

Maximum concurrent flow with incomplete data

Pierre-Olivier Bauguion¹, Claudia D'Ambrosio², and Leo Liberti²

¹ IRT SystemX, 8 Avenue de la Vauve, 91120 Palaiseau, France
`pierre-olivier.bauguion@irt-systemx.fr`

² CNRS LIX, Ecole Polytechnique, 91128 Palaiseau, France
`{dambrosio,liberti}@lix.polytechnique.fr`

Abstract. The Maximum Concurrent Flow Problem (MCFP) is often used in the planning of transportation and communication networks. We discuss here the MCFP with incomplete data. We call this new problem the Incomplete Maximum Concurrent Flow Problem (IMCFP). The main objective of IMCFP is to complete the missing information assuming the known and unknown data form a MCFP and one of its optimal solutions. We propose a new solution technique to solve the IMCFP which is based on a linear programming formulation involving both primal and dual variables, which optimally decides values for the missing data so that they are compatible with a set of scenarios of different incomplete data sets. We prove the correctness of our formulation and benchmark it on many different instances.

Keywords: Maximum concurrent flow, multi-commodity flow problems, incomplete data, unknown data, uncertainty, inverse optimization, transportation systems.

1 Introduction

Network flows have been introduced long ago (see, e.g., [12], enhanced later by [11]) to tackle single commodity flow problems, such as the max-flow problem. Since then, these models have been generalized for multiple commodities [20] and grouped under the label of multi-commodity flow models. Nowadays, multi-commodity flow formulations are extensively used in many contexts for their ability to capture the movements of different types of commodities in various real-world activities such as people in transportation models, data in telecommunication networks, water flows... (see, e.g., [1]). These formulations are generally used to help make the best cost-effective solution for allocating resources; this leads to optimize a cost function.

In real-world applications, the available data are often uncertain or incomplete, and their actual values may only be revealed at a time when the overall decision strategy has already been chosen. This is often the case in transportation systems where the parameters are time-dependent and event-sensitive. Statistical inference and data mining represent convenient ways to deal with this uncertainty. One of the best known inference models in transportation systems is

the Four Step Model [18], which is an algorithm that iterates over time according to an equilibrium criterion. More recently, a lot of attention has been devoted to machine learning approaches, which generally performs better on large scale datasets. In this context, [21] proposes bayesian networks and [17] uses a deep learning approach to forecast flow in transportation systems.

However, optimization methods that deal with uncertainty actually do exist. To the best of our knowledge, [10] was the first to propose a stochastic approach to tackle incompleteness of input data. It assumes that uncertain data follow some given probability distribution, and that the objective of this approach is not to optimize a certain cost, but an expected cost instead. On the other hand, [22] proposed a complementary approach by optimizing a robust criterion such as the worst case or the maximum regret. This particular method received renewed attention from [6] and [7], while [5] applies this approach to multi-commodity flow problems by considering a polyhedral uncertainty set of demands. Later, [3] mixed the recourse variables introduced in [10] with the robust approach for a network flow and design problem.

Optimization methods can also be used to optimally fit experimental measurements. In [15], multi-commodity flow optimization is used to model a gas transportation network while retrieving missing data. The problem discussed in [15] consists in recomposing the flow on each arc, knowing only the global amount of incoming and outgoing flows for each node. The problem of finding a minimal adjustment of the cost function to ensure the optimality of a given solution generated a particular interest with [9] under the label of inverse optimization. For example [2, 23] apply this concept to multi-commodity flow problems (especially min cost flow problem). The survey [14] on this subject includes situations where the inverse problem seeks parameters other than objective function coefficients.

The Maximum Concurrent Flow Problem (MCFP) has been extensively studied over time [20, 8, 4], but in this paper we present a new approach for finding optimal maximum concurrent flows using incomplete data. Our method seeks optimal solutions and completes the partial input. This problem typically arises when we have insights about the global behavior of a system while data are partially unknown [15]. Symmetrically it can validate/invalidate a hypothetical behavior by comparing it with the observed data. This is particularly relevant in transportation when the routing strategy of passengers is known while data are incomplete. We call this problem Incomplete Maximum Concurrent Flow Problem (IMCFP).

The rest of this paper is organized as follows: Section 2 recalls the MCFP and presents the IMCFP. In Section 3 we propose a formulation for the IMCFP by integrating both primal and dual formulations of the MCFP, and prove its correctness based on the complementary slackness conditions of Linear Programming (LP), which we recall for convenience [16]. Then, in Section 4 we present preliminary experiments for these formulations and their practical interest. Finally, Section 5 concludes the paper.

2 The maximum concurrent flow

Consider the following (well-known) problem.

MCFP. Given a simple directed graph $G = (V, A)$, an arc capacity function $c : A \rightarrow \mathbb{R}_+$, a set K of triplets $k = (o, d, D) \in V^2 \times \mathbb{R}_+$, find: a scalar $\gamma \geq 0$ (called *threshold*) and a set of flows f^k on G for each $k \in K$ such that (i) for each arc $a \in A$, the *arc load* of f on a (i.e., the sum of the flows on a) does not exceed the arc capacity c_a ; (ii) each “ $o - d$ ” flow f^k has value γD for each $k = (o, d, D) \in K$; (iii) γ is maximum.

We recall that, for a given $k = (o, d, D) \in K$, a flow f having value D in a graph G from node o to node d is a non-negative arc function $f : A \rightarrow \mathbb{R}_+$ such that $\sum_{j \in N^-(d)} f_{jd} - \sum_{j \in N^+(d)} f_{dj} = D$ and the following flow balance equations hold (we omit k index for clarity):

$$\forall i \in V \setminus \{o, d\} \quad \sum_{j \in N^-(i)} f_{ji} = \sum_{j \in N^+(i)} f_{ij}. \quad (1)$$

We also recall that $N^-(i)$ is the set of nodes j such that $(j, i) \in A$ and $N^+(i)$ such that $(i, j) \in A$, for each $i \in V$.

The MCFP was introduced in [20]. It can be formulated as follows using LP:

$$\left. \begin{array}{l} \max_{\gamma \geq 0, f \geq 0} \quad \gamma \\ \forall k = (o, d, D) \in K \quad \sum_{j \in N^-(d)} f_{jd}^k = \gamma D + \sum_{j \in N^+(d)} f_{dj}^k \\ \forall k = (o, d, D) \in K, i \in V \setminus \{o, d\} \quad \sum_{j \in N^-(i)} f_{ji}^k = \sum_{j \in N^+(i)} f_{ij}^k \\ \forall a = (i, j) \in A \quad \sum_{k \in K} f_a^k \leq c_a. \end{array} \right\} \quad (2)$$

Its dual is (one can choose appropriate inequalities instead of equalities for the first and second sets of constraints of (2)):

$$\left. \begin{array}{l} \min_{p \geq 0, u \geq 0} \quad \sum_{a \in A} u_a c_a \\ \forall k = (o, d, D) \in K, \forall a = (i, j) \in A, \quad p_i^k + u_a \geq p_j^k \\ \forall k = (o, d, D) \in K, \quad p_o^k = 0 \\ \sum_{k=(o,d,D) \in K} p_d^k D \geq 1 \end{array} \right\} \quad (3)$$

where u_a , (for $a \in A$) are the dual variables associated to the capacity constraints (last set of constraints of (2)) and p_i^k (for $k \in K, i \in V$) are the dual variables associated to the flow conservation constraints (first and second sets of constraints of (2)). This implies that the MCFP is polynomial-time solvable (for example with an interior point algorithm). The MCFP is also strongly polynomial-time solvable [19], but it appears to be common knowledge that, for instance sizes of

current practical interest, it is empirically more efficient to use a good LP solver on Problem (2) rather than the algorithm in [19]. The MCFP is often used in real-life applications in order to design networks or evaluate the arcs with highest risk of becoming saturated [13]. The main applied interest in the MCFP is that, through the ratio variable γ , it ensures a fairness of arc capacity utilization over all flows.

Our motivation for studying this problem stems from transportation networks, be they road or rail-oriented. As a critical increase of the load can induce a decrease of the quality of services, an hypothesis consists of assuming that the passengers traffic tends naturally to balance itself to an equilibrium [18]. One can model this problem by minimizing the maximum capacity utilization, and the latter can be reformulated as a MCFP [20]. In our context, we have historical traffic data including a partial observation of the arc loads for a certain subset of arcs. For some networks, we are also given a subset $A' \subset A$ of arcs with known capacities.

In general, however, we do not know the arc capacities. The problem we are interested in is the MCFP with incomplete arc capacities. The MCFP in LP formulation (2) without the capacity constraints is clearly an unbounded LP. To avoid this situation, we employ a given set S of scenarios from our historical arc load database. Each scenario $s = (A^s, \ell^s, K^s) \in S$ consists of a subset $A^s \subset A$ of arcs, a partial arc load function $\ell^s : A^s \rightarrow \mathbb{R}_+$, and a set of commodities K^s . We require that: (i) missing capacities should be estimated so as to allow the maximum known arc loads over all scenarios, (ii) arc loads from computed flows should be as close as possible to the loads given in the scenarios, and (iii) each flow solution for a scenario should describe an optimal solution of the MCFP w.r.t. capacity and commodity values. We therefore define the following problem, which is new as far as we could ascertain.

IMCFP. Given a graph $G = (V, A)$, a subset $A' \subset A$, a partial arc capacity function $c : A' \rightarrow \mathbb{R}_+$, and a set S of scenarios $(A^s \subset A, \ell^s, K^s)$ where $\ell^s : A^s \rightarrow \mathbb{R}_+$ and K^s is a set of triplets $k = (o, d, D) \in V^2 \times \mathbb{R}_+$, find: a threshold function $\gamma : S \rightarrow \mathbb{R}_+$, a complementary arc capacity function $c : A \setminus A' \rightarrow \mathbb{R}_+$, and a set of flows f^{sk} (for $s \in S$ and for $k \in K^s$) such that (i) for each arc $a \in A$ and for each $s \in S$, the arc load of f on a is bounded above by c_a ; (ii) for each $s \in S$ and arc $a \in A^s$, the arc load of f on a is as close as possible to the arc load $\gamma_s \ell_a^s$; (iii) for each $s \in S$, the flows f^s and γ^s should be optimal with respect to an MCFP defined over the capacities c_a over all $a \in A$ and the commodities K^s .

Although the IMCFP is natively cast in a multi-objective fashion (see condition (ii)), in practice we minimize a max norm over all arcs and all scenarios. We remark that condition (iii) is only apparently recursive: we want to decide f, c at the same time and also require that every f^s should be optimal flows w.r.t. a putative MCFP instance defined over the values of the c variables and the K^s parameters. We shall see below that the IMCFP can be formulated by means of a Mixed-Integer Linear Programming formulation that combines both primal and dual variables.

3 The IMCFP

In this section we shall first introduce a Mixed-Integer Linear formulation for the IMCFP, and then prove its correctness.

3.1 Formulation

– *Sets:*

- V : set of nodes
- A : set of arcs
- S : set of scenarios
- K^s : set of commodities for scenario s
- A' : subset of arcs from which the capacity is known
- A^s : subset of arcs from which the load is known.

– *Parameters:*

- $k = (o, d, D)$ for $k \in K^s, s \in S$: commodity data (origin $o \in V$, destination $d \in V$, demand value $D \in \mathbb{R}_+$)
- $\ell^s : A^s \rightarrow \mathbb{R}_+$: load function over the arcs A^s for each scenario $s \in S$
- $c : A' \rightarrow \mathbb{R}_+$: capacity function over the arcs A'
- M^w : “Big M” parameter associated to the binary weights w
- M^f : “Big M” parameter associated to the binary flows x .

– *Decision variables:*

- $f_a^{sk} \geq 0$ for $a \in A, s \in S, k \in K^s$: flow variable of arc $a \in A$, for scenario $s \in S$ and demand $k \in K^s$
- $\gamma_s \geq 0$ for $s \in S$: threshold variable for scenario $s \in S$
- $p_i^{sk} \geq 0$ for $i \in V, s \in S, k \in K^s$: potential variable (dual variable from MCFP’s conservation constraint) for node i , scenario s , and demand k
- $u_a^s \geq 0$ for $a \in A, s \in S$: weight variable (dual variable from MCFP’s capacity constraint) for arc $a \in A$ and scenario $s \in S$
- $w_a^s \in \{0, 1\}$ for $a \in A, s \in S$: binary variable that allows for a corresponding weight u_a^s to be greater than 0 or not
- $x_a^{sk} \in \{0, 1\}$ for $a \in A, s \in S, k \in K^s$: binary variable that allows for a corresponding flow f_a^{sk} to be greater than 0 or not
- $\Delta \geq 0$: maximal difference between the load parameters with the computed ones
- $c_a \geq 0$ for $a \in A \setminus A'$: capacity variable of arc a .

– *Objective function:*

$$\min_{f, \gamma, p, u, w, x, c} \Delta. \quad (4)$$

– *Constraints:*

- flow conservation:

$$\forall s \in S, \forall k = (o, d, D) \in K^s, \forall i \in V \setminus \{o, d\} \quad \sum_{j \in N^-(i)} f_{ji}^{sk} = \sum_{j \in N^+(i)} f_{ij}^{sk} \quad (5)$$

- demand satisfaction:

$$\forall s \in S, \forall k = (o, d, D) \in K^s \quad \sum_{j \in N^-(d)} f_{jd}^{sk} - \sum_{j \in N^+(d)} f_{dj}^{sk} = \gamma_s D \quad (6)$$

- min-cost node access:

$$\forall s \in S, \forall k \in K^s, \forall (i, j) \in A \quad p_i^{sk} + u_{ij}^s - p_j^{sk} \geq 0 \quad (7)$$

- min-cost path condition:

$$\forall s \in S, \forall k \in K^s, \forall (i, j) \in A \quad p_i^{sk} + u_{ij}^s - p_j^{sk} \leq 1 - x_{ij}^{sk} \quad (8)$$

- origin access:

$$\forall s \in S, \quad \forall k = (o, d, D) \in K^s \quad p_o^{sk} = 0 \quad (9)$$

- complementary slackness condition on capacity:

$$\forall s \in S, \forall a \in A \quad c_a \leq \sum_{k \in K^s} f_a^{sk} + M^w(1 - w_a^s) \quad (10)$$

- binary flow constraint:

$$\forall s \in S, \forall k \in K^s, \forall a \in A \quad f_a^{sk} \leq M^f x_a^{sk} \quad (11)$$

- binary weight constraint:

$$\forall s \in S, \forall a \in A \quad u_a^s \leq w_a^s \quad (12)$$

- dual weights constraint:

$$\forall s \in S \quad \sum_{a \in A} u_a^s = 1 \quad (13)$$

- threshold bound:

$$\forall s \in S, \forall a \in A^s \quad c_a \geq \gamma_s \ell_a^s \quad (14)$$

- feasibility bound:

$$\forall s \in S, \forall a \in A \quad \sum_{k \in K^s} f_a^{sk} \leq c_a \quad (15)$$

- max norm (i):

$$\forall s \in S, \forall a \in A^s \quad \Delta \geq \gamma_s \ell_a^s - \sum_{k \in K^s} f_a^{sk} \quad (16)$$

- max norm (ii):

$$\forall s \in S, \forall a \in A^s \quad \Delta \geq \sum_{k \in K^s} f_a^{sk} - \gamma_s \ell_a^s. \quad (17)$$

The model aims at minimizing the maximal error Δ , which is the absolute value of the difference for each scenario and each arc between the load of f and the load $\gamma_s \ell_a^s$ (see constraints (16) and (17)) as stated in condition (ii) of the IMCFP. Constraints (5) and (6) are flow conservation constraints already mentioned in (1). Constraints (7), (8), (9), and (11) provide the system that ensures the optimality of the routing in the sense of the weights u , which correspond to the dual variables in (3) of capacity constraint in MCFP (2). Constraints (10) and (12) describe the slackness conditions for the dual variables of the capacities. Constraint (13) ensures a feasible dual solution of (3) for each scenario. Finally, Constraints (15) and (14) guarantee the feasibility of (2) as condition (i) of IMCFP. Note that c_a of (14), (15), and (10) can either be a parameter or a variable, depending whether a is in A' or not. The condition (iii) of IMCFP follows by Proposition 1 and Theorem 2 (below).

3.2 Correctness

Proposition 1 *Let $f = (f_a^k \mid a \in A, k \in K) \geq 0$ and $\gamma \geq 0$ be a feasible flow solution of the MCFP (2) and $p = (p_i^k \mid k \in K, i \in V) \geq 0$ and $u = (u_a \mid a \in A) \geq 0$ be a feasible solution of its dual (3). These two assertions are equivalent:*

1. (a) $\forall a \in A$, we have $u_a(c_a - \sum_{k \in K} f_a^k) = 0$
 (b) $\forall a = (i, j) \in A, \forall k \in K$, we have $f_a^k(p_i^k + u_a - p_j^k) = 0$
 (c) $\sum_{a \in A} u_a > 0$
2. f and γ are optimal for the MCF (2), and p and u (both scaled by $\sum_{k=(o,d,D) \in K} p_d^k D$) are optimal for its dual (3).

Proof. 2 \Rightarrow 1: If f and γ (resp. p and u) are optimal for (2) (resp. for (3)), the complementary slackness conditions [16] state immediately the first two equations. The third complementary slackness condition of MCFP states $\sum_{k \in K} p_d^k D = 1$. Therefore, (3) ensures there exists a so that $u_a > 0$, meaning $\sum_{a \in A} u_a > 0$.

1 \Rightarrow 2: If $\sum_{a \in A} u_a > 0$ and $\forall a \in A, u_a \geq 0$, there exists $a \in A$ so that $u_a > 0$.

The first condition implies that for this a we have $c_a = \sum_{k \in K} f_a^k$. We can assume the arcs have a non-zero capacity (otherwise the arc should not have existed), meaning there exists a $k = (o, d, D)$ for which $f_a^k > 0$. The second condition implies that going through this arc satisfies the ‘‘min cost path condition’’, meaning $p_d^k \geq u_a > 0$. We can then construct $Q = \sum_{k=(o,d,D) \in K} p_d^k D > 0$. Choose u'_a so

that $u'_a = \frac{u_a}{Q}$, and, for each $k = (o, d, D) \in K$, $p'_d^k = \frac{p_d^k}{Q}$. Then we trivially have

$\sum_{k=(o,d,D) \in K} p'^k_d D = 1$. Substituting u by u' and p by p' , one can observe that $\forall a \in A$, $u_a(c_a - \sum_{k \in K} f_a^k) = 0$ and $\forall a = (i, j) \in A, \forall k \in K$ $f_a^k(p_i^k + u_a - p_j^k) = 0$

still hold and (u, p) becomes a feasible solution of (3). The proposition follows from the LP complementary slackness conditions [16]. \square

Theorem 2 *Given a graph $G = (V, A)$ and a feasible solution (f, c) of IMCFP, for each $s \in S$, f^s corresponds to an optimal solution of an MCFP with respect to c and K^s .*

Proof. Let us choose a $s \in S$. f^s is feasible for MCFP (2) with parameters c and K^s due to (5), (6), (15). p^s and u^s describes a feasible solution of (3) with parameters c and K^s due to (13) and (7) (one can scale the solution to ensure $\sum_{k=(o,d,D) \in K^s} p_d^{sk} D \geq 1$). The combination of (10) and (12) ensure the first slackness condition, as $u_a^s = 0$ if the $c_a - \sum_{k \in K^s} f_a^{sk} > 0$. Constraints (7)-(8) validate the second slackness condition. The third slackness condition is obtained by (13). Proposition 1 validates the optimality of f^s for MCFP (2) with parameters c and K^s . As it holds for each $s \in S$, it concludes the proof. \square

4 Numerical results

We can solve IMCFP instances by simply formulating them as the Mixed-Integer Linear Program in Sect. 3.1 and solving them using an off-the-shelf solver. By the polynomial number of constraints and variables of our formulation, we know that the decision version of IMCFP is in **NP**. Although the MCFP is polynomial-time solvable, our IMCFP formulation introduces the need of binary variables to ensure optimality among the different scenarios, meaning it is still an open question whether IMCFP is **NP**-complete or not. To solve all the following instances, we solved the IMCFP model with the MILP solver IBM CPLEX 12.6 with default settings and a time limit of 1200 CPU seconds on a personal computer (Intel Core i7-6820HQ 2.70 GHz, 16 GB DDR3 RAM). All graphs used in the following experiments are based on the topology of the Paris subway network, often restricted to the left bank. Therefore, each node represents a connection between one or more different metro lines, and each arc represents a section of a line. The network is strongly connected, meaning we can generate complete sets of demands ($|K| = (|V| - 1)|V|$). The demand value for each commodity $k \in K$ is an integer uniformly chosen in the interval $[1, 10]$.

Firstly, to evaluate the prediction performance of the proposed formulations, we generated integer capacities for the MCFP in the interval $[1, 15]$ and then calculated an optimal solution of MCFP for each scenario of demands, keeping the same capacities C among them. It allows us to give the total configuration of loads ℓ_a^s and K^s as input where an optimal solution with zero Δ value exists. This let us compare the computed capacities in $A \setminus A'$, with those we chose to construct our instances of MCFP (the generated ones). Secondly, we generated ℓ_a^s , K^s , and c_a according to a feasible flow solution (in terms of conservation and capacity constraints) which is not MCFP compliant (i.e., whose input data do not follow an optimal MCFP solution pattern). The second set of instances

aims at observing how our model deals with data based on a wrong hypothesis (structure of an optimal solution of MCFP) and, hence, how the objective value is impacted. Moreover, we studied the impact of the quantity of known and unknown data by giving a fixed percentage of capacities for MCFP ($\frac{|A'|}{|A|} \times 100$) as an input.

The resulting tables are organized as follows. The number of nodes ($|V|$) and arcs ($|A|$) are reported in the caption of each table. The proportion of known capacities is specified in the first column “ $C(\%)$ ”. The rest of the table is divided in 3 subsets of columns, 4 for each value of cardinality of S . The four columns report: (i) the amount of capacities “ $c(\%)$ ” that has been successfully predicted (the percentage of successfully predicted capacities may be lower than the given ones due to the truncation process of “ $C(\%)|A'|$ ”); (ii) the maximal absolute gap (“Gap”) observed between the predicted capacities and the generated ones; (iii) the CPU time in seconds “ $T(s)$ ” (we denote termination due to time limit by *); and (iv) objective value “ Δ ”.

$C(\%)$	$ S = 3$				$ S = 4$				$ S = 5$			
	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ
0	42.11	13.00	2.95	0.00	42.11	14.00	4.31	0.00	63.16	7.00	6.11	0.00
10	60.53	9.72	2.64	0.00	65.79	11.00	3.07	0.00	63.16	9.00	7.43	0.00
20	68.42	11.00	2.12	0.00	71.05	10.00	118.00	0.00	73.68	10.00	4.82	0.00
30	60.53	14	1.83	0.00	78.95	8.58	4.48	0.00	76.32	9.00	*	0.04
40	44.74	12.83	2.18	0.00	81.58	6.00	3.00	0.00	73.68	13.00	5.40	0.00
50	84.21	10.46	2.14	0.00	86.84	6.00	3.31	0.00	78.95	11.77	6.80	0.00
60	81.58	10.00	2.59	0.00	81.58	7.00	4.12	0.00	89.47	13.00	14.09	0.00
70	81.58	9.26	2.01	0.00	89.47	9.00	2.11	0.00	84.21	12.72	6.05	0.00
80	86.84	12.00	2.56	0.00	97.37	6.00	3.09	0.00	94.74	11.25	6.63	0.00
90	100.00	0.00	1.90	0.00	97.37	0.69	3.46	0.00	92.11	4.00	9.95	0.00
100	100.00	0.00	1.42	0.00	100.00	0.00	2.47	0.00	100.00	0.00	3.74	0.00

Table 1. IMCFP, Paris left bank subway network topology ($|V| = 13, |A| = 38$).

$C(\%)$	$ S = 3$				$ S = 4$				$ S = 5$			
	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ
0	13.40	12.50	*	0.28	14.83	9.44	*	3.66	64.60	12.11	*	1.00
10	15.31	13.78	*	0.33	17.70	11.84	*	2.21	68.42	11.04	*	3.15
20	67.46	12.53	*	10.54	69.86	10.77	*	5.20	60.29	13.21	*	1.09
30	76.55	13.44	*	2.00	74.16	12.45	*	0.80	75.12	13.29	*	0.71
40	73.68	13.41	*	1.00	79.43	11.00	*	1.52	76.56	12.04	*	1.03
50	80.38	9.82	*	1.00	54.07	12.54	390.44	0.00	84.21	11.54	*	2.36
60	78.95	12.90	*	0.33	86.12	11.43	*	2.65	83.73	12.02	*	9.00
70	73.2	13.02	*	0.04	90.90	10.45	*	3.00	90.43	9.16	*	3.85
80	90.43	9.50	*	2.00	92.34	10.61	*	1.39	92.82	11.26	*	6.03
90	90.43	12.94	*	0.82	94.26	11.71	*	5.00	97.13	12.10	*	3.13
100	100.00	0.00	*	3.00	100.00	0.00	*	1.00	100.00	0.00	*	1.83

Table 2. IMCFP, Paris subway network topology ($|V| = 57, |A| = 209$).

$C(\%)$	$ S = 3$				$ S = 4$				$ S = 5$			
	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ	$c(\%)$	Gap	$T(s)$	Δ
0	0	13.40	1.87	0.00	10.53	13.54	*	0.2	5.26	11.97	8.25	0.00
10	15.79	12.62	*	0.34	10.53	13.18	121.12	0.00	10.53	10.24	*	0.15
20	23.68	14.06	*	0.30	28.95	9.67	4.48	0.00	21.05	12.61	5.07	0.00
30	28.95	12.64	*	0.00	28.95	13.66	4.31	0.00	31.58	13.53	4.1	0.00
40	39.47	11.36	*	2.00	42.11	12.26	*	0.95	50.00	12.89	*	1.00
50	55.26	8.95	*	3.17	50.00	11.93	6.50	0.00	57.89	10.09	*	1.00
60	57.90	12.63	*	0.43	57.89	9.55	*	1.12	65.79	11.53	5.16	1.00
70	68.42	11.89	859.08	1.00	68.42	12.93	2.67	0.00	73.68	9.00	7.28	1.00
80	81.58	7.49	2.21	1.00	78.95	9.47	4.00	2.00	81.58	8.69	5.16	1.00
90	89.47	12.08	1.16	0.50	92.10	12.08	1.07	1.00	89.47	9.89	6.33	1.35
100	100.00	0.00	0.52	14.00	100.00	0.00	0.66	11.67	100.00	0.00	0.80	14.13

Table 3. IMCFP, Paris left bank subway network topology ($|V| = 13, |A| = 38$). Input data not MCFP compliant.

The most striking thing one can note is the effectiveness of our methodology in predicting arc capacities (see Table 1). Even if without insights of capacities the prediction correctness remains low, it skyrockets to more than 50% of additional correct predictions with only three scenarios. This can be explained by the fact that even with few capacities the γ_s become strongly bounded. This proportion of corrected predictions tends naturally to arise when more and more capacity parameters are known, but the number of scenarios seems to remain a strong source of insights for the network, especially when the number of known capacities is low (see Table 2). However, the number of scenarios seems more and more important as the size of the instance grows. Indeed, the average per-

centage of correct predictions is quite close compared from $|S| = 4$ to $|S| = 5$ (around 81% for both) in the instances of Table 1 as if this increase did not bring further information. But this average percentage rises from 70% to 79% between $|S| = 4$ and $|S| = 5$ in the instances of Table 2. This could mean that the larger the instance, the higher the number of required scenarios to reach a similar prediction performance. It is also interesting to note that some of these results are obtained without even reaching optimality, especially on the total metro network instances, meaning that the routing problems are quite hard to solve even when all capacities are known. This is confirmed by the amount of time consumed by the instances when all the capacities are set. This suggests that MCFPs are also quite hard to solve. This is a well-known result that motivated the design of approximation schemes to solve it (see e.g. [8]). As one can expect, confronted with inconsistent data (input data that do not follow a structure of an optimal solution of the MCFP), the prediction effectiveness does not perform as well as the previous case (see Table 3). Nevertheless, our model often tends to bring a solution that fits significantly the input loads, especially when few capacities are known. Generally speaking, this means that a low objective value does not guarantee necessarily that the MCFP pattern hypothesis on our input data is right. However, when all capacities are set ($A' = A$), this seems to constraint drastically our model whose objective value skyrockets. Therefore, it suggests that our model can still be used to validate/invalidate this hypothesis on condition that all capacities are known.

5 Conclusion and future work

In this paper we studied a new problem called the IMCFP and proposed a solution technique to tackle it. The purpose of the new formulation proposed is to find a routing of commodities that fits input data at best (namely loads and demand matrix) as [15]. But, in addition, the solution has to follow an optimal structure of MCFP regarding a set of known and unknown data. The practical interest of the problem we discussed arises in transportation systems when it comes to recompose unknown data assuming hypothesis of data structure (namely a MCFP and one of its optimal solutions). Moreover, this formulation can validate/invalidate hypothesis of a MCFP's optimal routing by confronting it to observed data. More theoretically, we showed that this problem can be tackled by embedding primal and dual formulations and complementary slackness conditions all together into a single Mixed Integer Linear Program. We proved the correctness of such models, and led experiments to emphasize empirical behaviours and computational hardness. Future work will focus on computational performance improvements so that larger graphs can be treated.

References

1. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms, and applications*. Prentice Hall Inc., 1993.

2. R. K. Ahuja and J. B. Orlin. Inverse optimization, part I: Linear programming and general problem, 1998.
3. A. Atamturk and M. Zhang. Two-stage robust network flow and design under demand uncertainty. *Operations Research*, 55(4):662–673, 2007.
4. P. O. Banguion, W Ben-Ameur, and E. Gourdin. Efficient algorithms for the maximum concurrent flow problem. *Networks*, 65(1):56–67, 2015.
5. W. Ben-Ameur and H. Kerivin. Routing of uncertain traffic demands. *Optimization and Engineering*, 3:283–313, 2005.
6. A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Mathematical Programming*, 99(2):351–376, 2004.
7. D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.
8. D. Bienstock and O. Raskina. Asymptotic analysis of the flow deviation method for the maximum concurrent flow problem. *Mathematical Programming, Ser. B*, 91:479–492, 2002.
9. D. Burton and Ph. L. Toint. On an instance of the inverse shortest paths problem. *Mathematical Programming*, 53(1):45–61, 1992.
10. G. B. Dantzig. Linear programming under uncertainty. *Management Science*, 1(3-4):197–206, 1955.
11. J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.
12. L. R. Ford and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.
13. M. Gerla. A cut saturation algorithm for topological design of packet-switched communication networks. In *Proceedings of the National Telecommunication Conference*, pages 1074–1085, 1974.
14. C. Heuberger. Inverse combinatorial optimization: A survey on problems, methods, and results. *Journal of Combinatorial Optimization*, 8(3):329–361, 2004.
15. J. N. Hooker. *Inferring network flows from incomplete information with application to natural gas flows*. Oak Ridge National Laboratory, 1980.
16. H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, Calif., 1951. University of California Press.
17. Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang. Traffic flow prediction with big data: A deep learning approach. *IEEE Transactions on Intelligent Transportation Systems*, 16(2):865–873, 2015.
18. M. L. Manheim. *Fundamentals of transportation systems analysis*. Cambridge, Mass.: MIT Press, 1979.
19. C. Norton, S. Plotkin, and E. Tardos. Using separation algorithms in fixed dimension. *Journal of Algorithms*, 13:79–98, 1992.
20. F. Shahrokhi and D. W. Matula. The maximum concurrent flow problem. *Journal of the ACM*, 37(2):318–334, 1990.
21. S. Shiliang, Z. Changshui, and Y. Guoqiang. A bayesian network approach to traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):124–132, 2006.
22. A. L. Soyster. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Operation Research*, 21:1154–1157, 1973.
23. J. Zhang and Z. Liu. Calculating some inverse linear programming problems. *Journal of Computational and Applied Mathematics*, 72(2):261 – 273, 1996.