found that if the capacity provisions have to be kept as in Test 2, there will be almost no scope to further reduce the C/D#. However, the manual solution achieves a better one due to different capacity provisions from the ones in Test 2, where a small number of unbalanced arcs in Test 2 are replaced by balanced arcs in the manual solution. From Equation (14), flows over unbalanced arcs are the other reason for increasing C/D operations besides underutilized arcs. AFUB, which only focuses on underutilized arcs, is thus difficult to make improvements in the case of unbalanced arcs. However it remains an open question whether a similar scheme as the AFUB is still workable for unbalanced arcs. A more detailed discussion over this issue is given in Section 7.

In instances TPE and TPE2, multiple unit types are involved. Similar patterns in different strategies as in instances GE-360 and Northern-156 can still be found. In instance TPE, the number of C/D can be reduced from 47 to 36, very close to 34, as in the manual schedule. In instance TPE2, the number of C/D can be reduced from 31 to 18 by prioritizing verification rules and to 17 on average if using randomly selected underutilized arcs. Since this is a new timetable, no manual schedules are available for comparison. Figures 17 and 18 further illustrates the above results.

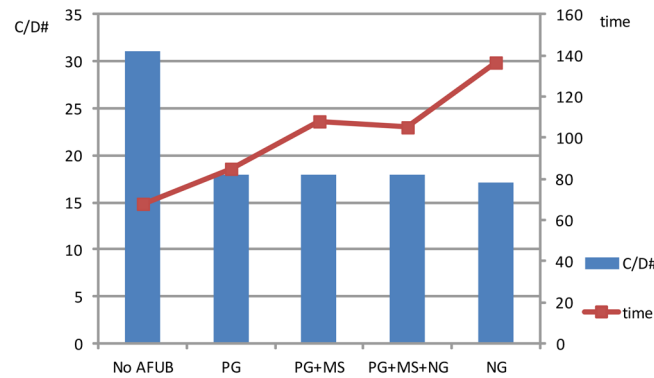


FIGURE 18 TPE2: C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

TABLE 4 Results on instances solved by model (M'_{xy})

Instance	WS	Oper-obj	Fleet size	BB#	Time (s)	r% (other/AFUB/block)	C/D#	gap%	WS%
GE	Y	22.91	20	235	180	7/10/83	17	0.175	92.9
GE	N	22.95	20	1000	758	2/3/95	21	0.923	–
Manual	–	–	20	–	–	–	24	–	–
North	Y	64.01	56	1000	32 156	29/2/69	31	0.460	75.8
North	N	Failed	–	1000	39 786	–	–	–	–
Manual	–	–	56	–	–	–	23	–	–
TPE	Y	66.4	59	1000	25 004	5/1/94	28	0.403	83.41
TPE	N	Failed	–	1000	41 861	–	–	–	–
Manual	–	–	59	–	–	–	34	–	–
TPE2	Y	56.28	51	1000	9119	7/1/92	8	0.213	77.7
TPE2	N	57.29	51	1000	10 670	7/3/90	13	2.012	–
(M'_x) best	–	–	–	–	–	–	18	–	–

6.2 | Warm-start for model (M'_{xy})

Experiments were conducted based on the same instances of GE-360, Northern-156, TPE and TPE2. The branching strategy (e.g., variable selection and node selection) for model (M'_{xy}) were all inherited from the branch-and-price solver for model (M'_x) including the AFUB with $GP + MS + NG(F_{max} = 3)$, and additionally block-arc variables y are also branched when all the other branching targets are satisfied. Because of the difficulty in solving model (M'_{xy}), we were only interested in finding near-optimal operable solutions in reasonable time. Thus, besides the same stopping criteria of a maximum of 1000 BB tree nodes and 12 hours running time, only a maximum of 800 column generation iterations per tree node were set and the branch-and-price process would also terminate as soon as two operable solutions were found. The weight on C/D operation costs was set as $W = 0.01$. All the other settings were kept the same as in Section 6.1. x^\dagger , the optimal solution from model (M'_x) used for warm-starting (M'_{xy}), was derived by the AFUB strategy of $GP + MS$.

Table 4 summarizes the results from running instances GE-360 (GE), Northern-156 (North), TPE and TPE2 within each a comparison between two runs with and without warm-start (WS) is made. Information on the manual solutions or best solutions from (M_x) (if no manual solution is available) is also provided. In “r%,” the percentage of variable branching on block-arc variables is reported as “block.” The percentage of solution relative gaps (gap%) and the percentage of the warm-start arcs (i.e., the ones in x^\dagger) found in the final solution (WS%) are also reported.

As for GE-360, due to the exactness of model (M_{xy}), solutions with fewer numbers of C/D operations and the same fleet size compared with the manual schedule can be obtained, and the one warm-started by x^\dagger from using AFUB in (M'_x) yields an even fewer number of C/D and a smaller gap percentage while taking only 23.4% of computational time for the one without warm-start. A high percentage of 92.9% of the arcs in x^\dagger were found in the final solution given by (M'_{xy}), which made an important contribution to the success of warm-start for this run. Figure 19 further illustrates the C/D number and computational time from the two runs.

The instance Northern-156 is a difficult one. Compared with 35, the best C/D number that can be yielded by model (M'_x), model (M'_{xy}) can reduce it to 31 with warm-start while retaining the same fleet size, which is still not close enough to the manual

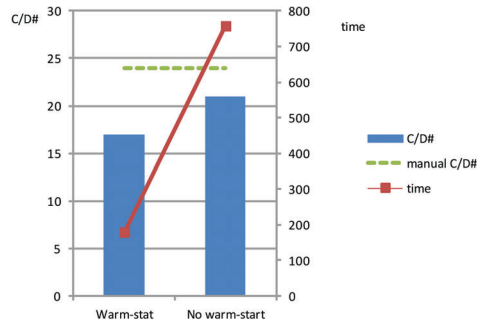


FIGURE 19 GE-360, model (M'_{xy}): C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

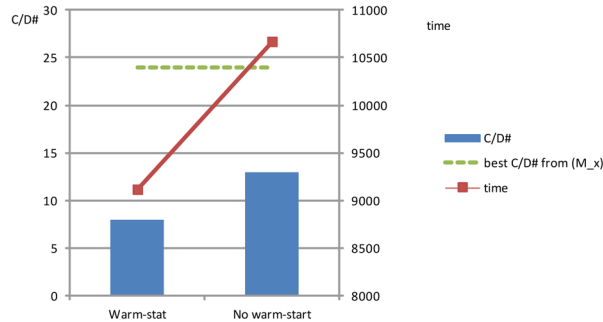


FIGURE 20 TPE2, model (M'_{xy}): C/D number and computational time from different strategies [Color figure can be viewed at wileyonlinelibrary.com]

C/D number 23 because of the difficulty of this instance. 75.8% of the arcs in x^\dagger were found in the final solution given by (M'_{xy}). On the other hand, the run without warm-start failed within 1000 BB nodes without getting any feasible solution.

As for the results from running instance TPE with three types of units, compared with 34, the manual C/D number and 36, the best C/D number given by model (M'_x), model (M'_{xy}) can further reduce the number to 28 with warm-start while retaining the same fleet size. 83.41% of the arcs in x^\dagger were found in the final solution given by (M'_{xy}). The other run without warm-start failed within 1000 BB nodes without any feasible solution.

Finally as for the results from running instance TPE2, compared with 18, the best C/D number found by model (M_x), model (M_{xy}) can reduce this number to 8 with warm-start and to 13 without warm-start while retaining the same fleet size. The solution without warm-start gives a gap that is almost 10 times larger than the solution with warm-start. 77.7% of the arcs in x^\dagger were found in the final solution given by (M'_{xy}). Figure 20 further illustrates the C/D number and computational time from different strategies.

7 | CONCLUSIONS AND FUTURE RESEARCH

Unnecessary flow demerging and remerging appears in some network flow problems where additional controls are needed and are often achieved by extra fixed-charge binary variables, which may affect the computational efficiency. In order to better deal with this problem, we have upgraded and significantly enhanced a heuristic branching scheme named AFUB proposed in [21].

The AFUB scheme is based on the observation that the number of M/D operations is closely related to the number of underutilized arcs, and by heuristically detecting and removing such arcs, there is a potential to reduce UDR. It is important to verify whether full utilization on an arc can guarantee an improvement, since applying AFUB to randomly selected underutilized arcs is ineffective and time consuming. Two methods for verifying the validity of fully utilizing a chosen arc are proposed. The first one (*PG*) is based on the idea of potential graphs and is suitable for an FCG where either the focused arc is balanced and commodities are interchangeable, or the focused arc is strongly balanced and the maximum flow amount for the nodes is up to 4. The use of potential graphs have been also extended to unbalanced cases if commodities are interchangeable. The second one (*MS*) is based on an integer programming model that tries to find a holistic improvement over the trips of a single location. Note that (*MS*) is not limited by the arc balance or commodity interchangeability conditions as for (*PG*).

An example of UDR in practice is unnecessary C/D operations from the TUSP. Experiments based on real-world instances from several UK train operating companies were conducted using model (M_x) plus AFUB. The C/D counts from manual schedules were used as benchmarks for evaluating the quality of AFUB. We found that *PG* and *MS* are effective in removing unnecessary C/D compared with the runs without any AFUB or the runs where AFUB was only arbitrarily applied to

non-guaranteed underutilized arcs (NG). For some instances, AFUB with PG and MS can find solutions with the same or very close C/D count as in the manual solution (with the same number of units used). Only one instance failed in getting a C/D count close to the manual solutions, although there was still a great improvement compared with the run without AFUB. Experiments for warm-starting the integer FCMF model (M'_{xy}) by an optimal solution from the model (M'_x) plus AFUB were also carried out, and the warm-start scheme is proved to be vitally important in deriving high-quality solutions in model (M_{xy}) in much shorter time. In conclusion, AFUB can give a step-change in efficiently obtaining train unit schedules with most unnecessary C/D removed either by being solely used in model (M_x) or to assist the solution process in model (M_{xy}) by warm-starting.

As for future research, the focus will be on unbalanced arcs in relation to the number of C/D operations. As shown by Equation (14), the total number of C/D operations can be decomposed into one part due to underutilized arcs and the other part due to unbalanced arcs. Changing unbalanced arcs into balanced ones might also reduce the C/D count, while our research has not considered this aspect yet. As shown in the experiments on Northern-156, there is still scope for further improvement by changing unbalanced arcs into balanced ones (without violating the mandatory constraints such as passenger demands or coupling upper bounds). Nonetheless, the proportion of unbalanced arcs over all arcs is often higher by magnitudes than the proportion of underutilized arcs, and most of them are solved to correct flows to provide satisfactory capacities. This makes the design of a corresponding algorithm more difficult than AFUB. There seems to be no obvious way, such as the potential graph or model (MS_e) as in the case of underutilized arcs, to verify the validity of balancing an unbalanced arc before actually enforcing it in the next LP relaxation. Moreover, forcing an unbalanced arc into a balanced one will have a much greater effect over the entire network. It is also very likely to be conflicting with other operating costs such as mileage. Therefore, it remains an open question whether it is needed to design a heuristic approach for changing some unbalanced arcs into balanced ones.

Another important direction is to further investigate the warm-start strategy as presented in Section 5.4 in conjunction with other techniques to speed up the solution process for model (M_{xy}). The experiments in Section 6.2 show that the effectiveness of AFUB, but currently only small to medium instances were tested. The future research in this aspect will look for more effective branching strategies combining both the x and y variables, valid inequalities and a hybridized heuristic framework SLIM [18] designed for large scale TUSP instances.

ACKNOWLEDGMENTS

This research is supported by an Engineering and Physical Sciences Research Council (EPSRC) project EP/M007243/1. We would like to also thank First TransPennine Express, Abellio (UK), and Tracsis PLC for their kind and helpful collaboration.

DATA AVAILABILITY STATEMENT

First TransPennine Express, Abellio (UK), and Tracsis PLC have provided relevant data for the research, part of which is commercially sensitive. Where possible, the data that can be made publicly available is deposited in <http://archive.researchdata.leeds.ac.uk/>

ORCID

Zhiyuan Lin  <https://orcid.org/0000-0001-7361-5469>

REFERENCES

- [1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows: Theory, Algorithms and Applications*, Prentice Hall, Englewoods Cliffs, NJ, 1993.
- [2] F. Alvelos, Branch-and-price and multicommodity flows, Ph.D. thesis, Escola de Engenharia, Universidade do Minho, Portugal, 2005.
- [3] C. Barnhart, C.A. Hane, and P.H. Vance, *Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems*, *Oper. Res.* **48** (2000), 318–326.
- [4] M. Bellemore, G. Bennington, and S. Lubore, *A multivehicle tanker scheduling problem*, *Transp. Sci.* **5** (1971), 36–47.
- [5] V. Cacchiani, A. Caprara, and P. Toth, *Solving a real-world train-unit assignment problem*, *Math. Program. Ser. B* **124** (2010), 207–231.
- [6] T.G. Crainic, A. Frangioni, and B. Gendron, *Bundle-based relaxation methods for multicommodity capacitated fixed charge network design*, *Discrete Appl. Math.* **112** (2001), 73–99.
- [7] T.G. Crainic, M. Gendreau, and J.M. Farvolden, *A simplex-based tabu search method for capacitated network design*, *INFORMS J. Comput.* **12** (2000), 223–236.
- [8] G. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J. USA, 1998.
- [9] B.L. Dietrich, *Monge sequences, antimatroids, and the transportation problem with forbidden arcs*, *Linear Algebra Appl.* **139** (1990), 133–145.
- [10] L.R. Ford and D.R. Fulkerson, *A suggested computation for maximal multi-commodity network flows*, *Manag. Sci.* **5** (1958), 97–101.
- [11] R. Freling, R.M. Lentink, L.G. Kroon, and D. Huisman, *Shunting of passenger train units in a railway station*, *Transp. Sci.* **39** (2005), 261–272.
- [12] B. Gendron, T.G. Crainic, and A. Frangioni, *Multicommodity Capacitated Network Design*, Springer US, Boston, MA, 1999, pp. 1–19.
- [13] G.M. Guisewite and P.M. Pardalos, *Minimum concave-cost network flow problems: Applications, complexity, and algorithms*, *Ann. Oper. Res.* **25** (1990), 75–99.
- [14] M. Hewitt, G. Nemhauser, and M.W. Savelsbergh, *Branch-and-price guided search for integer programs with an application to the multicommodity fixed charge network flow problem*, *INFORMS J. Comput.* **25** (2013), 302–316.

[15] M. Hewitt, G.L. Nemhauser, and M.W.P. Savelsbergh, *Combining exact and heuristic approaches for the capacitated fixed-charge network flow problem*, *INFORMS J. Comput.* **22** (2010), 314–325.

[16] D.S. Hochbaum and A. Segev, *Analysis of a flow problem with fixed charges*, *Networks* **19** (1989), 291–312.

[17] L.G. Kroon, R.M. Lentink, and A. Schrijver, *Shunting of passenger train units: An integrated approach*, *Transp. Sci.* **42** (2008), 436–449.

[18] R.S.K. Kwan, Z. Lin, P.J. Copado-Mendez, and L. Lei, *Multi-commodity flow and station logistics resolution for train unit scheduling*, MISTA 2017, Kuala Lumpur, Malaysia, 2017.

[19] M.W. Lewis, *On the use of guided design search for discovering significant decision variables in the fixed-charge capacitated multicommodity network design problem*, *Networks* **53** (2009), 6–18.

[20] Z. Lin and R.S. Kwan, *A branch-and-price approach for solving the train unit scheduling problem*, *Transp. Res. Part B* **94** (2016), 97–120.

[21] Z. Lin and R.S. Kwan, *Redundant coupling/decoupling in train unit scheduling optimization*, *Electron. Notes Discrete Math.* **64** (2018), 45–54.

[22] Z. Lin and R.S.K. Kwan, *A two-phase approach for real-world train unit scheduling*, *Public Transp.* **6** (2014), 35–65.

[23] Z. Lin and R.S.K. Kwan, *Local convex hulls for a special class of integer multicommodity flow problems*, *Comput. Optim. Appl.* **64** (2016), 881–919.

[24] R.M. Lusby, J.T. Haahr, J. Larsen, and D. Pisinger, *A branch-and-price algorithm for railway rolling stock rescheduling*, *Transp. Res. Part B* **99** (2017), 228–250.

[25] P. Robers and L. Cooper, *A study of the fixed charge transportation problem*, *Comput. Math. Appl.* **2** (1976), 125–135.

[26] S. Schrenk, G. Finke, and V.-D. Cung, *Two classical transportation problems revisited: Pure constant fixed charges and the paradox*, *Math. Comput. Model.* **54** (2011), 2306–2315.

[27] R. Shamir, *A fast algorithm for constructing Monge sequences in transportation problems with forbidden arcs*, *Discrete Math.* **114** (1993), 435–444.

How to cite this article: Lin Z, Kwan RSK. Avoiding unnecessary demerging and remerging of multi-commodity integer flows. *Networks*. 2020;76:206–231. <https://doi.org/10.1002/net.21969>

APPENDIX

We explain how the node flow values w_j are replaced by the endpoint values μ_a, M_a from the second equation to the third and fourth equations in (14). For every incoming and outgoing arcs associated with node $j \in N$, that is, $a \in \delta_{\pm}(j)$, the arc is either a source/sink arc ($a \in A_0 : w_j = \mu_a$), or a balanced connection arc ($a \in A : w_j = \mu_a = M_a$), or an unbalanced connection arc ($a \in A : w_j = \mu_a \neq M_a$ or $w_j = M_a \neq \mu_a$). Therefore, we can rewrite $\sum_{a \in \delta_{\pm}(j)} \frac{x_a}{w_j}$ to,

$$\begin{aligned} \sum_{a \in \delta_{\pm}(j)} \frac{x_a}{w_j} &= \sum_{a \in \delta_{\pm}(j) \cap A_0} \frac{x_a}{\mu_a} + \sum_{a \in \delta_{\pm}(j) \cap A : w_j = \mu_a \neq M_a} \frac{x_a}{\mu_a} \\ &+ \sum_{a \in \delta_{\pm}(j) \cap A : w_j = M_a \neq \mu_a} \frac{x_a}{M_a} + \sum_{a \in \delta_{\pm}(j) \cap A : w_j = \mu_a = M_a} \frac{x_a}{\mu_a}, \quad \forall j \in N. \end{aligned} \tag{A1}$$

Figure A1 gives an example with five intermediate nodes i, j, l, m, n and how the summation term is constructed per Equation (A1).

When summing up the terms as in (A1) for all the intermediate nodes $j \in N$, note that each source/sink arc will only be counted once, while for each connection arc, it will be counted twice. We further have,

$$\sum_{j \in N} \sum_{a \in \delta_{\pm}(j)} \frac{x_a}{w_j} = \sum_{a \in A_0} \frac{x_a}{\mu_a} + \sum_{a \in A : \mu_a \neq M_a} \left(\frac{1}{\mu_a} + \frac{1}{M_a} \right) x_a + \sum_{a \in A : \mu_a = M_a} \frac{2x_a}{\mu_a} \tag{A2}$$

For instance, after summing up the five summation terms as given in Figure A1, we have

$$\begin{aligned} \sum_{h \in \{i, j, k, l, m\}} \sum_{a \in \delta_{\pm}(h)} \frac{x_a}{w_h} &= \frac{x_{si}}{\mu_{si}} + \frac{x_{sj}}{\mu_{sj}} + \frac{x_{kt}}{\mu_{kt}} + \frac{x_{lt}}{\mu_{lt}} + \frac{x_{mt}}{\mu_{mt}} \\ &+ \left(\frac{1}{\mu_{jk}} + \frac{1}{M_{ij}} \right) x_{jk} + \left(\frac{1}{\mu_{jl}} + \frac{1}{M_{jl}} \right) x_{jl} + \left(\frac{1}{\mu_{jm}} + \frac{1}{M_{jm}} \right) x_{jm} \\ &+ \frac{2}{\mu_{ik}} x_{ik} + \frac{2}{\mu_{il}} x_{il}. \end{aligned} \tag{A3}$$

By rearranging the terms in Equation (A2) and continue the calculation process, it will finally lead to the result given in (14).

$$\sum_{a \in \delta_{\pm}(i)} \frac{x_a}{w_i} = \frac{x_{si}}{\mu_{sj}} + \frac{x_{ik}}{\mu_{ik}} + \frac{x_{il}}{\mu_{il}}$$

$$\sum_{a \in \delta_{\pm}(j)} \frac{x_a}{w_j} = \frac{x_{sj}}{\mu_{sj}} + \frac{x_{jk}}{M_{ik}} + \frac{x_{jl}}{M_{jl}} + \frac{x_{jm}}{M_{jm}}$$

$$\sum_{a \in \delta_{\pm}(k)} \frac{x_a}{w_k} = \frac{x_{kt}}{\mu_{kt}} + \frac{x_{ik}}{\mu_{ik}} + \frac{x_{jk}}{\mu_{jk}}$$

$$\sum_{a \in \delta_{\pm}(l)} \frac{x_a}{w_l} = \frac{x_{lt}}{\mu_{lt}} + \frac{x_{il}}{\mu_{il}} + \frac{x_{jl}}{\mu_{jl}}$$

$$\sum_{a \in \delta_{\pm}(m)} \frac{x_a}{w_m} = \frac{x_{mt}}{\mu_{mt}} + \frac{x_{jm}}{\mu_{jm}}$$

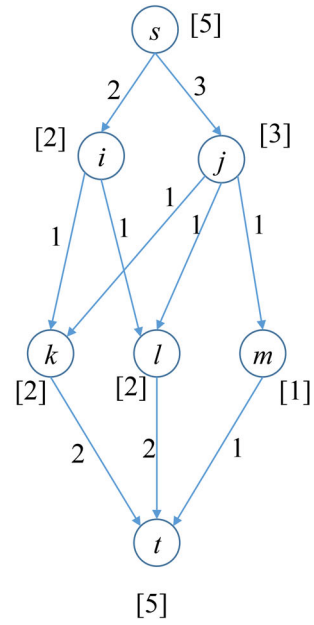


FIGURE A1 An illustrative example for equations in (14). The node flow amount is marked by $[w_j]$ for node j . The arc flow amount is marked along each arc [Color figure can be viewed at wileyonlinelibrary.com]