ORIGINAL ARTICLE

Affine Routing for Robust Network Design

Yacine Al-Najjar[1,2] | Walid Ben-Ameur[1] | Jérémie Leguay[2] | Jocelyne Elias[3]

[1]Samovar, Telecom SudParis, Institut Polytechnique de Paris, France.

[2]Huawei Technologies, Paris Research Center, France.

[3]Dipartimento di Informatica - Scienza e Ingegneria (DISI), University of Bologna, Italy.

Correspondence
Email: yacine.alnajjar@huawei.com

Funding information
CIFRE Grant from ANRT n°2019/0121.

Taking into account the dynamic nature of traffic in telecommunication networks, the robust network design problem is to fix the edge capacities so that all demand vectors belonging to a given polyhedral set can be routed. While a common heuristic for this co-NP hard problem is to compute, in polynomial time, an optimal static routing, affine routing can be used to obtain better solutions also with polynomial-time algorithms. It consists in restricting the routing to depend on the demands in an affine way. We first show that a node-arc formulation is less conservative than an arc-path formulation. We also provide a natural cycle-based formulation that is shown to be equivalent to the node-arc formulation. To further reduce the solution's cost, several new formulations are proposed. They are based on the relaxation of flow conservation constraints. The obtained formulations have been further improved through aggregation. As might be expected, aggregation allows us to reduce the size of formulations. A more striking result is that aggregation reduces the solution's cost. Finally, some numerical experiments are presented.

KEYWORDS
Network Design, Robust Optimization, Affine Routing, Adjustable Robust Optimization, Uncertainty.

*

1 | INTRODUCTION

Network optimization [34, 51] plays a crucial role for telecommunication operators allowing them to carefully invest in their infrastructure, i.e., reducing capital expenditures. As Internet traffic is ever increasing, the network capacity needs to be expanded through careful investments every year or even half-year. Beyond traditional carrier networks that build and maintain their own physical infrastructure, Over-The-Top (OTT) operators are also building wide area overlay networks to obtain worldwide, long-haul and cost effective services. These companies are renting and reselling bulk capacity from transit operators and Internet eXchange Points (IXP). For instance, SD-WAN (Software-Defined Wide Area Networks) operators build high-performance and low cost WANs (Wide Area Networks) for enterprises based on Software-Defined Networking [36] technologies by efficiently mixing expensive private lines with low cost Internet access. Other OTT such as cloud service providers or online platforms (e.g., video streaming, social networks) are leasing a mixture of connectivity services to interconnect central and regional data centers. In all cases, network connectivity, whether leased or owned, has a cost and needs to be carefully designed to optimize profits.

Ideally, the network capacity should follow the demand. However, in most of the cases, the demand varies over time and re-configurations can only be realized at a slow pace. When the traffic demand can be precisely known, several approaches have been proposed to solve the capacitated network design problem using for instance decomposition methods and cutting planes [23, 27, 46]. But in practice, perfect knowledge of future traffic is not available at the time the decision needs to be taken. The dynamic nature of the traffic due to ordinary daily fluctuations, long term evolution and unpredictable events requires us to consider uncertainty on traffic demands when dimensioning network resources. While overestimated traffic forecasts could be used to solve a deterministic optimization problem, it is likely to produce a costly over-provisioning of the network capacities, which is not acceptable. Therefore, robust optimization under uncertainty sets is a must for the design of network capacities. In this context, our paper proposes an in-depth study of *affine routing* [41, 44], a particular restriction of the robust network design problem [20] that permits its resolution with polynomial-time algorithms.

More formally, let us consider a directed graph $G = (V(G), E(G))$ representing a communication network. The traffic is characterized by a set of commodities $h \in \mathcal{H}$ and associated to different source and sink pairs. The routing of a commodity can be represented by a flow $f^h \in \mathbb{R}^{E(G)}$ of intensity d^h. To take into account the changing nature of the demand, d is assumed to be uncertain and more precisely to belong to a polyhedral set $\mathcal{D}$. The *polyhedral model* was introduced in [8, 9] as an extension of the *hose model* [25, 26], where limits on the total traffic going into (resp. out of) a node are considered. Other types of *polyhedra* are typically used for robust network design, in particular the *Budget* uncertainty set [19] that considers a maximum deviation for each nominal demand and a global budget for possible variations. Also of interest, the *All Routable Demands* uncertainty from [6] contains all demand vectors that can be routed through the network.

When solving a robust network design problem, several objective functions can be considered. Given a capacity c_e for each edge e, one might be interested in minimizing the congestion given by $\max_{e \in E(G)} \frac{u_e}{c_e}$ where u_e is the reserved capacity on edge e. Another common objective function is given by the linear reservation cost $\sum_{e \in E(G)} \lambda_e u_e$. This can also represent the average congestion by taking $\lambda_e = \frac{1}{c_e}$. The goal is to choose a reservation vector u so that the network has enough capacity to support any demand vector $d \in \mathcal{D}$, i.e., there exists a (fractional) routing serving every commodity such that the total flow on each edge e is less than the reservation u_e. As presented in [44], this problem can be viewed as a two-stage optimization where (first stage) capacity design decisions are made in the long term while the actual (second stage) routing is adjusted based on observed user demands. In practice, this second stage is realized using a centralized network controller that performs continuous traffic engineering, for instance a Path Computation Element (PCE) [42], or using a distributed routing protocol, such as OSPF [38], where link weights are periodically engineered to accommodate traffic.

The robust network design problem where a linear reservation cost is minimized was proved to be co-NP hard in [32, 37] when the graph is directed. A stronger co-NP hardness result is given in [21] where the graph is undirected (this implies the directed case result). Some exact solution methods for robust network design have been considered in [5, 22, 35, 50]. In the case where the objective function of minimizing congestion is considered, a well-known $O(\log n)$ approximation ratio was presented in [47]. In a recent study on the approximability of robust network design [2], we proved that the robust network design problem with minimum congestion or with linear reservation cost cannot be approximated within any constant factor. Using the ETH conjecture, we also obtained a $\Omega(\frac{\log n}{\log \log n})$ lower bound which implies that the $O(\log n)$ approximation ratio [47] is tight. Robust network design is also referred to as *dynamic routing* in the literature since the network is optimized such that any realization of the traffic matrix in the uncertainty set has its own routing.

Routing with uncertain demands has received significant interest from the community. As opposed to dynamic routing, *static routing* or *stable routing* was introduced in [8]: it consists in choosing a fixed flow x^h of value 1 for each commodity h. The actual flow $f^h(d)$ for the demand scenario d will then be scaled by the actual demand d^h of commodity h, i.e., $f^h(d) = d^h x^h$. Static routing is also called *oblivious routing* in [4, 6]. In this case, polynomial-time algorithms to compute optimal static routing (with respect to either congestion or linear reservation cost) have been proposed [4, 6, 8, 9] based on either duality or cutting-plane algorithms.

To further improve solutions of static routing and overcome complexity issues related to dynamic routing, a number of restrictions on routing have been considered to design polynomial-time algorithms (see [11, 20, 43] for more references). This includes, for example, the multi-static approach [3, 7, 12] where the uncertainty set is partitioned using some linear inequalities and routing is restricted to be static over each partition. This idea has been generalized in [48] to unrestricted covers of the uncertainty set and an extension to share the demand between routing templates, called volume routing, has been proposed in [53]. Further extensions appeared in [49]. An approach encompassing the previous approaches is the multipolar approach proposed in [10, 13]. Finally, based on affine adjustable robust counterparts introduced in [15], restricting the recourse to be an affine function of the uncertainties, [40, 41] applied affine routing for robust network design leading to an arc-path formulation. The performance of this framework has been extensively compared to the static and dynamic routing, both theoretically and empirically in [43, 44] where a node-arc formulation is studied. In practice, affine routing provides a good approximation of the dynamic routing while it can be solved in reasonable time thanks to polynomial-time solution methods.

In this paper, we extensively study affine routing formulations for the robust network design problem and we make the following contributions:

- We study the relationship between the original affine routing formulations [41, 44], namely the node-arc and the arc-path formulations. We show that the node-arc formulation (denoted by $\mathcal{F}_=$) is less conservative than (i.e., it reduces cost for some instances) the arc-path formulation (denoted by $\mathcal{F}_{path}$). We also derive a natural cycle-based formulation equivalent to the node-arc formulation but that uses fewer variables and constraints. (Section 2)
- We introduce two ways of relaxing the flow conservation constraints in the node-arc formulation (respectively denoted $\mathcal{F}_-$ and $\mathcal{F}_+$). We prove that this leads to feasible solutions and then we show that they can both dominate the standard node-arc formulation. (Section 3)
- We propose $\mathcal{F}_{cut}$, a cut based formulation, as an improved solution over both relaxed-flow conservation formulations. However we show that, unless $P = NP$, it cannot be solved in polynomial time. (Section 4)
- We combine the two relaxed-flow conservation formulations using an extended graph. We prove that this for-

mulation, called $\overline{\mathcal{F}}$, dominates both relaxations $\mathcal{F}_-$ and $\mathcal{F}_+$, and that it can be solved in polynomial time. (Section 5)

- To drastically reduce the size of models and consequently the solution times, we present variants of the formulations based on the aggregation of different flows having the same source and/or sink in the node-arc formulation. Moreover, we show that this can also improve the cost of the solutions. (Section 6)
- We numerically test the different formulations that can be solved with polynomial-time algorithms after a reformulation using a classical duality-based method. We compare the solutions and execution times for two polyhedra, two topologies and two objective functions. (Section 7)

To close this section, we just wanted to point out that the work presented in this paper falls in the framework of robust optimization and more specifically multi-stage adjustable robust optimization. Several general works can be cited here such as [10, 15, 16, 17, 18, 28, 33, 45]. Further references can be found in [24, 52].

2 | POSSIBLE AFFINE FORMULATIONS

We start by recalling some standard node-arc formulations that we will improve later in Section 3. Then we recall an arc-path formulation that might be more practical when paths can be enumerated easily and we show that it is dominated by node-arc formulations. Finally, we close this section by proposing a natural cycle-based formulation that is equivalent to node-arc formulations but with slightly fewer variables and constraints.

In the rest of the paper, we indifferently say that a formulation $\mathcal{F}_1$ dominates $\mathcal{F}_2$ (or is less conservative than $\mathcal{F}_2$) when the following holds:
- Starting from any feasible solution of $\mathcal{F}_2$, one can build a feasible solution of $\mathcal{F}_1$ having the same objective value.
- There exists at least one instance for which the value of the optimal objective function of $\mathcal{F}_1$ is strictly smaller than the one of $\mathcal{F}_2$.

2.1 | Initial node-arc formulation

Let's consider a directed graph G = (V, E) representing a communication network. The traffic is characterized by a set of commodities $h \in \mathcal{H}$ associated to different node pairs. For a given commodity h, the traffic originates at $s(h)$ and ends at $t(h)$. As introduced in [44], for each demand scenario, the flow $f_e^h(d)$ related to commodity h and edge $e \in E$, is restricted to affinely depend on the demand vector d. It represents the capacity reservation in the robust network design problem. This flow $f_e^h(d)$ is of the form $f_e^h(d) = x_e^{h0} + \sum_{h' \in \mathcal{H}} x_e^{hh'} d^{h'}$ where coefficients x_e^{h0} and $x_e^{hh'}$ are subject to optimization.

The affine routing with congestion minimization can be then modeled as follows:

$$\min m$$

$$\sum_{e\in\delta_+(v)}\left(x_e^{h0}+\sum_{h'\in\mathcal{H}}x_e^{hh'}d^{h'}\right)-\sum_{e\in\delta_-(v)}\left(x_e^{h0}+\sum_{h'\in\mathcal{H}}x_e^{hh'}d^{h'}\right)=\begin{cases}d_h, & \text{if } v=s(h)\\ -d_h, & \text{if } v=t(h)\\ 0 & \text{otherwise}\end{cases}$$

$$\forall h\in\mathcal{H}, v\in V, d\in\mathcal{D} \tag{1a}$$

$$\sum_{h\in\mathcal{H}}\left(x_e^{h0}+\sum_{h'\in\mathcal{H}}x_e^{hh'}d^{h'}\right)\le c_e m,\quad \forall e\in E, d\in\mathcal{D} \tag{1b}$$

$$x_e^{h0}+\sum_{h'\in\mathcal{H}}x_e^{hh'}d^{h'}\ge 0,\quad \forall e\in E,\forall h\in\mathcal{H}, d\in\mathcal{D} \tag{1c}$$

Constraints (1a) are standard flow conservation constraints, while (1b) express capacity limitation. Finally, constraints (1c) impose positivity on capacity reservations. $\delta_-(v)$ and $e\in\delta_+(v)$ respectively denote incoming and outgoing edges from node $v\in V$. Notice that the flow conservation constraint related to $v=t(h)$ can be skipped since it can be obtained by summing the constraints related to the other vertices.

While in practical applications, congestion can never go above 1, we keep calling m the congestion even if $m>1$. In fact, another interpretation of m can be obtained by considering the problem where demands can be multiplied by the same number γ that should be maximized and such that the "new" scaled demands can be carried by the network. This is the classical maximum concurrent-flow problem. Then m is just the inverse of the biggest value of γ. Thus $m>1$ occurs if $\gamma<1$ implying that only a fraction γ of demands can be routed through the network.

As already mentioned in Section 1, another objective function that is very often used in the literature is a linear objective function expressed as $\sum_{e\in E}\lambda_e u_e$ where λ_e are scalars corresponding to the unit cost of underlying resources and u_e are reserved capacity variables (so $c_e m$ is replaced by u_e in constraint (1b)). While we use the congestion function in the analytical sections of the paper, all proposed formulations can be naturally adapted to the linear objective case. Both objective functions will be considered in the numerical section.

$\mathcal{D}$ is supposed to be fully dimensional (it contains a ball). This assumption is not really restrictive since, in practice, one should not expect that there is any linear equation satisfied by all demand vectors. Moreover, if the assumption is not satisfied, then one can eliminate some of the coefficients $x_e^{hh'}$. For example, if we always have $d^{h'}=\sum_{h''}\alpha_{h''}d^{h''}$ then there is clearly no need to consider any dependency on h' implying that coefficients $x_e^{hh'}$ are useless. Under this hypothesis, two affine functions a_1,a_2 are equal over $\mathcal{D}$ (i.e., $a_1(d)=a_2(d),\forall d\in\mathcal{D}$), if and only if, $a_1=a_2$ (i.e., all coefficients of the affine functions a_1,a_2 are equal). Using this fact, as proposed in [44], we can replace the flow conservation constraints (1a) by the following equivalent reformulation:

$$\sum_{e\in\delta_+(v)}x_e^{hh'}-\sum_{e\in\delta_-(v)}x_e^{hh'}=\begin{cases}1, & \text{if } v=s(h)\text{ and } h=h'\\ -1, & \text{if } v=t(h)\text{ and } h=h'\\ 0 & \text{otherwise (including } h'=0)\end{cases} \tag{2}$$

The obtained formulation proposed by [44] is given below.

$$\min m$$

$$\sum_{e \in \delta_+(v)} x_e^{hh'} - \sum_{e \in \delta_-(v)} x_e^{hh'} = \begin{cases} 1, & \text{if } v = s(h) \text{ and } h = h' \\ -1, & \text{if } v = t(h) \text{ and } h = h' \\ 0 & \text{otherwise} \end{cases}$$

$$\forall h \in \mathcal{H}, h' \in \mathcal{H} \cup \{0\}, v \in V \tag{3a}$$

$$\sum_{h \in \mathcal{H}} \left(x_e^{h0} + \sum_{h' \in \mathcal{H}} x_e^{hh'} d^{h'} \right) \leq c_e m, \quad \forall e \in E, d \in \mathcal{D} \tag{3b}$$

$$x_e^{h0} + \sum_{h' \in \mathcal{H}} x_e^{hh'} d^{h'} \geq 0, \quad \forall e \in E, h \in \mathcal{H}, d \in \mathcal{D} \tag{3c}$$

As will be recalled in Section 7, when some linear constraints with uncertain coefficients need to be considered, we can handle uncertainty either by using constraint generation or by duality-based reformulation techniques. This should be done for each constraint. The main advantage of (3) is that the uncertainty appearing in the flow conservation constraints (1a) is already handled using (2). However, the other constraints (3b) and (3c) still need to be treated using the techniques briefly recalled in Section 7 and used for the numerical evaluation.

2.2 | Arc-path formulation

Another natural formulation is the one obtained by considering path variables. This might lead to solution methods that are easier to implement in communication networks when only a small number of paths is used for each commodity or the total number of paths that could be handled by each router/node is limited. Due to engineering rules, certain types of paths might not be used. For example, when Internet routing protocols are considered, one can only route along shortest paths in the sense of some administrative weights.

As proposed in [41], the flow on each path affinely depends on the demand vector d. This leads to the following model, denoted by $\mathcal{F}_{path}$, where $\mathcal{P}^h$ is a set of (possibly all) paths from $s(h)$ to $t(h)$.

$$\min m$$

$$\sum_{h \in \mathcal{H}} \sum_{p \in \mathcal{P}^h : p \ni e} \left(x_p^{h0} + \sum_{h' \in \mathcal{H}} x_p^{hh'} d^{h'} \right) \leq c_e m, \qquad \forall e \in E, d \in \mathcal{D} \tag{4a}$$

$$\sum_{p \in \mathcal{P}^h} \left(x_p^{h0} + \sum_{h' \in \mathcal{H}} x_p^{hh'} d^{h'} \right) = d^h, \qquad \forall h \in \mathcal{H}, d \in \mathcal{D} \tag{4b}$$

$$x_p^{h0} + \sum_{h' \in \mathcal{H}} x_p^{hh'} d^{h'} \geq 0, \qquad \forall h \in \mathcal{H}, p \in \mathcal{P}^h, d \in \mathcal{D} \tag{4c}$$

Observe that constraints (4c) impose the non-negativity of the flow on each path. Notice that when there is no uncertainty (i.e., when $\mathcal{D}$ contains only one demand vector), then (1) and (4) are equivalent when $\mathcal{P}^h$ contains all possible paths. One might wonder whether this still holds for any $\mathcal{D}$. Each solution of (4) can be transformed into a feasible solution of (1) and (3) (by setting $x_e^{h0} = \sum_{p \in \mathcal{P}^h : p \ni e} x_p^{h0}$ and $x_e^{hh'} = \sum_{p \in \mathcal{P}^h : p \ni e} x_p^{hh'}$). However, the following example shows that (4) can reduce congestion compared to (1) even if all paths are considered.

Proposition 2.1. *Formulation* (1) *is less conservative than formulation* (4).

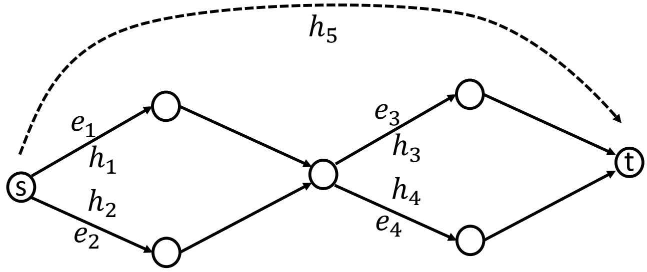

FIGURE 1 An example with five commodities: h_i with $i = 1, 2, 3, 4$ having the same source and sink as edges e_i, and h_5 from source node s to sink node t. In this example we show that formulation (1) is less conservative than formulation (4).

Proof As mentioned above, starting from a feasible solution of (4), one can build a solution of (1) having the same congestion just by identifying x_e^{h0} with $\sum_{p \in \mathcal{P}^h : p \ni e} x_p^{h0}$ and $x_e^{hh'}$ with $\sum_{p \in \mathcal{P}^h : p \ni e} x_p^{hh'}$. Let us then find an instance for which the optimal objective value of (1) is strictly smaller than the one of (4). In the example of Figure 1, there is a demand h_1 (resp. $h_2, h_3, \text{ and } h_4$) having the same source and sink as edge e_1 (resp. $e_2, e_3, \text{ and } e_4$) and a demand h_5 having the node s as a source and the node t as sink. The demand polyhedron $\mathcal{D}$ is defined as the set of $d \in \mathbb{R}^5$ satisfying the equations $d^{h_1} + d^{h_2} = 1, d^{h_3} + d^{h_4} = 1$ and $d^{h_5} = 1$ in addition to non-negativity constraints. The capacity of each edge is equal to 1.

This demand polyhedron can be routed with model (1) without exceeding one unit of flow on each edge. This can be seen by considering the following solution: $f_{e_i}^{h_i}(d) = d^{h_i}$, $f_{e_i}^{h_5}(d) = 1 - d^{h_i}$ $\forall i = 1, \ldots, 4$ and $f_{e_j}^{h_i}(d) = 0$ $\forall i, j = 1, \ldots, 4, i \neq j$. By taking $m = 1$, all capacity constraints and flow conservation constraints of (1) are satisfied.

Now we are going to show that a solution to model (4) necessarily uses strictly more than one unit of flow on at least one edge. By contradiction, suppose that there exists a solution of (4) such that $m = 1$. First observe that given the equalities defining the demand polyhedron $\mathcal{D}$, it is sufficient that the affine flow function depends on the demand's values d^{h_1} and d^{h_3}. Let p_1 denote the path that uses edges e_1 and e_3, while p_2 contains edges e_1 and e_4, p_3 includes edges e_2 and e_3, and p_4 goes through edges e_2 and e_4. Notice that by considering $f_{p_i}^{h_5}(d) \geq 0$ for the vector d defined by $d^{h_1} = d^{h_2} = 0$, we find that all variables $x_{p_1}^{h_5 0}$, $x_{p_2}^{h_5 0}$, $x_{p_3}^{h_5 0}$ and $x_{p_4}^{h_5 0}$ are non-negative. Since the total flow that uses edge e_i must be less than 1 and the demand h_i must necessarily use edge e_i for $i = 1, \ldots, 4$, we have the following inequalities:

$$1 - d^{h_1} \geq f_{p_1}^{h_5}(d) + f_{p_2}^{h_5}(d) \quad (5a)$$

$$d^{h_1} \geq f_{p_3}^{h_5}(d) + f_{p_4}^{h_5}(d) \quad (5b)$$

$$1 - d^{h_3} \geq f_{p_1}^{h_5}(d) + f_{p_3}^{h_5}(d) \quad (5c)$$

$$d^{h_3} \geq f_{p_2}^{h_5}(d) + f_{p_4}^{h_5}(d) \quad (5d)$$

By summing inequalities (5), we get $2 \geq 2 \times \sum_{i=1,\ldots,4} f_{p_i}^{h_5}(d)$. Since $\sum_{i=1,\ldots,4} f_{p_i}^{h_5}(d) = d^{h_5} = 1$, all inequalities (5) should be equalities. Remember that for each path p_i, we have $f_{p_i}^{h_5}(d) = x_{p_i}^{h_5 0} + x_{p_i}^{h_5 h_1} d^{h_1} + x_{p_i}^{h_5 h_3} d^{h_3}$. Writing the four equalities above for the vector d where $d^{h_1} = d^{h_3} = 0$, we get: $1 = x_{p_1}^{h_5 0} + x_{p_2}^{h_5 0}, 0 = x_{p_3}^{h_5 0} + x_{p_4}^{h_5 0}, 1 = x_{p_1}^{h_5 0} + x_{p_3}^{h_5 0}$, $0 = x_{p_2}^{h_5 0} + x_{p_4}^{h_5 0}$. This implies that $x_{p_2}^{h_5 0} = x_{p_3}^{h_5 0} = -x_{p_4}^{h_5 0}$. Using the non-negativity of $x_{p_1}^{h_5 0}$, $x_{p_2}^{h_5 0}$, $x_{p_3}^{h_5 0}$ and $x_{p_4}^{h_5 0}$, one can deduce that

$x_{p_1}^{h_5 0} = 1, x_{p_i}^{h_5 0} = 0, \forall i = 2, 3, 4.$

Furthermore, by considering the positivity constraint (4c) for path p_2 and the demand vector where $d^{h_1} = 1$ and $d^{h_3} = 0$, we can deduce that $f_{p_2}^{h_5}(d) = x_{p_2}^{h_5 0} + x_{p_2}^{h_5 h_1} = x_{p_2}^{h_5 h_1} \geq 0$. Writing equality (5a) leads to $-1 = x_{p_1}^{h_5 h_1} + x_{p_2}^{h_5 h_1}$. Combination with the previous inequality implies that $x_{p_1}^{h_5 h_1} \leq -1$. Similarly, by considering the demand vector where $d^{h_3} = 1$ and $d^{h_1} = 0$, the positivity constraint related to path p_3 and equality (5c) lead to $x_{p_1}^{h_5 h_3} \leq -1$. Let us now consider the case where $d^{h_3} = 1$ and $d^{h_1} = 1$. The positivity of $f_{p_1}^{h_5}(d)$ is equivalent to $x_{p_1}^{h_5 0} + x_{p_1}^{h_5 h_1} + x_{p_1}^{h_5 h_3} \geq 0$ implying that $x_{p_1}^{h_5 h_1} + x_{p_1}^{h_5 h_3} \geq -1$. This is clearly not possible since $x_{p_1}^{h_5 h_1} \leq -1$ and $x_{p_1}^{h_5 h_3} \leq -1$, and it ends the proof. □

2.3 | Cycle-based formulation

We have seen that the arc-path formulation is dominated by the node-arc formulation. The main reason for that is the positivity constraint imposed for each path and each $d \in \mathcal{D}$. Then, if one tries to relax these positivity constraints and replace them by constraints expressing the fact that the total flow on each directed edge is non-negative (i.e., $f_e^h(d) \geq 0$ for each h, e and d), then we will get a new formulation where circulations appear. However, an easy way to present the new formulation is proposed below starting from formulation (3).

Let us first recall a basic result (e.g., see [29]) about circulation decomposition as a sum of elementary circulations through elementary cycles. We fix a spanning tree T of the graph G (supposed to be connected). For each $e \in E(G) \backslash E(T)$, there is a unique elementary cycle σ in $T \cup \{e\}$. We denote by χ_σ this circulation that has a value of 1 on the edges oriented in the same direction as edge e, -1 in the other direction and 0 on the edges outside σ. We denote by $\Sigma(T)$ the set of cycles. It is well-known that every circulation ϕ can be (uniquely) written as: $\phi = \sum_{\sigma \in \Sigma(T)} x_\sigma \chi_\sigma$ for some scalars $x_\sigma \in \mathbb{R}$.

Let us now go back to constraints (3a). For each commodity h, let p^h be any arbitrary fixed undirected path connecting $s(h)$ and $t(h)$ in T and let χ_{p^h} be the flow of value 1 on p^h and zero elsewhere (the value of the flow on each edge is either 1 or -1 depending on the direction of the edge). Since $x^{hh} - \chi_{p^h}$ is a circulation, x^{hh} can be written as: $x^{hh} = \chi_{p^h} + \sum_{\sigma \in \Sigma(T)} x_\sigma^{hh} \chi_\sigma$. Furthermore, for $h \neq h'$, $x^{hh'}$ is a circulation and thus it can be written as $x^{hh'} = \sum_{\sigma \in \Sigma(T)} x_\sigma^{hh'} \chi_\sigma$. We then obtain the new model (6) by substituting x_e^{hh} in model (1) by $\chi_{p^h,e} + \sum_{\sigma \in \Sigma(T)} x_\sigma^{hh} \chi_{\sigma,e}$ and replacing $x_e^{hh'}$ by $\sum_{\sigma \in \Sigma(T)} x_\sigma^{hh'} \chi_{\sigma,e}$.

$$
\begin{aligned}
& \min m \\
& \sum_{h \in \mathcal{H}} \left(\sum_{\sigma \ni e} \left(x_\sigma^{h0} \chi_{\sigma,e} + \sum_{h' \in \mathcal{H}} x_\sigma^{hh'} \chi_{\sigma,e} d^{h'} \right) + \sum_{p^h \ni e} \chi_{p^h,e} d^h \right) \leq c_e m, \quad \forall e \in E, d \in \mathcal{D} \\
& \sum_{\sigma \ni e} \left(x_\sigma^{h0} \chi_{\sigma,e} + \sum_{h' \in \mathcal{H}} x_\sigma^{hh'} \chi_{\sigma,e} d^{h'} \right) + \sum_{p^h \ni e} \chi_{p^h,e} d^h \geq 0, \quad \forall e \in E, h \in \mathcal{H}, d \in \mathcal{D} \\
& x_\sigma^{hh'} \in \mathbb{R}, \quad \forall \sigma \in \Sigma(T), h \in \mathcal{H}, h' \in \mathcal{H} \cup \{0\}
\end{aligned} \tag{6}
$$

Observe that since the number of elementary cycles $|\Sigma(T)|$ is equal to the cyclomatic number $|E| - |V| + 1$, the number of x variables in (6) is equal to $(|\mathcal{H}| + 1) \times |\mathcal{H}| \times (|E| - |V| + 1)$ whereas formulation (3) has $|\mathcal{H}| \times (|\mathcal{H}| + 1) \times |E|$ variables (there are also variables related to duality to take into account uncertainty as will be recalled in Section 7 but their number is the same in both formulations). Then formulation (6) has around $|\mathcal{H}|^2 \times |V|$ fewer variables than formulation (3). Formulation (6) has also around $|\mathcal{H}|^2 \times |V|$ fewer constraints than formulation (3) due to constraints (3a). Formulation (6) is obviously equivalent to formulations (3) and (1) since it was obtained from (3) using decomposition.

3 | RELAXING THE FLOW CONSERVATION CONSTRAINTS

In this section, we present some improvements of the node-arc formulation (1) described in Section 2, by relaxing the flow conservation constraints (1a). Such improvements permit us to further reduce the congestion and minimize the gap with the solution given by the dynamic routing. The standard formulation (1) might be denoted by $\mathcal{F}_=$ ("=" means that we have equalities in constraints (1a)). Let $\mathcal{F}_+$ be the formulation obtained from (1) by replacing (1a) by the following inequalities.

$$\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d) \begin{cases} \geq d^h, & \text{if } v = s(h) \\ \geq 0 & \text{if } v \neq s(h), t(h) \end{cases} \tag{7}$$

Notice that by summing all inequalities (for some h) we obtain $\sum_{e\in\delta_-(t(h))} f_e^h(d) - \sum_{e\in\delta_+(t(h))} f_e^h(d) \geq d^h$ which is equivalent to $\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d) \leq -d^h$ for $v = t(h)$. Since the quantities $f_e^h(d)$ no longer satisfy flow conservation constraints, we have no more the notion of flow. However, we can interpret $f_e^h(d)$ as being the amount of resources that is reserved for commodity h on edge e when the demand scenario d is considered. We will prove that for each demand vector d, it is possible to route each commodity h without exceeding the capacity $f_e^h(d)$ of edge e. We will then say that $\mathcal{F}_+$ is a valid formulation. This concept of valid formulation will also be used in the rest of the paper for any formulation that computes $f_e^h(d)$, allowing the routing of each commodity h for each $d \in \mathcal{D}$.

Proposition 3.1. *$\mathcal{F}_+$ is a valid formulation.*

Proof Consider any commodity $h \in \mathcal{H}$ and any cut $\delta_+(C)$ separating $s(h)$ and $t(h)$ (so $C \subset V$, $s(h) \in C$ and $t(h) \notin C$). By summing all constraints (7) for vertices inside C, we get

$$\sum_{e\in\delta_+(C)} f_e^h(d) - \sum_{e\in\delta_-(C)} f_e^h(d) = \sum_{v\in C}\left(\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d)\right) \geq d^h.$$

Using the positivity constraint on capacities $f_e^h(d)$, we deduce that $\sum_{e\in\delta_+(C)} f_e^h(d) \geq d^h$. Since this holds for any separating cut, it is possible by the maximum-flow minimum-cut theorem to send from $s(h)$ to $t(h)$ a flow of value d^h using the capacities $f_e^h(d)$. □

Since $\mathcal{F}_+$ is obtained from $\mathcal{F}_=$ by relaxing some constraints, the congestion m computed by $\mathcal{F}_+$ is less than or equal to the congestion given by $\mathcal{F}_=$. One might wonder whether there is any gain by considering $\mathcal{F}_+$ instead of $\mathcal{F}_=$. The example given below positively answers the question.

Proposition 3.2. *Formulation $\mathcal{F}_+$ is less conservative than formulation $\mathcal{F}_=$.*

Proof Each solution of $\mathcal{F}_=$ is a feasible solution of $\mathcal{F}_+$. Moreover, Figure 2 illustrates a simple graph with two commodities $h_{i,i=1,2}$ having the same source and sink as edges $e_{i,i=1,2}$. There is an additional commodity h_3 having node s

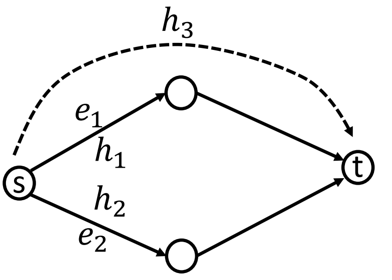

FIGURE 2 An example with three commodities: h_i with $i = 1, 2$ having the same source and sink as edges e_i, and h_3 from source node s to sink node t. In this example we show that $\mathcal{F}_+$ dominates formulation $\mathcal{F}_=$.

as source and node t as sink. All edges have a capacity equal to 1. The polyhedron $\mathcal{D}$ is defined as the set of $d \in \mathbb{R}^3_+$ satisfying the two inequalities $d_{h_1} + d_{h_2} \leq 1$ and $d_{h_3} \leq 1$ in addition to non-negativity constraints.

First, observe that the solution given by $f^{h_3}_{e_1}(d) = 1 - d^{h_1}$, $f^{h_3}_{e_2}(d) = d^{h_1}$, $f^{h_1}_{e_1}(d) = d^{h_1}$, $f^{h_1}_{e_2}(d) = 0$, $f^{h_2}_{e_2}(d) = d^{h_2}$, $f^{h_2}_{e_1}(d) = 0$, satisfies the constraints of $\mathcal{F}_+$. Consequently, $m = 1$ is the optimal congestion found by $\mathcal{F}_+$.

Let us now show that any solution of $\mathcal{F}_=$ should necessarily use more than one unit of flow on at least one edge. By contradiction, let's assume that there exists a solution of $\mathcal{F}_=$ such that $m = 1$. First, observe that if the demand for a commodity, let say h_3, is equal to zero then the flow for this commodity must also be 0 in model (1) (i.e., $\mathcal{F}_=$). This is due to the fact that we are dealing here with flows and there are no directed cycles in the graph. Consequently, for each edge $e \in E$ we have $f^{h_3}_e(0) = x^{h_3 0}_e = 0$. Considering the demand vector 1_{h_1} where $d^{h_1} = 1$ and the two other demands are 0, we obtain $f^{h_3}_e(1_{h_1}) = x^{h_3 0}_e + x^{h_3 h_1}_e = 0$ leading to $x^{h_3 h_1}_e = 0$. Similarly, by considering the demand vector 1_{h_2}, we prove that $x^{h_3 h_2}_e = 0$. Combining the previous facts leads to $f^{h_3}_e(d) = x^{h_3 h_3}_e d^{h_3}$.

Let us consider the demand vector $1_{h_1} + 1_{h_3}$ ($d^{h_1} = 1$, $d^{h_2} = 0$, $d^{h_3} = 1$). Then $f^{h_1}_{e_1}(1_{h_1} + 1_{h_3}) = 1$ since the only path to route the demand h_1 is through e_1. Moreover, the assumption $m = 1$ implies that $f^{h_3}_{e_1}(1_{h_1} + 1_{h_3}) + f^{h_1}_{e_1}(1_{h_1} + 1_{h_3}) \leq 1$. Combining the two observations leads to $f^{h_3}_{e_1}(1_{h_1} + 1_{h_3}) \leq 0$. Using the positivity constraint, we simply get $f^{h_3}_{e_1}(1_{h_1} + 1_{h_3}) = 0$. Using the fact that $f^{h_3}_{e_1}(d) = x^{h_3 h_3}_{e_1} d^{h_3}$, we finally deduce that $x^{h_3 h_3}_{e_1} = 0$.

Using a similar argument, we find that $f^{h_3}_{e_2}(1_{h_2} + 1_{h_3}) = x^{h_3 h_3}_{e_2} = 0$. Hence, all coefficients related to commodity h_3 are zero which is nonsense. □

Remark 3.1. *Observe that the polytope used in the proof of Proposition 4.1 contains dominated demand vectors. Since it has been shown in [44] that removing dominated demands can improve the solution of the affine routing, one can legitimately wonder if the observation that formulation $\mathcal{F}_+$ is less conservative than formulation $\mathcal{F}_=$ is only due to the existence of dominated demand vectors. It turns out that this is not true. We will now describe how the instance used in the proof of Proposition 4.1 can be simply modified so that the demand polytope does not contain dominated vectors. Let us add two isolated edges e_5, e_6 of capacity 1 and two new commodities h_5 and h_6 having the same extremities as e_5 and e_6, respectively. We replace the constraints $d^{h_1} + d^{h_2} \leq 1$ and $d^{h_3} \leq 1$ by the constraints $d^{h_1} + d^{h_2} + d^{h_5} = 1$ and $d^{h_3} + d^{h_6} = 1$. This new polytope does not contain dominated demand vectors. The projection of the new polytope on the variables $d^{h_1}, d^{h_2}, d^{h_3}$ is exaclty the polytope considered in the proof above. Since $d^{h_5} = 1 - d^{h_1} - d^{h_2}$ and $d^{h_6} = 1 - d^{h_3}$, an affine solution depending on $d^{h_1}, d^{h_2}, d^{h_3}, d^{h_5}, d^{h_6}$ can then be rewritten as an affine function of $d^{h_1}, d^{h_2}, d^{h_3}$. Therefore the set of affine solutions for commodities h_1, h_2, h_3 remains the same and $\mathcal{F}_+$ is still less conservative than formulation $\mathcal{F}_=$.*

Another way to relax the flow conservation constraints consists in replacing (1a) by the following inequalities:

$$\sum_{e \in \delta_+(v)} f_e^h(d) - \sum_{e \in \delta_-(v)} f_e^h(d) \begin{cases} \leq -d^h, & \text{if } v = t(h) \\ \leq 0 & \text{if } v \neq s(h), t(h) \end{cases} \tag{8}$$

The obtained formulation can then be called the $\mathcal{F}_-$ formulation which is completely symmetrical to $\mathcal{F}_+$. The validity of $\mathcal{F}_-$ can be shown in almost the same way (the proof of Proposition 3.1 can be modified by summing inequalities (8) through all vertices belonging to $V \setminus C$). Proposition 4.1 also holds for $\mathcal{F}_-$ (the same example provided in the proof can still be used). While both $\mathcal{F}_-$ and $\mathcal{F}_+$ dominate $\mathcal{F}_=$, they are not comparable (for some instances $\mathcal{F}_-$ provides better results than $\mathcal{F}_+$ and vice-versa).

4 | A CUT-BASED FORMULATION

To show the validity of either $\mathcal{F}_+$ or $\mathcal{F}_-$, we just proved that every cut separating the source and the sink of a commodity has enough capacity to carry the demand. This suggests the advantage of proposing a formulation based on cuts. More precisely, for each commodity $h \in \mathcal{H}$ and each cut $\delta_+(C)$ separating $s(h)$ and $t(h)$ ($C \subset V$, $s(h) \in C$ and $t(h) \notin C$) we require that

$$\sum_{e \in \delta_+(C)} f_e^h(d) \geq d^h. \tag{9}$$

The cut formulation denoted by $\mathcal{F}_{cut}$ is then obtained from (1) by replacing flow conservation constraints (1a) by (9). $\mathcal{F}_{cut}$ is obviously a valid formulation.

Since solutions of $\mathcal{F}_+$ and $\mathcal{F}_-$ satisfy the constraints of $\mathcal{F}_{cut}$, the solution provided by $\mathcal{F}_{cut}$ is at least as good as those of either $\mathcal{F}_+$ or $\mathcal{F}_-$. We provide below an example showing that $\mathcal{F}_{cut}$ can dominate $\mathcal{F}_+$ and $\mathcal{F}_-$.

Proposition 4.1. *Formulation $\mathcal{F}_{cut}$ is less conservative than $\mathcal{F}_+$ and $\mathcal{F}_-$.*

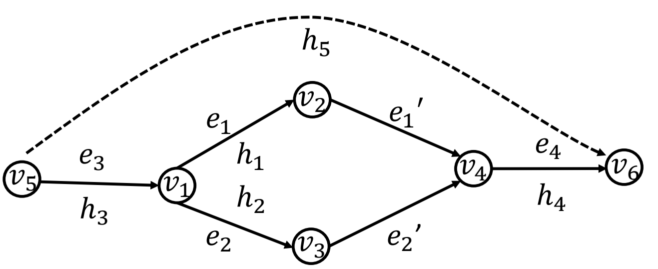

FIGURE 3 An example with five commodities: h_i with $i = 1, 2, 3, 4$ having the same source and sink as edges e_i, and h_5 from source node v_5 to sink node v_6. In this example we show that $\mathcal{F}_{cut}$ dominates $\mathcal{F}_+$ and $\mathcal{F}_-$.

Proof Each feasible solution of either $\mathcal{F}_+$ and $\mathcal{F}_-$ is a feasible solution of $\mathcal{F}_{cut}$. Consider the graph of Figure 3. It contains 6 directed edges each of capacity 1: $e_1 = v_1v_2$, $e_2 = v_1v_3$, $e'_1 = v_2v_4$, $e'_2 = v_3v_4$, $e_3 = v_5v_1$ and $e_4 = v_4v_6$. 5 commodities have to be carried through the network: h_i, $i = 1, 2, 3, 4$ having the same source and sink as edges e_i, and

h_5 from source node v_5 to sink node v_6. The polyhedral uncertainty set $\mathcal{D}$ is defined by constraints: $d^{h_1} + d^{h_2} = 1$, $d^{h_3} + d^{h_5} = 1$ and $d^{h_4} + d^{h_5} = 1$ in addition to non-negativity constraints. We can then assume that routing depends only on d^{h_1} and d^{h_5}.

Let us first show that the optimal solution of $\mathcal{F}_{cut}$ has a congestion $m = 1$. Consider the following solution: $f^{h_5}_{e_1}(d) = f^{h_5}_{e'_1}(d) = 1 - d^{h_1}$, $f^{h_5}_{e_2}(d) = f^{h_5}_{e'_2}(d) = 1 - d^{h_2}$ and $f^{h_5}_{e_3}(d) = f^{h_5}_{e_4}(d) = d^{h_5}$. Observe that each cut separating v_5 and v_6 that either contains e_3 or e_4 has a capacity greater than or equal to d^{h_5}. Moreover, a separating cut containing neither e_3 nor e_4 will necessarily contain either e_1 or e'_1 and either e_2 or e'_2. The capacity of such cut will then be at least $1 - d^{h_1} + d^{h_1} = 1 \geq d^{h_5}$. Let us now show that the congestion obtained by solving either $\mathcal{F}_+$ or $\mathcal{F}_-$ is $\frac{4}{3}$. Consider the following solution: $f^{h_5}_{e_1}(d) = \frac{1}{3} - \frac{d^{h_1}}{3} + \frac{d^{h_5}}{3}$, $f^{h_5}_{e_2}(d) = \frac{d^{h_1}}{3} + \frac{d^{h_5}}{3}$, $f^{h_5}_{e_3}(d) = f^{h_5}_{e_4}(d) = \frac{1}{3} + \frac{2}{3}d^{h_5}$ while the assignment related to the other commodities is obvious since only one path is available for each of them. The congestion m related to this solution is $m = \frac{4}{3}$. Observe that this solution is feasible for both $\mathcal{F}_+$ and $\mathcal{F}_-$. Let us prove that any solution of, for example $\mathcal{F}_+$, cannot have a congestion that is strictly less than $\frac{4}{3}$. The following inequalities are valid:

$$
\begin{aligned}
-x^{h_5 0}_{e_1} - x^{h_5 h_1}_{e_1} &\leq 0 && \text{(10a)}\\
x^{h_5 0}_{e_1} + x^{h_5 h_1}_{e_1} + 1 + x^{h_5 h_5}_{e_1} &\leq m && \text{(10b)}\\
-x^{h_5 0}_{e_2} &\leq 0 && \text{(10c)}\\
x^{h_5 0}_{e_2} + x^{h_5 h_5}_{e_2} + 1 &\leq m && \text{(10d)}\\
-x^{h_5 0}_{e_3} - x^{h_5 h_1}_{e_3} - x^{h_5 h_5}_{e_3} &\leq -1 && \text{(10e)}\\
(x^{h_5 0}_{e_3} - x^{h_5 0}_{e_1} - x^{h_5 0}_{e_2}) + (x^{h_5 h_1}_{e_3} - x^{h_5 h_1}_{e_1} - x^{h_5 h_1}_{e_2}) + (x^{h_5 h_5}_{e_3} - x^{h_5 h_5}_{e_1} - x^{h_5 h_5}_{e_2}) &\leq 0 && \text{(10f)}\\
(x^{h_5 0}_{e_1} + x^{h_5 0}_{e_2}) + (x^{h_5 h_1}_{e_1} + x^{h_5 h_1}_{e_2}) + 1 &\leq m. && \text{(10g)}
\end{aligned}
$$

Constraint (10a) is obtained by writing $f^{h_5}_{e_1}(1_{h_1}) \geq 0$. Constraint (10b) is a consequence of $f^{h_5}_{e_1}(1_{h_1}+1_{h_5})+f^{h_1}_{e_1}(1_{h_1}+1_{h_5}) \leq m$, while constraint (10c) follows from $f^{h_5}_{e_2}(0) \geq 0$. Writing $f^{h_5}_{e_2}(1_{h_5}) + f^{h_2}_{e_2}(1_{h_5}) \leq m$ leads to (10d). Using $f^{h_5}_{e_3}(d) \geq d^{h_5}$ for $d = 1_{h_1} + 1_{h_5}$ implies (10e). Since we have considered a feasible solution of $\mathcal{F}_+$, $f^{h_5}_{e_1}(1_{h_1} + 1_{h_5}) + f^{h_5}_{e_2}(1_{h_1} + 1_{h_5}) \geq f^{h_5}_{e_3}(1_{h_1} + 1_{h_5})$ is valid which is equivalent to (10f). More generally, we have $f^{h_5}_{e_1}(d) + f^{h_5}_{e_2}(d) \geq f^{h_5}_{e_3}(d)$, $f^{h_5}_{e'_1}(d) \geq f^{h_5}_{e_1}(d)$, $f^{h_5}_{e'_2}(d) \geq f^{h_5}_{e_2}(d)$ and $f^{h_5}_{e_4}(d) \geq f^{h_5}_{e'_2}(d) + f^{h_5}_{e'_1}(d)$. This implies that $f^{h_5}_{e_4}(d) \geq f^{h_5}_{e_3}(d)$ for any vector d. Combining this inequality with $f^{h_5}_{e_4}(d) + f^{h_4}_{e_4}(d) \leq m$ for $d = 1_{h_1}$ immediately gives (10g). Summing up all inequalities (10) leads to $4 \leq 3m$ ending the proof. □

We already mentioned in Section 1 that the variant where routing is dynamic is theoretically difficult to solve. Let us use $\mathcal{F}_{dyn}$ to denote the corresponding formulation. The difference between $\mathcal{F}_{cut}$ and $\mathcal{F}_{dyn}$ lies in the affine form of $f^h_e(d)$ that is imposed only for $\mathcal{F}_{cut}$. A possible formulation $\mathcal{F}_{dyn}$ is given below.

$$
\begin{aligned}
&\min m\\
&\sum_{e \in \delta_+(C)} f^h_e(d) \geq d^h, \forall d \in \mathcal{D}, h \in \mathcal{H}, C \subset V, s(h) \in C, t(h) \notin C\\
&\sum_{h \in \mathcal{H}} f^h_e(d) \leq c_e m, \quad \forall e \in E, d \in \mathcal{D}\\
&f^h_e(d) \geq 0, \quad \forall e \in E, h \in \mathcal{H}, d \in \mathcal{D}
\end{aligned}
\tag{11}
$$

Let us now study the complexity of $\mathcal{F}_{cut}$. If the number of separating cuts in the graph is polynomial (in fact one should only consider those not included in larger cuts), then $\mathcal{F}_{cut}$ can still be solved using standard robust optimization techniques (See Section 7). However, we will show that solving $\mathcal{F}_{cut}$ is unfortunately NP-hard.

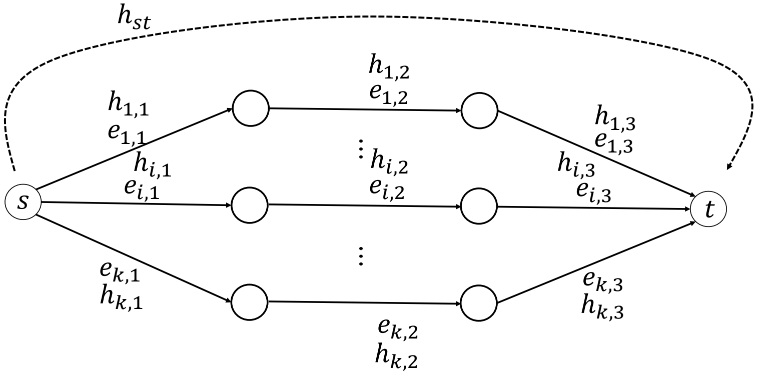

FIGURE 4 An example with $3k+1$ commodities: $h_{i,j}$ with $i = 1, ..., k, j = 1, 2, 3$ having the same source and sink as edges $e_{i,j}$, and h_{st} from source node s to sink node t. In this example we prove that it is NP-hard to solve $\mathcal{F}_{cut}$.

Proposition 4.2. *It is NP-hard to solve $\mathcal{F}_{cut}$.*

Proof We are going to propose a reduction from the 3-SAT problem. Let us consider a 3-SAT formula φ with k clauses and r variables. We denote by $\mathcal{L} = \{l_1, \ldots, l_r, \neg l_1, \ldots, \neg l_r\}$ the set of the literals appearing in formula φ and $l_{i,j}$ the literal appearing in the i-th clause C_i at the j-th position for $i = 1, ..., k$ and $j = 1, 2, 3$. We create a polyhedron Ξ by considering for each literal $l \in \mathcal{L}$ a non-negative variable ξ_l and for $p = 1, ..., r$, we add the constraint $\xi_{l_p} + \xi_{\neg l_p} = 1$.

We build as follows a graph G, a set of commodities $\mathcal{H}$ and a polyhedral uncertainty set $\mathcal{D}$. For each $i = 1, ..., k$, $j = 1, 2, 3$ we add 3 consecutive directed edges $e_{i,j}$ (see Figure 4) and 3 commodities $h_{i,j}$ with $s(h_{i,j}) = s(e_{i,j})$ and $t(h_{i,j}) = t(e_{i,j})$, and $d^{h_{i,j}} \leq \xi_{l_{i,j}}$. We impose that all nodes $s(e_{i,1})$ (resp. $t(e_{i,3})$) for $i = 1, ..., k$ are equal to a single node noted s (resp. t) (see Figure 4). We consider an additional commodity h_{st} from s to t whose value satisfies $d^{h_{st}} \leq 1$. The uncertainty polyhedron $\mathcal{D}$ is then obtained by projecting Ξ on the space of d^h variables. Finally, the capacity c_e of each edge e is here equal to 1 ($c_e = 1$).

Let us now prove that the optimal objective value of $\mathcal{F}_{cut}$ is $m = 1$ if and only if the 3-SAT formula φ is not satisfiable. If φ is satisfiable, then there is a demand vector (induced by the truth assignment) such that for each path between s and t (there is one path corresponding to each clause), at least one commodity whose endpoints are on the path is equal to 1 (a commodity corresponding to a true literal). This implies that all paths are blocked and thus $m > 1$ since one has to route commodity h_{st} through the network (since $\mathcal{F}_{cut}$ is more restrictive than $\mathcal{F}_{dyn}$).

If φ is not satisfiable, then for each extreme point of $\mathcal{D}$, there is at least one free path to route the demand $d^{h_{st}}$. In other words, each extreme demand vector of $\mathcal{D}$ can be routed through the network. Since each demand vector d inside $\mathcal{D}$ can be written as a convex combination of the extreme points of $\mathcal{D}$, d can also be routed through the network without requiring more than one unit of capacity on each edge. It is then clear that the solution defined by $m = 1$ and $f^{h_{st}}_{e_{i,j}}(d) = 1 - d^{h_{i,j}}$ is feasible for $\mathcal{F}_{cut}$. The optimal congestion is hence equal to 1. □

Notice that the separation problem related to the polyhedron $\mathcal{D}$ introduced in the proof above (i.e., given some vector d, check whether d belongs to $\mathcal{D}$ or provide a cut separating d from $\mathcal{D}$) can obviously be solved in polynomial time [30]. Otherwise the NP-hardness of solving $\mathcal{F}_{cut}$ would be a direct consequence of the difficulty of the separation problem related to $\mathcal{D}$.

We know from Proposition 4.1 that $\mathcal{F}_{cut}$ dominates $\mathcal{F}_{+}$ and $\mathcal{F}_{-}$. We just proved that $\mathcal{F}_{cut}$ is NP-hard to solve. We also recalled in Section 1 that the robust network design problem is NP-hard (i.e., $\mathcal{F}_{dyn}$ is NP-hard to solve). One can then wonder whether there is any difference between $\mathcal{F}_{cut}$ and the dynamic routing formulation $\mathcal{F}_{dyn}$. The following proposition answers this question.

Proposition 4.3. *$\mathcal{F}_{dyn}$ is less conservative than $\mathcal{F}_{cut}$.*

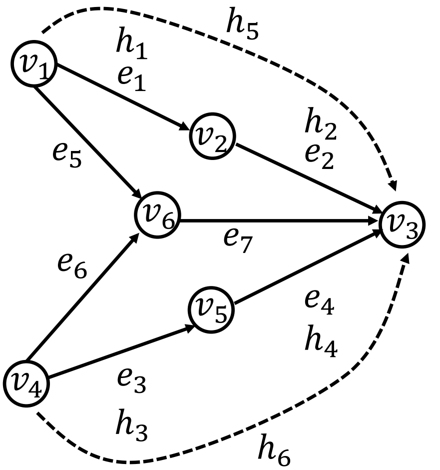

FIGURE 5 An example with six commodities: h_i with $i = 1, 2, 3, 4$ having the same source and sink as edges e_i, h_5 from source node v_1 to sink node v_3, and h_6 from source node v_4 to sink node v_3. In this example we prove that $\mathcal{F}_{dyn}$ is less conservative than $\mathcal{F}_{cut}$.

Proof Let us consider again the example of Figure 5 containing 6 vertices and the 7 directed edges: $e_1 = v_1v_2$, $e_2 = v_2v_3$, $e_3 = v_4v_5$, $e_4 = v_5v_3$, $e_6 = v_4v_6$, $e_5 = v_1v_6$ and $e_7 = v_6v_3$, of capacity 1 each. It also contains 6 commodities (see Figure 5): h_i with $i = 1, 2, 3, 4$ having the same source and sink as edges e_i, h_5 from source node v_1 to sink node v_3, and h_6 from source node v_4 to sink node v_3. The uncertainty set $\mathcal{D}$ is here defined by the constraints: $d^{h_1} + d^{h_3} \leq 1$, $d^{h_1} + d^{h_4} \leq 1$, $d^{h_2} + d^{h_3} \leq 1$, $d^{h_2} + d^{h_4} \leq 1$, $d^{h_5} \leq 1$ and $d^{h_6} \leq 1$ in addition to non-negativity constraints.

To show that the optimal congestion provided by $\mathcal{F}_{dyn}$ is equal to 1 we only have to prove that each extreme point of $\mathcal{D}$ can be routed without using more than 1 unit of capacity. It is clear that the more constraining scenarios are those where $d^{h_5} = d^{h_6} = 1$. We also either have $d^{h_1} = d^{h_2} = 1$ and $d^{h_3} = d^{h_4} = 0$ or $d^{h_1} = d^{h_2} = 0$ and $d^{h^3} = d^{h_4} = 1$. By symmetry, we can just focus on the first case ($d^{h_5} = d^{h_6} = 1$, $d^{h_1} = d^{h_2} = 1$ and $d^{h_3} = d^{h_4} = 0$) where one can clearly route commodity h_5 completely through the path containing e_5 and e_7 while h_6 is routed through e_3 and e_4. In other words, $m = 1$ for formulation (11).

Let us now prove by contradiction that the congestion obtained by $\mathcal{F}_{cut}$ is strictly greater than 1. Assume it to be equal to 1. Observe that when either $d^{h_1} = 1$ or $d^{h_2} = 1$, commodity h_5 is necessarily routed through e_7. This implies that $f_{e_7}^{h_5}(1_{h_1} + 1_{h_5} + 1_{h_6}) = f_{e_7}^{h_5}(1_{h_2} + 1_{h_5} + 1_{h_6}) = f_{e_7}^{h_5}(1_{h_1} + 1_{h_2} + 1_{h_5} + 1_{h_6}) = 1$, where 1_{h_i} denotes the demand vector where all demands are equal to 0 while $d^{h_i} = 1$. Since $f_{e_7}^{h_5}(d) = x_{e_7}^{h_5 0} + x_{e_7}^{h_5 h_1} d^{h_1} + x_{e_7}^{h_5 h_2} d^{h_2} + x_{e_7}^{h_5 h_3} d^{h_3} + x_{e_7}^{h_5 h_4} d^{h_4} +$

$x_{e_7}^{h_5h_5}d^{h_5} + x_{e_7}^{h_5h_6}d^{h_6}$, the previous equalities imply that $x_{e_7}^{h_5h_1} = 0$ and $x_{e_7}^{h_5h_2} = 0$. Observe that when either d^{h_3} or d^{h_4} is equal to 1 and demand $d^{h_6} = 1$ then commodity h_6 is fully routed through e_7 which requires that h_5 does not use e_7. Consequently, we have $f_{e_7}^{h_5}(1_{h_3} + 1_{h_5} + 1_{h_6}) = f_{e_7}^{h_5}(1_{h_4} + 1_{h_5} + 1_{h_6}) = f_{e_7}^{h_5}(1_{h_3} + 1_{h_4} + 1_{h_5} + 1_{h_6}) = 0$. These equalities lead to $x_{e_7}^{h_5h_3} = 0$ and $x_{e_7}^{h_5h_4} = 0$. From $f_{e_7}^{h_5}(1_{h_3} + 1_{h_4} + 1_{h_5} + 1_{h_6}) = 0$, we obtain $x_{e_7}^{h_5h_5} + x_{e_7}^{h_5h_6} = 0$, while $f_{e_7}^{h_5}(1_{h_1} + 1_{h_2} + 1_{h_5} + 1_{h_6}) = 1$ leads to the contradictory equality $x_{e_7}^{h_5h_5} + x_{e_7}^{h_5h_6} = 1$. □

5 | EXTENDED GRAPH FORMULATION

We have seen that formulations $\mathcal{F}_-$ and $\mathcal{F}_+$ can be strictly tighter than $\mathcal{F}_=$ (i.e., closer to $\mathcal{F}_{dyn}$). The difference between $\mathcal{F}_-$ and $\mathcal{F}_+$ lies in the sign of the terms $\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d)$ for $v \in V \setminus \{s(h), t(h)\}$ required to be negative for $\mathcal{F}_-$ and positive for $\mathcal{F}_+$. Our first attempt to improve both $\mathcal{F}_-$ and $\mathcal{F}_+$ led to formulation $\mathcal{F}_{cut}$. However, $\mathcal{F}_{cut}$ is generally NP-hard to solve. We would like to propose a stronger formulation that is still easy to solve, where the features of $\mathcal{F}_-$ and $\mathcal{F}_+$ are combined in some way.

We propose the following. For each commodity $h \in \mathcal{H}$, and for each node $v \in V \setminus \{s(h), t(h)\}$, we add to G the two directed edges $t(h)v$ and $vs(h)$. We also add an edge directed from $t(h)$ to $s(h)$. For each commodity h, an $s(h)t(h)$ flow f^h is considered in the extended graph. Notice that the extra edges we added $t(h)v$, $vs(h)$ and $t(h)s(h)$ can only be used by commodity h. Flow conservation constraints can be expressed as follows.

$$f_{vs(h)}^h(d) + \sum_{e\in\delta_+(v)} f_e^h(d) - f_{t(h)v}^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d) = 0 \text{ if } v \neq s(h), t(h) \tag{12}$$

$$\sum_{e\in\delta_+(s(h))} f_e^h(d) - \sum_{v\in V\setminus\{s(h)\}} f_{vs(h)}^h(d) = d^h. \tag{13}$$

Notice that $\delta_+(v)$ and $\delta_-(v)$ contain only edges belonging to G.

For sake of completeness, we give below the new formulation $\overline{\mathcal{F}}$. We use here $\overline{E}$ to denote $\{t(h)s(h)\} \cup E \cup \bigcup_{v\in V\setminus\{s(h),t(h)\}} \{vs(h), t(h)v\}$.

$$\min m$$

$$\sum_{e\in\delta_+(v)\cup\{vs(h)\}} \left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) - \sum_{e\in\delta_-(v)\cup\{t(h)v\}} \left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) = 0$$

$$\forall h \in \mathcal{H}, v \in V \setminus \{s(h), t(h)\}, d \in \mathcal{D} \tag{14a}$$

$$\sum_{e\in\delta_+(s(h))} \left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) - \sum_{v\in V\setminus\{s(h)\}} \left(x_{vs(h)}^{h0} + \sum_{h'\in\mathcal{H}} x_{vs(h)}^{hh'} d^{h'}\right) = d^h$$

$$\forall h \in \mathcal{H}, d \in \mathcal{D}$$

$$\sum_{h\in\mathcal{H}} \left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) \le c_e m, \quad \forall e \in E, d \in \mathcal{D} \tag{14b}$$

$$x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'} \ge 0, \quad \forall d \in \mathcal{D}, h \in \mathcal{H}, e \in \overline{E} \tag{14c}$$

Observe that there are no explicit capacity limitations for the edges not belonging to E (the added edges of type $vs(h)$ and $t(h)v$). However, positivity is required for the flow on these edges.

It is easy to see that $\mathcal{F}_+$ (resp. $\mathcal{F}_-$) is a special case of $\overline{\mathcal{F}}$ since the term $\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d)$ (resp. $\sum_{e\in\delta_-(v)} f_e^h(d) - \sum_{e\in\delta_+(v)} f_e^h(d)$) is positive in $\mathcal{F}_+$ (resp. $\mathcal{F}_-$) and can be seen as the flow going through an additional edge $t(h)v$ (resp. $vs(h)$). In other words, by considering only edges of type $t(h)v$ (resp. $vs(h)$) and solving $\overline{\mathcal{F}}$ we get $\mathcal{F}_+$ (resp. $\mathcal{F}_-$) .
Let us now prove that $\overline{\mathcal{F}}$ is a valid formulation.

Proposition 5.1. *$\overline{\mathcal{F}}$ is a valid formulation.*

Proof For each commodity $h \in \mathcal{H}$ and for each $d \in \mathcal{D}$ the solution induced by $\overline{\mathcal{F}}$ is a $s(h)t(h)$ flow in the extended graph of value d^h. Consequently, each cut of the extended graph that separates $s(h)$ and $t(h)$ has necessarily a capacity greater than or equal to d^h. Observe however that additional edges of type $t(h)v$, $vs(h)$ and $t(h)s(h)$ do not belong to any separating cut. This means that any separating cut in the extended graph contains only edges from the original graph. We can thus deduce that any separating cut in G has a capacity greater than or equal to d^h. By the maximum-flow minimum-cut theorem, it is then possible to route commodity h using the capacities $f_e^h(d)$ on the edges of G. □

We give below an example showing that $\overline{\mathcal{F}}$ dominates both formulations $\mathcal{F}_-$ and $\mathcal{F}_+$.

Proposition 5.2. *$\overline{\mathcal{F}}$ is less conservative than $\mathcal{F}_-$ and $\mathcal{F}_+$.*

Proof Consider a feasible solution of, for example, $\mathcal{F}_+$. Then it satisfies $\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d) \geq 0$. Let us then extend it to a feasible solution of $\overline{\mathcal{F}}$ by setting $f_{t(h)v}^h = \sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d)$ and $f_{vs(h)}^h = 0$. We also define $f_{t(h)s(h)}^h$ by $\sum_{e\in\delta_+(s(h))} f_e^h(d) - \sum_{e\in\delta_-(s(h)} f_e^h(d) - d^h$. This solution is feasible for $\overline{\mathcal{F}}$ and has the same congestion as the feasible solution of $\mathcal{F}_+$ that we started with.

Let us consider again the example of Figure 3. We have seen that the congestion obtained by both $\mathcal{F}_-$ and $\mathcal{F}_+$ is equal to $\frac{4}{3}$. We give here a feasible solution of $\overline{\mathcal{F}}$ for which the congestion is only 1.25. We only have to determine the assignments related to h_5. Consider the solution defined by:

- $f_{e_1}^{h_5}(d) = f_{e_1'}^{h_5}(d) = 0.5 + 0.25d^{h_5} - 0.5d^{h_1}$,
- $f_{e_2}^{h_5}(d) = f_{e_2'}^{h_5}(d) = 0.25d^{h_5} + 0.5d^{h_1}$, $f_{e_3}^{h_5}(d) = 0.25 + 0.75d^{h_5}$,
- $f_{e_4}^{h_5}(d) = 0.5 + 0.5d^{h_5} - 0.25d^{h_4}$, $f_{v_6v_5}^{h_5}(d) = 0.25 - 0.25d^{h_5} - 0.25d^{h_4}$,
- $f_{v_4v_5}^{h_5}(d) = 0.25d^{h_4}$, $f_{v_6v_1}^{h_5} = 0.25 - 0.25d^{h_5}$,

while $f_e^{h_5}(d) = 0$ for all other edges. Observe that flow conservation constraints are satisfied in the extended graph. The three last edges mentioned above (i.e., v_6v_5, v_4v_5 and v_6v_1) do not belong to G (they are of type $vs(h)$ and $t(h)v$). The fact that they appear in the solution means that $\sum_{e\in\delta_+(v)} f_e^h(d) - \sum_{e\in\delta_-(v)} f_e^h(d)$ will be always positive for $v = v_1$ and negative for $v = v_4$ (this is to say that this solution is feasible neither for $\mathcal{F}_-$ nor for $\mathcal{F}_+$). Observe also that the total capacity used on e_1 is given by $f_{e_1}^{h_5}(d) + d^{h_1} = 0.5 + 0.25d^{h_5} + 0.5d^{h_1} \leq 1.25$. Similarly, the capacity used on e_2 is equal to $0.25d^{h_5} + 0.5d^{h_1} + d^{h_2} = 0.25d^{h_5} + 0.5(d^{h_1} + d^{h_2}) + 0.5d^{h_2} \leq 1.25$ (since $d^{h_1} + d^{h_2} \leq 1$). One can check that the same holds for edges e_1' and e_2'. The positivity of the terms $f_e^{h_5}(d)$ is also easy to check for each edge e using the definition of $\mathcal{D}$. □

Since $\mathcal{F}_{cut}$ is the best formulation that one can get when $f_e^h(d)$ is assumed to be affine and is generally NP-hard to solve while $\overline{\mathcal{F}}$ can be solved in polynomial time, there are cases where $\overline{\mathcal{F}}$ leads to higher-cost solutions than those obtained from $\mathcal{F}_{cut}$. This is shown by the example used above (Figure 3) for which we already proved in Proposition 4.1 that $\mathcal{F}_{cut}$ leads to a congestion equal to 1 while $\overline{\mathcal{F}}$ gives a congestion equal to 1.25.

To close this section, we would like to add that formulations (3) and (6) that were proposed as an alternative to formulation (1) (i.e., $\mathcal{F}_=$), can also be expressed in the context of the extended graph. To write the cycle-based formulation (6), we only have to take into account the fact that the set of cycles will here depend on the commodities (since for each commodity h we added some edges that can only be used by this commodity).

6 | AGGREGATION

One standard way to solve classical linear multi-commodity problems in a more efficient way consists in aggregating commodities either by source or by sink [1]. Let us then try to do the same in the context of polyhedral uncertainty and affine decision rules.

6.1 | Aggregation for $\mathcal{F}_=$

Let $S \subset V$ and $T \subset V$ be two subsets such that for each $h \in \mathcal{H}$ we either have $s(h) \in S$ or $t(h) \in T$. All commodities having s as a source ($s \in S$) will be aggregated and considered as one commodity having a source s and several sinks. Similarly, all commodities having t as a sink ($t \in T$) are aggregated into one commodity having several sources and one sink t. It may happen that $s(h) \in S$ and $t(h) \in T$ simultaneously occur; then we arbitrarily decide whether h is aggregated by source or by sink. For each $s \in S$ (resp. $t \in T$), let us use $\mathcal{H}_s$ (resp. $\mathcal{H}^t$) to denote the set of commodities having s (resp. t) as a source (resp. sink) and aggregated by source (resp. sink). For any $h \in \mathcal{H}$, it will be more convenient here to use $d^{h_{s(h)t(h)}}$ to denote the demand value of the commodity (there is no ambiguity since we can assume that there are no demands having exactly the same source and the same sink).

We also define for each $s \in S$ the set $T(s) = \{v : h \in \mathcal{H}_s, v = t(h)\}$ to denote the set of vertices v such that there is a commodity aggregated by source s and having v as a sink. Similarly, for $t \in T$, let $S(t) = \{v : h \in \mathcal{H}^t, v = s(h)\}$.

Applying this aggregation for $\mathcal{F}_=$ leads to the following aggregated formulation $\mathcal{F}_{agg}$.

$$\min m$$

$$\sum_{e\in\delta_+(v)} \left(x_e^{s0} + \sum_{h'\in\mathcal{H}} x_e^{sh'} d^{h'} \right) - \sum_{e\in\delta_-(v)} \left(x_e^{s0} + \sum_{h'\in\mathcal{H}} x_e^{sh'} d^{h'} \right) = \begin{cases} -d^{h_{sv}} & \text{if } v \in T(s) \\ 0 & \text{otherwise} \end{cases}$$

$$\forall s \in S, v \in V \setminus \{s\}, d \in \mathcal{D} \tag{15a}$$

$$\sum_{e\in\delta_+(v)} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) - \sum_{e\in\delta_-(v)} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) = \begin{cases} d^{h_{vt}} & \text{if } v \in S(t) \\ 0 & \text{otherwise} \end{cases}$$

$$\forall t \in T, v \in V \setminus \{t\}, d \in \mathcal{D} \tag{15b}$$

$$\sum_{s\in S} \left(x_e^{s0} + \sum_{h'\in\mathcal{H}} x_e^{sh'} d^{h'} \right) + \sum_{t\in T} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) \le c_e m, \quad \forall e \in E, d \in \mathcal{D} \tag{15c}$$

$$x_e^{s0} + \sum_{h'\in\mathcal{H}} x_e^{sh'} d^{h'} \ge 0, \quad \forall e \in E, s \in S, d \in \mathcal{D} \tag{15d}$$

$$x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \ge 0, \quad \forall e \in E, t \in T, d \in \mathcal{D} \tag{15e}$$

Observe that all variables (except m) are either indexed by a source s or a sink t. The number of variables is almost proportional to $|S|+|T|$. Then to minimize the number of variables, one has to minimize $|S|+|T|$ which can obviously be done by computing a minimum vertex cover in a bipartite graph (the demand graph) and is equal to the cardinality of a maximum matching (by Konig's theorem) [31].

The validity of $\mathcal{F}_{agg}$ is a direct consequence of the validity of aggregation when there is no uncertainty.

It is also obvious that any solution of $\mathcal{F}_=$ can be used to build a solution for $\mathcal{F}_{agg}$ having the same congestion. For each $s \in S$ (resp. $t \in T$), we only have to sum the variables related to commodities belonging to $\mathcal{H}_s$ (resp. $\mathcal{H}^t$) to get those related to s (resp. t).

We know that when there is no uncertainty, $\mathcal{F}_=$ and $\mathcal{F}_{agg}$ are equivalent. One may wonder whether they are equivalent when polyhedral uncertainty is considered. The next proposition states that $\mathcal{F}_{agg}$ dominates $\mathcal{F}_=$.

Proposition 6.1. *Formulation $\mathcal{F}_{agg}$ is less conservative than formulation $\mathcal{F}_=$.*

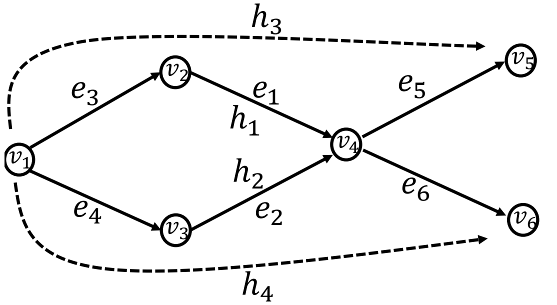

FIGURE 6 An example with four commodities: h_i with $i = 1, 2$ having the same source and sink as edges e_i, h_3 from source node v_1 to sink node v_5, and h_4 from source node v_1 to sink node v_6. In this example we prove that formulation $\mathcal{F}_{agg}$ is less conservative than formulation $\mathcal{F}_=$.

Proof Let us consider the graph of Figure 6 containing 6 edges of capacity 1 each. There are 4 commodities: h_i with $i = 1, 2$ having the same source and sink as edges e_i, h_3 from source node v_1 to sink node v_5, and h_4 from source node v_1 to sink node v_6.

The polyhedron $\mathcal{D}$ is defined as the set of demands $d \in \mathbb{R}^4_+$ satisfying the two equations $d^{h_1} + d^{h_2} = 1$ and $d^{h_3} + d^{h_4} = 1$. Due to the equalities defining $\mathcal{D}$, we can assume without generality loss that there is affine dependence on only d^{h_1} and d^{h_3}.

First, let us consider formulation $\mathcal{F}_{agg}$ with only source aggregation. We will then aggregate commodities h_3 and h_4. Consider the solution of $\mathcal{F}_{agg}$ defined by: $f^{v_1}_{e_1}(d) = f^{v_1}_{e_3}(d) = 1 - d^{h_1}$, $f^{v_1}_{e_2}(d) = f^{v_1}_{e_4}(d) = d^{h_1}$, $f^{v_1}_{e_5}(d) = d^{h_3}$ and $f^{v_1}_{e_6}(d) = 1 - d^{h_3}$. Variables related to the two other sources v_2 and v_3 are fixed in an obvious way. This solution allows a congestion equal to 1.

Let us now assume that there is a feasible solution of $\mathcal{F}_=$ with congestion equal to 1. Let f be such a solution. For each edge e, we have $f^{h_3}_e = x^{h_3 0}_e + x^{h_3 h_1}_e d^{h_1} + x^{h_3 h_3}_e d^{h_3}$. Since the graph is acyclic and f^{h_3} is a positive flow, if the demand for a commodity h_3 is zero then the flow for this commodity must also be zero in model (1). Then when d is the demand vector where $d^{h_3} = d^{h_1} = 0$, we should have $f^{h_3}_e(d) = 0$ implying that $x^{h_3 0}_e = 0$ for each edge e. Similarly, when d is such that $d^{h_3} = 0$ and $d^{h_1} = 1$ we also have $f^{h_3}_e(d) = 0$ leading to $x^{h_3 h_1}_e = 0$.

Let us now focus on edge e_1. When $d^{h_1} = d^{h_3} = 1$, e_1 already carries commodity h_1 whose value is here equal to 1. Then there are no more resources that can be used by commodity h_3 implying that $f_{e_1}^{h_3}(d) \leq 0$. Using the positivity constraint we can deduce that $f_{e_1}^{h_3}(d) = 0$ when $d^{h_1} = d^{h_3} = 1$. Thus, $x_{e_1}^{h_3 h_3} = 0$. In other words, $f_{e_1}^{h_3}(d) = 0$ for any d implying that commodity h_3 is never routed through e_1. It is then fully routed through e_2. This is of course not possible without violating the capacity constraint of e_2 since commodity h_2 is already routed through e_2. □

According to Proposition 6.1 it should be understood that aggregation is not only interesting for accelerating problem solving (as is the case for problems without uncertainty), but it also leads to better solutions since we are getting closer to $\mathcal{F}_{dyn}$. In fact, by aggregating commodities and solving $\mathcal{F}_{agg}$, the capacities reserved for each aggregated commodity is affine while the capacities used by each individual commodity making up the aggregated one are not necessarily affine in d.

6.2 | Sink aggregation for $\mathcal{F}_+$

Since $\mathcal{F}_+$ dominates $\mathcal{F}_=$, it would be interesting to perform some kind of aggregation to be able to solve larger problems and further reduce congestion.

We consider the aggregated formulation $\mathcal{F}_{agg+}$ given below where only sink aggregation is possible (so $S = \emptyset$). Observe also that equality constraints (15b) are replaced by inequalities (16a).

$$\min m$$

$$\sum_{e\in\delta_+(v)} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) - \sum_{e\in\delta_-(v)} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) \geq \begin{cases} d^{h_{vt}} & \text{if } v \in S(t) \\ 0 & \text{otherwise} \end{cases}$$

$$\forall t \in T, v \in V \setminus \{t\}, d \in \mathcal{D} \tag{16a}$$

$$\sum_{t\in T} \left(x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \right) \leq c_e m, \quad \forall e \in E, d \in \mathcal{D} \tag{16b}$$

$$x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'} \geq 0, \quad \forall e \in E, t \in T, d \in \mathcal{D} \tag{16c}$$

The validity of $\mathcal{F}_{agg+}$ is less obvious than the validity of $\mathcal{F}_{agg}$, and it is demonstrated hereafter.

Proposition 6.2. *$\mathcal{F}_{agg+}$ is a valid formulation.*

Proof Consider any feasible solution f of $\mathcal{F}_{agg+}$. Let us select any traffic vector $d \in \mathcal{D}$ and any sink $t \in T$. For each node $v \in V \setminus \{t\}$ we add to G the edge tv of infinite capacity. Let us also add to G a "virtual" node s_t and an edge from s_t to each vertex $v \in S(t)$ of capacity $d^{h_{vt}}$. Then, starting from $f_e^t(d) = x_e^{t0} + \sum_{h'\in\mathcal{H}} x_e^{th'} d^{h'}$ for each edge $e \in E$, f^t can be extended to a positive flow from s_t to t by taking

$$f_{tv}^t(d) = \begin{cases} \sum_{e\in\delta_+(v)} f_e^t(d) - \sum_{e\in\delta_-(v)} f_e^t(d) & \text{if } v \in V \setminus S(t) \\ \sum_{e\in\delta_+(v)} f_e^t(d) - \sum_{e\in\delta_-(v)} f_e^t(d) - d^{h_{vt}} & \text{if } v \in S(t) \end{cases}$$

and $f_{s_t v}^t(d) = d^{h_{vt}}$ for $v \in S(t)$. We are then sending a flow of value $\sum_{v\in S(t)} d^{h_{vt}}$ from s_t to t. Directed cycles can be cancelled in a standard way by decreasing flow on the edges of each directed cycle. We can therefore assume that the set of edges for which $f_e^t > 0$ does not contain any directed cycle. Since the flow on the "virtual" edges $s_t v$ is exactly equal to $d^{h_{vt}}$, the $s_t t$-flow induces $|S(t)|$ simultaneous positive flows, each from a vertex $v \in S(t)$ to t and of value exactly equal to $d^{h_{vt}}$. Each vt-flow ($v \in S(t)$) uses only original edges of G. This clearly implies that it is possible to simultaneously route the demand $d^{h_{vt}}$ for each $v \in S(t)$. Since this holds for any $t \in T$ and any $d \in \mathcal{D}$, the validity of the formulation is proved. □

One can easily modify the example of Figure 6 to show that $\mathcal{F}_{agg+}$ dominates $\mathcal{F}_{+}$. It is also easy to see that $\mathcal{F}_{agg+}$ dominates $\mathcal{F}_{agg}$ when only sink aggregation is considered to build $\mathcal{F}_{agg}$ (i.e., when $|S| = 0$).

Finally, we should mention that source aggregation cannot be used in combination with $\mathcal{F}_{+}$ even if there is no uncertainty. Consider, for example, a graph having four vertices, s, v, t_1 and t_2 and two edges st_1 and vt_2 having some capacity. Assume that we have two commodities h_1 and h_2 of value 1 each from s to t_1 and from s to t_2. Observe that there is even no path from s to t_2 so the network design problem has no solution. However, by taking $f^s_{vt_2} = 1$ and $f^s_{st_1} = 2$, we can ensure that all constraints related to the aggregated commodity will be satisfied (the flow entering t_1 is greater than 1, the flow going out of v is greater than what is going into v, the flow reaching t_2 is greater than 1, and we even have that what comes out of s is greater than the sum of the two demands).

6.3 | Source aggregation for $\mathcal{F}_{-}$

Aggregation can also be considered in combination with $\mathcal{F}_{-}$. However, only source aggregation can be used. The obtained formulation denoted by $\mathcal{F}_{agg-}$ would be the following.

$$\min m$$

$$\sum_{e \in \delta_+(v)} \left(x_e^{s0} + \sum_{h' \in \mathcal{H}} x_e^{sh'} d^{h'} \right) - \sum_{e \in \delta_-(v)} \left(x_e^{s0} + \sum_{h' \in \mathcal{H}} x_e^{sh'} d^{h'} \right) \leq \begin{cases} -d^{h_{sv}} & \text{if } v \in T(s) \\ 0 & \text{otherwise} \end{cases}$$
$$\forall s \in S, v \in V \setminus \{s\}, d \in \mathcal{D} \tag{17a}$$

$$\sum_{s \in S} \left(x_e^{s0} + \sum_{h' \in \mathcal{H}} x_e^{sh'} d^{h'} \right) \leq c_e m, \quad \forall e \in E, d \in \mathcal{D} \tag{17b}$$

$$x_e^{s0} + \sum_{h' \in \mathcal{H}} x_e^{sh'} d^{h'} \geq 0, \quad \forall e \in E, s \in S, d \in \mathcal{D} \tag{17c}$$

The proof of validity of $\mathcal{F}_{agg-}$ is very similar to the proof of validity of $\mathcal{F}_{agg+}$. One can also build examples showing that $\mathcal{F}_{agg-}$ dominates $\mathcal{F}_{-}$. It is also easy to see that $\mathcal{F}_{agg-}$ dominates $\mathcal{F}_{agg}$ when only source aggregation is considered to build $\mathcal{F}_{agg}$ (i.e., when $|T| = 0$).

Finally, we would like to mention that aggregation can also be considered in the context of formulation $\mathcal{F}_{cut}$. However, since solving $\mathcal{F}_{cut}$ is NP-hard and aggregation would not change the theoretical complexity, we are not going to study this kind of aggregation.

7 | SOLUTION METHODS AND NUMERICAL EVALUATION

Figure 7 summarizes the main domination relations between the models introduced or recalled in the paper. Notice that we assume here that $\mathcal{F}_{path}$ can potentially contain all possible paths (this is to say that $\mathcal{F}_{path}$ dominates $\mathcal{F}_{stat}$). However, a numerical evaluation is needed to quantify the difference in terms of performance between these variants.

In this section, we begin by presenting the two types of uncertainty sets considered in the evaluations. For the sake of completeness, we briefly recall in Section 7.2 standard duality-based methods to solve the introduced formulations. Data instances considered for evaluation are described in Section 7.3 and, finally, we present all the results in Section 7.4.

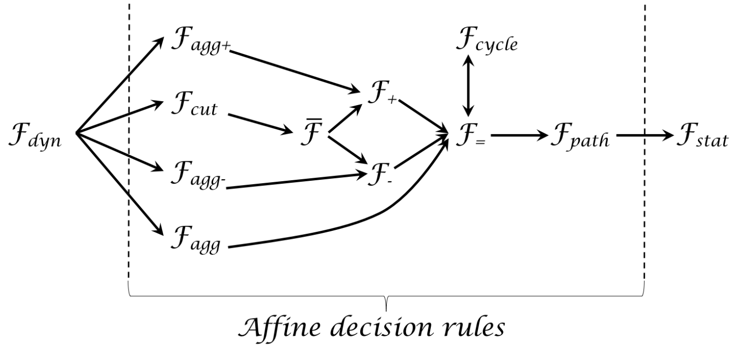

FIGURE 7 Domination relations between the models introduced or recalled in the paper.

7.1 | Uncertainty sets

For the numerical evaluation, we consider two different uncertainty sets. We first use the *Budget* uncertainty set [19] that considers a maximum deviation for each nominal demand with a global budget for possible variations. Second, we consider the *All Routable Demands* uncertainty set [6] which contains all demand vectors that can be routed through a given network where capacities are fixed and routing can be adapted to each demand vector. More formally, they are defined as follows:

- The *Budget uncertainty* set $\mathcal{D}$ is such that:

$$\mathcal{D} = \{d \in \mathbb{R}^{\mathcal{H}} : d^h = \overline{d}^h + z^h \hat{d}^h, \sum_{h \in \mathcal{H}} z^h \leq \Gamma, 0 \leq z^h \leq 1, \forall h \in \mathcal{H}\} \tag{18}$$

  where $\overline{d}^h$ is the nominal demand for commodity h, $\hat{d}^h$ is the maximum possible deviation from $\overline{d}^h$, and Γ is a parameter that specifies a limit (the budget) on the deviations of all demands at the same time with respect to the nominal values.
- The *All Routable Demands* uncertainty set $\mathcal{D}$ on a given capacity vector is formally the set of $d \in \mathbb{R}^{\mathcal{H}}$ such that there is a multi-commodity flow $f(d) \in \mathbb{R}^{\mathcal{H} \times E}$ satisfying:

$$\sum_{e \in \delta_+(v)} f_e^h(d) - \sum_{e \in \delta_-(v)} f_e^h(d) = \begin{cases} d^h, & \text{if } v = s(h) \\ -d^h, & \text{if } v = t(h) \\ 0 & \text{otherwise} \end{cases} \qquad \forall v \in V, h \in \mathcal{H} \tag{19}$$

$$\sum_{h \in \mathcal{H}} f_e^h(d) \leq c_e \qquad \forall e \in E$$

The two uncertainty sets introduced above are easy to handle (i.e., the separation problem related to each set can be solved in polynomial time).

7.2 | Solution methods

The models introduced in this paper involve constraints that must be satisfied for all traffic vectors $d \in \mathcal{D}$. When $\mathcal{D}$ is a polytope having a polynomial number of extreme points, some formulations such as $\mathcal{F}_{dyn}$ can be solved by considering the constraints related to each extreme point. However, for most of polytopes considered in the literature (such as those described above), the number of extreme points is not polynomial. Then there are mainly two methods to handle constraints involving d: either cutting-plane algorithms where traffic vectors are generated in an iterative way [8] or duality-based approaches [14]. While the cutting-plane approach can be applied for any tractable polytope (i.e., for which separation is polynomial), the second approach is recommended when the polyhedral set can be described using a limited number of variables and constraints [11, 14].

We are then going to use duality-based approaches to solve the problems introduced in the paper. Duality allows us to obtain equivalent compact linear programs of the original problems [14]. In the following we will describe, as an example, how this is done for model (3) and the *Budget* uncertainty set $\mathcal{D}$ described in (18). The same method can be (quite) straightforwardly applied to the other models. The numerical results that will be presented later in the section are obtained using this method.

For each edge e, constraint (3b) (recall that this latter is given as: $\sum_{h\in\mathcal{H}}\left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) \le c_e m$) is satisfied for all traffic vectors $d \in \mathcal{D}$ if and only if the constraint $\max_{d\in\mathcal{D}} \sum_{h\in\mathcal{H}}\left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'} d^{h'}\right) \le c_e m$ is satisfied. Thus by writing the polyhedron $\mathcal{D}$ in a more explicit form we obtain that a given solution (x, m) of (3) satisfies this constraint if and only if the solution of the following linear program gives a capacity reservation/congestion value that is less than $c_e m$.

$$\max_{z} \sum_{h\in\mathcal{H}}\left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'}(\overline{d}^{h'} + z^{h'}\hat{d}^{h'})\right)$$

$$\sum_{h'\in\mathcal{H}} z^{h'} \le \Gamma \tag{20a}$$

$$0 \le z^{h'} \le 1 \qquad \forall h' \in \mathcal{H} \tag{20b}$$

By linear programming duality theory, model (20) has an optimal solution of value less than $c_e m$ if and only if the following dual linear program has a feasible solution of value less than $c_e m$.

$$\min_{\pi,\mu} \sum_{h\in\mathcal{H}}\left(x_e^{h0} + \sum_{h'\in\mathcal{H}} x_e^{hh'}\overline{d}^{h'}\right) + \Gamma\pi_e + \sum_{h'\in\mathcal{H}} \mu_e^{h'}$$

$$\pi_e + \mu_e^{h'} \ge \hat{d}^{h'} \sum_{h\in\mathcal{H}} x_e^{hh'} \qquad \forall h' \in \mathcal{H}$$

$$\mu_e^{h'} \ge 0, \pi_e \ge 0 \qquad \forall h' \in \mathcal{H}$$

where π_e and $\mu_e^{h'}$ are the dual variables corresponding to constraints (20a) and (20b), respectively.

We can thus replace each constraint (3b) by the following inequalities.

$$\sum_{h\in\mathcal{H}}\left(x_e^{h0}+\sum_{h'\in\mathcal{H}}x_e^{hh'}\overline{d}^{h'}\right)+\Gamma\pi_e+\sum_{h'\in\mathcal{H}}\mu_e^{h'}\leq c_e m$$

$$\pi_e+\mu_e^{h'}\geq\hat{d}^{h'}\sum_{h\in\mathcal{H}}x_e^{hh'}\qquad\forall h'\in\mathcal{H}$$

$$\mu_e^{h'}\geq 0,\pi_e\geq 0\qquad\forall h'\in\mathcal{H}$$

Constraints (3c) can also be dualized in a very similar way.

Finally, we should mention that in formulations $\mathcal{F}_=$, $\mathcal{F}_+$, $\mathcal{F}_-$, $\mathcal{F}_{agg+}$, $\mathcal{F}_{agg-}$, $\mathcal{F}_{agg}$ and $\overline{\mathcal{F}}$ the flow conservation constraints are handled as done in (3). For example, to solve $\mathcal{F}_+$, by adding virtual edges of type $t(h)v$, we recover again equalities in the flow conservation constraints that should be satisfied for each $d\in\mathcal{D}$. These equalities are then replaced by a set of equalities that is similar to (3a).

The static routing approach (noted as $\mathcal{F}_{stat}$ in the result tables) consists in choosing a fixed routing for all demand scenarios (i.e., $f_e^h(d)=x_e^{h0}$ for any $d\in\mathcal{D}$). It is described in [6, 8]. Here we solve this problem with the same duality-based techniques.

7.3 | Network instances

We consider Abilene and Geant, two publicly available directed network topologies taken from the SNDlib [39] library and commonly used in the networking community for numerical evaluations. The former is of medium size (12 nodes and 30 links) while the latter is of larger size (24 nodes and 72 links). The arc capacities are those present in the SNDlib instances.

We compare the affine routing and static routing formulations considering the minimization of two classical objective functions: 1) the congestion m denoted as *Congestion* and 2) a linear reservation cost (denoted as *Linear*). The congestion m is expressed as $\max_{e\in E}\frac{u_e}{c_e}$ where c_e is the capacity of edge e and u_e is the reserved capacity on such edge, while the *Linear* objective is expressed as $\sum_{e\in E}\lambda_e u_e$ where λ_e are scalars corresponding to the unit cost of underlying resources. In practice, physical links are generally composed of multiples of standard capacity values. Unfortunately, this leads to NP-hard network design problems. Therefore, the linear objective function is often a suitable alternative to approximate more complex objective functions.

To generate different sets of commodities on each instance with an increasing number of demands, we begin with the set $\mathcal{H}_0$ consisting of all the possible commodities between sources and destinations. We generate a subset $\mathcal{H}_1\subseteq\mathcal{H}_0$ by selecting commodities from $\mathcal{H}_0$ with a uniform probability distribution. We re-iterate this process until $\mathcal{H}_1$ is of the size we desire. Next, we build a subset $\mathcal{H}_2$ of $\mathcal{H}_1$ with the same procedure to get a smaller set of demands. Successively, we obtain a sequence of demand sets of decreasing size that are successively included in each other.

For each topology and each objective function we also consider the two uncertainty sets, *Budget* and *All Routable Demands*. The parameters for the *Budget* polyhedron are set as follow: $\overline{d}^h=\hat{d}^h=\min_{e\in E}c_e$ for all commodity $h\in\mathcal{H}$ and $\Gamma=\sqrt{card(\mathcal{H})}$.

7.4 | Numerical results

We first analyze the solutions from the different formulations. Then, we compare solution times and model sizes.

We compare static routing (denoted as $\mathcal{F}_{stat}$) with affine routing formulations that can be categorized into 6 groups: (1) the node-arc formulation $\mathcal{F}_=$, (2) those based on the relaxation of flow conservation constraints ($\mathcal{F}_-$ and $\mathcal{F}_+$) of affine routing ($\mathcal{F}_=$), (3) the one based on aggregation ($\mathcal{F}_{agg}$), (4) those using a mix of the two former ($\mathcal{F}_{agg-}$ and $\mathcal{F}_{agg+}$), (5) the one using the elementary cycle formulation ($\mathcal{F}_{cycle}$), and (6) the one using the extended graph formulation ($\overline{\mathcal{F}}$).

In our implementation, we used the CPLEX solver version 12.6.3 on servers having four Intel Xeon E5-4627 v2 3.3 GHz CPU cores and 512GB of memory. In all our computations, CPLEX is configured without a time limit and the default optimality gap. We used Julia to model problems and interface with the solver.

7.4.1 | Comparison of objective values

We present two series of tables, Tables 1 and 2, with the solution of all formulations on Abilene and Geant, respectively. For each topology, we consider the two polyhedra and the two objective functions described above.

For each case, the table is organized as follows. The first row gives the number of demands (or commodities) $|K|$ ranging from 10 to 30 demands for the instances related to the *Budget* polyhedron (18), and from 5 to 15 for the more computationally expensive *All Routable Demands* polyhedron (19), except for Geant and the congestion objective (Table 2d) where the instances with 15 demands become prohibitively expensive to compute. The subsequent rows report the value of the objective function at the optimum for all affine routing variants presented in this paper: $\mathcal{F}_=$, $\mathcal{F}_-$, $\mathcal{F}_+$, $\mathcal{F}_{agg}$, $\mathcal{F}_{agg-}$, $\mathcal{F}_{agg+}$, $\mathcal{F}_{cycle}$, and $\overline{\mathcal{F}}$. We also give the cost of the static routing solution $\mathcal{F}_{stat}$. The last two rows contain the gap between the static routing formulation $\mathcal{F}_{stat}$ (resp. original affine routing formulation $\mathcal{F}_=$) and the extended flow formulation $\overline{\mathcal{F}}$. We denote by $BG_{\mathcal{F}_{stat}}$ (resp. $BG_{\mathcal{F}_=}$) those gaps and we compute them as $\frac{O_{\mathcal{F}_{stat}} - O_{\overline{\mathcal{F}}}}{O_{\mathcal{F}_{stat}}}$ (resp. $\frac{O_{\mathcal{F}_=} - O_{\overline{\mathcal{F}}}}{O_{\mathcal{F}_=}}$) where $O_{\mathcal{F}_{stat}}$, $O_{\mathcal{F}_=}$ and $O_{\overline{\mathcal{F}}}$ are, respectively, the cost of the solutions of $\mathcal{F}_{stat}$, $\mathcal{F}_=$ and $\overline{\mathcal{F}}$.

First of all, as expected, we can see that, in almost all the tables, all the variants of affine routing exhibit better solutions compared to the static routing, especially when the number of demands is large (this was also observed in [44]). Also, the solution given by $\mathcal{F}_=$ is, on one hand, almost always strictly dominated by the solution obtained by $\overline{\mathcal{F}}$. The solution given by $\overline{\mathcal{F}}$ seems to be the best one with respect to all the other variants of affine routing and static routing in all the considered scenarios (e.g., see Tables 2a and 2c at the 30 commodities column). We also observe that the solution of $\mathcal{F}_+$, and $\mathcal{F}_-$ can give strictly better solutions than $\mathcal{F}_=$ (e.g., see Table 2d). Furthermore, observe that the solutions of $\mathcal{F}_{agg}$, $\mathcal{F}_{agg+}$, and $\mathcal{F}_{agg-}$ can give slightly strictly better solutions than $\mathcal{F}_=$, $\mathcal{F}_+$, and $\mathcal{F}_-$ respectively (see, for example, Table 2a with 30 commodities for $\mathcal{F}_-$ and $\mathcal{F}_+$, and Table 2c with 30 commodities for $\mathcal{F}_=$).

Let us now look more closely at solutions from the different formulations and in particular compare $\mathcal{F}_=$, the original affine formulation, and $\overline{\mathcal{F}}$. For instance, for Abilene with *Congestion* and *All Routable Demands* we obtain a percentage gap up to 9.914 % between $\mathcal{F}_=$ and $\overline{\mathcal{F}}$, and up to 10.538 % between $\mathcal{F}_{stat}$ and $\overline{\mathcal{F}}$. Similarly, for Geant, we have the same trend, with slightly lower percentage gaps (up to 4.458 % and 5.259 %).

7.4.2 | Comparison of model sizes and solution times

We now present two series of tables displaying the solution times and the model sizes. In the first series (Table 3) we compare the two polyhedra on the Abilene topology with the *Congestion* objective. And in the second series (Table 4) we compare both topologies with the *All Routable Demands* polyhedron and the *Linear* objective. The solution times are in seconds. For the size of models, we display for each formulation the number of columns (i.e., variables), denoted as #col, and the number of rows (i.e., constraints), denoted as #row.

We can observe that, in general, the computation time for the scenarios with *All Routable Demands* (Table 3b) can be several hundred times longer than with *Budget* (Table 3a). This can be explained by the fact that the *All Routable*

TABLE 1 Solutions for scenarios with Abilene topology.

(a) *Budget*.
Linear objective.

	Nb demands				
	10	15	20	25	30
$\mathcal{F}_{stat}$	44.0	69.87	90.47	115.0	141.91
$\mathcal{F}_{=}$	44.0	69.18	89.68	112.0	134.57
$\mathcal{F}_{cycle}$	44.0	69.18	89.68	112.0	134.57
$\mathcal{F}_{+}$	44.0	69.08	89.68	112.0	134.57
$\mathcal{F}_{-}$	44.0	68.93	89.68	112.0	134.57
$\overline{\mathcal{F}}$	44.0	67.63	87.97	108.0	129.82
$\mathcal{F}_{agg}$	44.0	69.18	89.68	112.0	134.57
$\mathcal{F}_{agg+}$	44.0	69.08	89.60	112.0	134.57
$\mathcal{F}_{agg-}$	44.0	68.93	89.68	112.0	134.57
$BG_{\mathcal{F}_{=}}$	0.0 %	2.2 %	1.9 %	3.6 %	3.5 %
$BG_{\mathcal{F}_{stat}}$	0.0 %	3.2 %	2.8 %	6.1 %	8.5 %

(b) *All Routable Demands*.
Linear objective.

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	21.0	30.5	35.5
$\mathcal{F}_{=}$	21.0	30.5	35.5
$\mathcal{F}_{cycle}$	21.0	30.5	35.5
$\mathcal{F}_{+}$	21.0	30.5	35.5
$\mathcal{F}_{-}$	21.0	30.5	35.5
$\overline{\mathcal{F}}$	21.0	29.6	34.43
$\mathcal{F}_{agg}$	21.0	30.5	35.5
$\mathcal{F}_{agg+}$	21.0	30.5	35.5
$\mathcal{F}_{agg-}$	21.0	30.5	35.5
$BG_{\mathcal{F}_{=}}$	0.0 %	3.0 %	3.0 %
$BG_{\mathcal{F}_{stat}}$	0.0 %	3.0 %	3.0 %

(c) *Budget*.
Congestion objective.

	Nb demands				
	10	15	20	25	30
$\mathcal{F}_{stat}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{=}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{cycle}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{+}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{-}$	3.0	4.936	6.236	6.5	7.739
$\overline{\mathcal{F}}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{agg}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{agg+}$	3.0	4.936	6.236	6.5	7.739
$\mathcal{F}_{agg-}$	3.0	4.936	6.236	6.5	7.739
$BG_{\mathcal{F}_{=}}$	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %
$BG_{\mathcal{F}_{stat}}$	0.0 %	0.0 %	0.0 %	0.0 %	0.0 %

(d) *All Routable Demands*.
Congestion objective.

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	1.300	1.424	1.491
$\mathcal{F}_{=}$	1.291	1.420	1.486
$\mathcal{F}_{cycle}$	1.291	1.420	1.486
$\mathcal{F}_{+}$	1.245	1.390	1.452
$\mathcal{F}_{-}$	1.240	1.381	1.448
$\overline{\mathcal{F}}$	1.163	1.312	1.380
$\mathcal{F}_{agg}$	1.291	1.420	1.486
$\mathcal{F}_{agg+}$	1.244	1.379	1.446
$\mathcal{F}_{agg-}$	1.240	1.380	1.447
$BG_{\mathcal{F}_{=}}$	9.9 %	7.6 %	7.1 %
$BG_{\mathcal{F}_{stat}}$	10.5 %	7.9 %	7.4 %

Demands polyhedron (19) leads to a larger model size in terms of number of variables and constraints. Observe, for example, that the model related to $\overline{\mathcal{F}}$ with the *Budget* polyhedron and 15 demands on the Abilene topology has five times more variables and six times more constraints than the model with the *All Routable Demands* polyhedron (Tables 3c and 3d).

We further observe that increasing the number of demands greatly increases the required solution time for the affine variants without aggregation (e.g., $\mathcal{F}_{=}$, $\mathcal{F}_{-}$, $\mathcal{F}_{+}$, $\mathcal{F}_{cycle}$, and $\overline{\mathcal{F}}$). The aggregation technique for the affine routing that we introduced (e.g., $\mathcal{F}_{agg}$, $\mathcal{F}_{agg-}$, $\mathcal{F}_{agg+}$) permits us to alleviate this drawback for large enough commodity set size (Tables 3a and 3c). This is somewhat explained by the fact that the number of x variables varies quadratically with the number of demands in the non-aggegated model while it is linear with the number of demands in the aggregated models.

Let us now focus on a given scenario, for instance, Abilene with *Budget* and the *Linear* objective (Table 1a), to compare the solution times. It can be observed that $\overline{\mathcal{F}}$, the routing scheme based on the extended graph formulation, takes

TABLE 2 Solutions for scenarios with Geant topology.

(a) *Budget*. *Linear* objective.

	Nb demands				
	10	15	20	25	30
$\mathcal{F}_{stat}$	48.00	71.87	94.00	118.00	131.48
$\mathcal{F}_{=}$	47.07	70.25	92.47	114.00	127.75
$\mathcal{F}_{cycle}$	47.07	70.25	92.47	114.00	127.75
$\mathcal{F}_{+}$	47.01	70.04	92.47	114.00	127.50
$\mathcal{F}_{-}$	46.96	69.83	91.96	113.71	127.39
$\overline{\mathcal{F}}$	46.94	69.20	90.83	110.53	123.22
$\mathcal{F}_{agg}$	47.07	70.25	92.47	114.00	127.75
$\mathcal{F}_{agg+}$	46.94	69.78	91.96	113.50	126.99
$\mathcal{F}_{agg-}$	46.94	69.46	91.79	113.50	127.04
$BG_{\mathcal{F}_{=}}$	0.3 %	1.5 %	1.8 %	3.1 %	3.6 %
$BG_{\mathcal{F}_{stat}}$	2.8 %	3.7 %	3.4 %	6.3 %	6.3 %

(b) *All Routable Demands*. *Linear* objective.

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	33.0	52.0	63.0
$\mathcal{F}_{=}$	33.0	52.0	63.0
$\mathcal{F}_{cycle}$	33.0	52.0	63.0
$\mathcal{F}_{+}$	33.0	52.0	63.0
$\mathcal{F}_{-}$	33.0	51.0	63.0
$\overline{\mathcal{F}}$	33.0	50.5	62.5
$\mathcal{F}_{agg}$	33.0	52.0	63.0
$\mathcal{F}_{agg+}$	33.0	51.0	63.0
$\mathcal{F}_{agg-}$	33.0	50.5	62.5
$BG_{\mathcal{F}_{=}}$	0.0 %	2.9 %	0.8 %
$BG_{\mathcal{F}_{stat}}$	0.0 %	2.8 %	0.8 %

(c) *Budget*. *Congestion* objective.

	Nb demands				
	10	15	20	25	30
$\mathcal{F}_{stat}$	1.621	2.311	3.174	3.252	3.836
$\mathcal{F}_{=}$	1.515	2.218	2.868	3.000	3.427
$\mathcal{F}_{cycle}$	1.515	2.218	2.868	3.000	3.427
$\mathcal{F}_{+}$	1.505	2.218	2.868	3.000	3.422
$\mathcal{F}_{-}$	1.515	2.218	2.868	3.000	3.427
$\overline{\mathcal{F}}$	1.505	2.218	2.868	3.000	3.422
$\mathcal{F}_{agg}$	1.515	2.218	2.868	3.000	3.422
$\mathcal{F}_{agg+}$	1.505	2.218	2.868	3.000	3.422
$\mathcal{F}_{agg-}$	1.515	2.218	2.868	3.000	3.422
$BG_{\mathcal{F}_{=}}$	0.6 %	0.0 %	0.0 %	0.0 %	0.2 %
$BG_{\mathcal{F}_{stat}}$	7.2 %	4.0 %	9.6 %	7.7 %	10.8 %

(d) *All Routable Demands*. *Congestion* objective.

	Nb demands	
	5	10
$\mathcal{F}_{stat}$	1.154	1.312
$\mathcal{F}_{=}$	1.133	1.301
$\mathcal{F}_{cycle}$	1.133	1.301
$\mathcal{F}_{+}$	1.091	1.282
$\mathcal{F}_{-}$	1.111	1.285
$\overline{\mathcal{F}}$	1.091	1.243
$\mathcal{F}_{agg}$	1.133	1.301
$\mathcal{F}_{agg+}$	1.091	1.282
$\mathcal{F}_{agg-}$	1.111	1.285
$BG_{\mathcal{F}_{=}}$	3.7 %	4.4 %
$BG_{\mathcal{F}_{stat}}$	5.5 %	5.3 %

the longest time to compute the optimal solution with respect to all the other approaches when varying the demand in the range [5-30]. This is indeed expected since the formulation of $\overline{\mathcal{F}}$ is more complex: for each commodity h, we added to the original graph $\mathcal{G}$ directed edges from the sink of h (i.e., $t(h)$) to each node $v \in V$ of $\mathcal{G}$, from each node v to the source of h (i.e., $s(h)$), and from $t(h)$ to $s(h)$. The complexity of this approach naturally increases with the number of demands and number of nodes. However, as mentioned above, $\overline{\mathcal{F}}$ shows the best solutions among other formulations.

Table 4 shows the impact of the topology on the problem size and the solution time. The problems related to Geant topology clearly have larger size than those related to Abilene and require much more time to be solved. While the number of nodes and edges is twice as large in Geant compared to Abilene, the number of variables and constraints for models can be between five and seven times larger.

TABLE 3 Scenarios with *Abilene* topology and *Congestion* objective: Impact of the polyhedron on the solution time.

(a) *Budget*.
Solution times (s).

	Nb demands				
	10	15	20	25	30
$\mathcal{F}_{stat}$	< 1	< 1	< 1	< 1	< 1
$\mathcal{F}_{=}$	< 1	1	5	12	29
$\mathcal{F}_{cycle}$	< 1	3	5	41	76
$\mathcal{F}_{+}$	< 1	2	7	23	89
$\mathcal{F}_{-}$	1	2	7	20	37
$\overline{\mathcal{F}}$	< 1	3	12	30	61
$\mathcal{F}_{agg}$	< 1	< 1	2	4	6
$\mathcal{F}_{agg+}$	< 1	< 1	2	5	6
$\mathcal{F}_{agg-}$	< 1	1	2	4	6

(b) *All Routable Demands*.
Solution times (s).

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	< 1	< 1	1
$\mathcal{F}_{=}$	6	447	2929
$\mathcal{F}_{cycle}$	9	170	2421
$\mathcal{F}_{+}$	9	415	5284
$\mathcal{F}_{-}$	9	402	3806
$\overline{\mathcal{F}}$	29	1136	11405
$\mathcal{F}_{agg}$	6	132	590
$\mathcal{F}_{agg+}$	8	340	2342
$\mathcal{F}_{agg-}$	6	189	2127

(c) *Budget*.
Size of the models.

		Nb demands				
		10	15	20	25	30
$\mathcal{F}_{stat}$	#col	961	1411	1861	2311	2761
	#row	770	1125	1480	1835	2190
$\mathcal{F}_{=}$	#col	6931	14881	25831	39781	56731
	#row	4950	10560	18270	28080	39990
$\mathcal{F}_{cycle}$	#col	5721	12241	21211	32631	46501
	#row	3630	7680	13230	20280	28830
$\mathcal{F}_{+}$	#col	9791	21121	36751	56681	80911
	#row	6380	13680	23730	36530	52080
$\mathcal{F}_{-}$	#col	9791	21121	36751	56681	80911
	#row	6380	13680	23730	36530	52080
$\overline{\mathcal{F}}$	#col	14851	32161	56071	86581	123691
	#row	7590	16320	28350	43680	62310
$\mathcal{F}_{agg}$	#col	4291	9121	15751	19501	23251
	#row	3036	6384	10962	13572	16182
$\mathcal{F}_{agg+}$	#col	8251	17761	30871	38221	45571
	#row	3828	8112	13986	17316	20646
$\mathcal{F}_{agg-}$	#col	9571	17761	28351	35101	41851
	#row	4411	8112	12873	15938	19003

(d) *All Routable Demands*.
Size of the models.

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	5251	8701	12151
	9415	18770	28125
$\mathcal{F}_{=}$	16201	49501	100801
	28440	103950	226560
$\mathcal{F}_{cycle}$	15871	48291	98161
	28080	102630	223680
$\mathcal{F}_{+}$	22116	69131	141946
	38580	144380	317430
$\mathcal{F}_{-}$	22116	69131	141946
	38580	144380	317430
$\overline{\mathcal{F}}$	27841	88381	182521
	47160	178590	394320
$\mathcal{F}_{agg}$	16201	44971	75481
	28410	94389	169696
$\mathcal{F}_{agg+}$	22741	64843	109669
	37770	127977	231208
$\mathcal{F}_{agg-}$	18703	51367	90793
	31152	101611	191712

TABLE 4 Scenarios with *Linear* objective and the *All Routable Demands* polyhedra: Impact of the topology on the solution time.

(a) *Abilene*.
Solution time (s).

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	< 1	< 1	< 1
$\mathcal{F}_{=}$	< 1	17	122
$\mathcal{F}_{cycle}$	< 1	19	213
$\mathcal{F}_{+}$	1	33	495
$\mathcal{F}_{-}$	1	32	606
$\overline{\mathcal{F}}$	2	457	3292
$\mathcal{F}_{agg}$	< 1	14	60
$\mathcal{F}_{agg+}$	1	30	272
$\mathcal{F}_{agg-}$	< 1	27	145

(b) *Geant*.
Solution times (s).

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	0.271	0.536	1.34
$\mathcal{F}_{=}$	13.57	406.57	3094.01
$\mathcal{F}_{cycle}$	18.5	340.56	3339.84
$\mathcal{F}_{+}$	33.05	1628.02	11155.9
$\mathcal{F}_{-}$	49.87	2220.21	12307.3
$\overline{\mathcal{F}}$	85.24	7480.31	53291.0
$\mathcal{F}_{agg}$	12.07	265.31	787.58
$\mathcal{F}_{agg+}$	18.0	1030.97	4561.98
$\mathcal{F}_{agg-}$	58.59	819.89	6972.41

(c) *Abilene*.
Size of the models.

		Nb demands		
		5	10	15
$\mathcal{F}_{stat}$	#col	5281	8731	12181
	#row	9415	18770	28125
$\mathcal{F}_{=}$	#col	16231	49531	100831
	#row	28440	103950	226560
$\mathcal{F}_{cycle}$	#col	15901	48321	98191
	#row	28080	102630	223680
$\mathcal{F}_{+}$	#col	22146	69161	141976
	#row	38580	144380	317430
$\mathcal{F}_{-}$	#col	22146	69161	141976
	#row	38580	144380	317430
$\overline{\mathcal{F}}$	#col	27871	88411	182551
	#row	47160	178590	394320
$\mathcal{F}_{agg}$	#col	16231	45001	75511
	#row	28410	94389	169696
$\mathcal{F}_{agg+}$	#col	22771	64873	109699
	#row	37770	127977	231208
$\mathcal{F}_{agg-}$	#col	18733	51397	90823
	#row	31152	101611	191712

(d) *Geant*.
Size of the models.

	Nb demands		
	5	10	15
$\mathcal{F}_{stat}$	25921	41401	56881
	52809	105474	158139
$\mathcal{F}_{=}$	78697	231337	463177
	158772	581372	1267872
$\mathcal{F}_{cycle}$	78067	229027	458137
	158112	578952	1262592
$\mathcal{F}_{+}$	99742	298727	602212
	200862	749502	1645992
$\mathcal{F}_{-}$	99742	298727	602212
	200862	749502	1645992
$\overline{\mathcal{F}}$	119872	363187	735202
	241122	910322	2007672
$\mathcal{F}_{agg}$	52345	189145	289081
	105786	475536	792144
$\mathcal{F}_{agg+}$	65611	246521	378379
	129942	604192	1009152
$\mathcal{F}_{agg-}$	100807	246521	495193
	199002	604192	1319232

8 | CONCLUSION

We have presented variants of the original affine routing formulation to further improve the solutions of the robust network design problem. We proposed a formulation $\mathcal{F}_{cycle}$ based on cycle decomposition that is equivalent to the initial node-arc formulation $\mathcal{F}_=$ of [44]. We also proved that a formulation based on paths is dominated by $\mathcal{F}_=$. Then two main ideas have been proposed: relaxation of flow conservation constraints and aggregation. The first idea led to $\mathcal{F}_-$ and $\mathcal{F}_+$ that have been combined into a stronger formulation $\overline{\mathcal{F}}$ by considering an extended graph. All these formulations are less conservative than $\mathcal{F}_=$. The second idea allowed us to build new formulations $\mathcal{F}_{agg}$, $\mathcal{F}_{agg+}$ and $\mathcal{F}_{agg-}$ that are respectively less conservative than formulations $\mathcal{F}_=$, $\mathcal{F}_+$ and $\mathcal{F}_-$. The striking fact is that aggregation simultaneously reduces the size of formulations as well as the solution's cost. Furthermore, we have proposed a cut-based formulation $\mathcal{F}_{cut}$ that improves over formulation $\overline{\mathcal{F}}$ but is generally NP-hard to solve. Finally, we illustrated our results with a numerical evaluation on two popular network topologies, two objective functions and two polyhedra.

Despite the efficiency of the new proposed formulations, several challenges remain. To solve larger size problems and reduce solution cost, it would be nice to find some way to combine aggregation with the extended-graph-based formulation $\overline{\mathcal{F}}$. Combining the uncertainty partitioning techniques (i.e., [3, 7, 12]) recalled in the introduction with some of the formulations introduced in the paper would be another challenge. Further investigations and comparisons with other routing schemes such as dynamic routing can also be conducted.

Acknowledgments

The authors are grateful to anonymous referees for their helpful suggestions that greatly improved the quality of the paper. We also point out that Remark 3.1 and the full proof of Proposition 4.1 have been added to address some comments of the reviewers.

References

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, 1993.

[2] Y. Al-Najjar, W. Ben-Ameur, and J. Leguay, *On the approximability of robust network design*, Theor. Comput. Sci. **860** (2021), 41–50.

[3] Y. Al-Najjar, S. Paris, J. Elias, J. Leguay, and W. Ben-Ameur, *Optimal routing configuration clustering through dynamic programming*, 21èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (ALGOTEL), 2019.

[4] D. Applegate and E. Cohen, *Making intra-domain routing robust to changing and uncertain traffic demands: Understanding fundamental tradeoffs*, Proc. ACM SIGCOMM, 2003, pp. 313–324.

[5] J. Ayoub and M. Poss, *Decomposition for adjustable robust linear optimization subject to uncertainty polytope*, Computational Manage. Scienc **13** (2016), 219–239.

[6] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Räcke, *Optimal oblivious routing in polynomial time*, J. Comput. System Sci. **69** (2004), 383–394.

[7] W. Ben-Ameur, *Between fully dynamic routing and robust stable routing*, 6th International Workshop on Design and Reliable Communication Networks, 2007, pp. 1–6.

[8] W. Ben-Ameur and H. Kerivin, *New economical virtual private networks*, Commun. ACM **46** (2003), 69–69.

[9] W. Ben-Ameur and H. Kerivin, *Routing of uncertain traffic demands*, Optim. Eng. **6** (2005), 283–313.

[10] W. Ben-Ameur, A. Ouorou, G. Wang, and M. Żotkiewicz, *Multipolar robust optimization*, EURO J. Comput. Optim. **6** (2018), 395–434.

[11] W. Ben-Ameur, A. Ouorou, and M. Żotkiewicz, *Robust routing in communication networks*, Progress in Combinatorial Optimization, A. R. Mahjoub, Ed., Wiley, New York, (2011), 353–390.

[12] W. Ben-Ameur and M. Żotkiewicz, *Robust routing and optimal partitioning of a traffic demand polytope*, Int. Trans. Oper. Res. **18** (2011), 307–333.

[13] W. Ben-Ameur and M. Żotkiewicz, *Multipolar routing: Where dynamic and static routing meet*, Electron. Notes Discr. Math. **41** (2013), 61 – 68.

[14] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, Princeton University Press, 2009.

[15] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski, *Adjustable robust solutions of uncertain linear programs*, Math. Program. **99** (2004), 351–376.

[16] D. Bertsimas and V. Goyal, *On the power and limitations of affine policies in two-stage adaptive optimization*, Math. Program. **134** (2012), 491–531.

[17] D. Bertsimas, V. Goyal, and X. Sun, *A geometric characterization of the power of finite adaptability in multistage stochastic and adaptive optimization*, Math. Oper. Res. **36** (2011), 24–54.

[18] D. Bertsimas, D.A. Iancu, and P.A. Parrilo, *Optimality of affine policies in multistage robust optimization*, Math. Oper. Res. **35** (2010), 363–394.

[19] D. Bertsimas and M. Sim, *Robust discrete optimization and network flows*, Math. Program. **98** (2003), 49–71.

[20] C. Chekuri, *Routing and network design with robustness to changing or uncertain traffic demands*, SIGACT News **38** (2007), 106–129.

[21] C. Chekuri, F.B. Shepherd, G. Oriolo, and M.G. Scutella, *Hardness of robust network design*, Networks **50** (2007), 50–54.

[22] G. Claßen, A.M.C.A. Koster, M. Kutschka, and I. Tahiri, *Robust metric inequalities for network loading under demand uncertainty*, Asia-Pacific J. Oper. Res. **32** (2015).

[23] A.M. Costa, *A survey on Benders decomposition applied to fixed-charge network design problems*, Comput. Oper. Res. **32** (2005), 1429 – 1450.

[24] E. Delage and D.A. Iancu, *Robust multistage decision making*, INFORMS TutORials Oper. Res. (2015), 20–46.

[25] N.G. Duffield, P. Goyal, A. Greenberg, P. Mishra, K.K. Ramakrishnan, and J.E. van der Merive, *A flexible model for resource management in virtual private networks*, SIGCOMM ('99), 95–108.

[26] J. Fingerhut, S. Suri, and J.S. Turner, *Designing least-cost nonblocking broadband networks*, J. Algorithms **24** (1997), 287 – 309.

[27] A. Frangioni and B. Gendron, *0–1 reformulations of the multicommodity capacitated network design problem*, Discr. Appl. Math. **157** (2009), 1229 – 1241.

[28] A. Georghiou, W. Wiesemann, and D. Kuhn, *Generalized decision rule approximations for stochastic programming via liftings*, Math. Program. **152** (2015), 301–338.

[29] M. Gondran, M. Minoux, and S. Vajda, *Graphs and Algorithms*, John Wiley & Sons, Inc., USA, 1984.

[30] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Vol. 2, Springer Science & Business Media, 2012.

[31] O. Günlük, *A new min-cut max-flow ratio for multicommodity flows*, International Conference on Integer Programming and Combinatorial Optimization, 2002, pp. 54–66.

[32] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, and B. Yener, *Provisioning a virtual private network: A network design problem for multicommodity flow*, Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing, 2001, pp. 389–398.

[33] D. Kuhn, W. Wiesemann, and A. Georghiou, *Primal and dual linear decision rules in stochastic and robust optimization*, Math. Program. **130** (2011), 177–209.

[34] H.P.L. Luna, "*Network planning problems in telecommunications*," *Handbook of Optimization in Telecommunications*, Springer, 2006, pp. 213–240.

[35] S. Mattia, *The robust network loading problem with dynamic routing*, Comput. Optim. Appl. **54** (2013), 619–643.

[36] O. Michel and E. Keller, *SDN in wide-area networks: A survey*, 4th International Conference on Software Defined Systems (SDS), 2017, pp. 37–42.

[37] M. Minoux, *Robust network optimization under polyhedral demand uncertainty is NP-hard*, Discr. Appl. Math. **158** (2010), 597 – 603.

[38] J.T. Moy, *OSPF: Anatomy of an Internet Routing Protocol*, Addison-Wesley Professional, 1998.

[39] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski, *SNDlib 1.0—Survivable network design library*, Networks **55** (2010), 276–286.

[40] A. Ouorou, *Tractable approximations to a robust capacity assignment model in telecommunications under demand uncertainty*, Comput. Oper. Res. **40** (2013), 318–327.

[41] A. Ouorou and J. Vial, *A model for robust capacity planning for telecommunications networks under demand uncertainty*, Workshop on Design and Reliable Communication Networks, 2007.

[42] F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, and P. Castoldi, *A survey on the path computation element (PCE) architecture*, IEEE Commun. Surveys Tutorials **15** (2013), 1819–1841.

[43] M. Poss, *A comparison of routing sets for robust network design*, Optim. Lett. **8** (2014), 1619–1635.

[44] M. Poss and C. Raack, *Affine recourse for the robust network design problem: Between static and dynamic routing*, Network **61** (2013), 180–198.

[45] K. Postek and D. den Hertog, *Multistage adjustable robust mixed-integer optimization via iterative splitting of the uncertainty set*, INFORMS J. Computi **28** (2016), 553–574.

[46] C. Raack, A.M. Koster, S. Orlowski, and R. Wessäly, *On cut-based inequalities for capacitated network design polyhedra*, Networks **57** (2011), 141–156.

[47] H. Räcke, *Optimal hierarchical decompositions for congestion minimization in networks*, Proceedings of the fortieth annual ACM symposium on Theory of computing, 2008, pp. 255–264.

[48] M.G. Scutellà, *On improving optimal oblivious routing*, Oper. Res. Lett. **37** (2009), 197–200.

[49] M. Silva, M. Poss, and N. Maculan, *Solving the bifurcated and nonbifurcated robust network loading problem with k-adaptive routing*, Networks **72** (2018), 151–170.

[50] D. Simchi-Levi, H. Wang, and Y. Wei, *Constraint generation for two-stage robust network flow problems*, INFORMS J. Optim. **1** (2019), 49–70.

[51] N. Wang, K.H. Ho, G. Pavlou, and M. Howarth, *An overview of routing optimization for internet traffic engineering*, IEEE Commun. Surveys Tutorials **10** (2008), 36–56.

[52] I. Yanikoglu, B. Gorissen, and D. Den Hertog, *A survey of adjustable robust optimization*, Eur. J. Oper. Res. **277** (2019), 799–813.

[53] M. Żotkiewicz and W. Ben-Ameur, *Volume-oriented routing and its modifications*, Telecommunication Syst. **52** (2013), 935–945.