# Network optimization problems subject to max-min fair flow allocation

Edoardo Amaldi, Antonio Capone, Stefano Coniglio, and Luca G. Gianoli

***Abstract*—We propose a novel way to consider the max-min fairness (MMF) paradigm in traffic engineering. Since MMF appears as a reference model for a fair capacity allocation when the traffic flows are elastic and rates are adapted based on resource availability, we consider it as a requirement due to the way resources are shared by the transportation protocol, rather than the routing objective. In particular, we address the traffic engineering problem where, given a network topology with link capacities and a set of communications to route, we must select a single path for each communication so as to maximize a network utility function, assuming a MMF bandwidth allocation. We give a compact mixed-integer linear programming formulation as well as a restricted path model. Computational experiments show that the exact formulation can be solved in a reasonable amount of computing time for medium-size networks and that the restricted path model provides solutions of comparable quality much faster.**

**Index Terms – Max-Min Fairness, Traffic Engineering, Bandwidth Sharing, Utility-based Routing**

## I. Introduction

Recently, a growing attention has been devoted to the problem of fair bandwidth (or flow or rate) allocation in telecommunication networks, with emphasis on the so-called *max-min fair* (MMF) paradigm, see the survey [1] and the references therein. Informally, a bandwidth allocation is max-min fair if there is no way to give more bandwidth to any communication without decreasing the allocation to a communication receiving less or equal bandwidth. In other words, this amounts to lexicographically maximize the bandwidth allocated to the various communications, considering the communications in non-decreasing order of bandwidth.

The MMF paradigm is of substantial interest for IP (Internet Protocol) networks because it is considered the reference model for a fair allocation of network capacity in the case of traffic flows that are elastic and can adapt their rate based on resource availability. The concept of best-effort service in the Internet can be associated to that of MMF since the network is expected to provide the best possible service in terms of rate without privileging any traffic flow.

Previous works on MMF network optimization deal with different bandwidth allocation and routing settings. If a routing path has already been selected for each communication, a simple polynomial-time algorithm, known as *Water (or Proportional) filling*, suffices to allocate the bandwidth to the communications in a MMF way, see e.g. [2]. If the routing paths are not known a priori, algorithms have been proposed to determine a routing pattern such that the MMF bandwidth allocation is as fair as possible for splittable routing (see [3, Chapter 8] and the survey [1]) or unsplittable routing (see [4], [5], and [3, Chapter 8]). In case of a general optimization problem with a convex feasible region, the solution approach amounts to solving a sequence of convex problems, at most one for each communication, see e.g. [6]. The reader is also referred to [1] and [7].

To the best of our knowledge, so far the MMF paradigm has only been considered as a routing objective, rather than as a requirement of a more general traffic engineering problem. This is in spite of the fact that in IP networks the distributed congestion control mechanism, due to transport protocols such as TCP (Transmission Control Protocol), leads to an average bandwidth allocation which, after the routing paths have been provided by the IP layer (assuming similar delays for all the flows) is well approximated by MMF [8].

In practice, network operators are interested in optimizing routing according to one of the classical traffic engineering objectives, while assuming that the bandwidth is allocated in a MMF way which cannot be directly controlled. The resulting network routing problem can thus be viewed as a bilevel optimization problem where, at the upper level, a leader (network operator) chooses a single routing path for each communication so as to maximize a utility function and, at the lower level, a follower (transport protocol) allocates the bandwidth to the paths chosen by the leader, according to the MMF paradigm.

In this work, we consider the problem of, given a network topology with link capacities and a set of communications, selecting a single path for each communication so as to maximize a network utility function, subject to MMF bandwidth allocation. We show how this MMF-Constrained Traffic Engineering problem (MMF-CTE) can be formulated as a single-level Mixed-Integer Linear Program (MILP) with a polynomial number of constraints and 0-1 variables, which is solvable in a reasonable amount of computing time for medium-size networks. We also provide a restricted path formulation and compare its solutions to those obtained with the previous exact formulation in

E. Amaldi, A. Capone and S. Coniglio are with Dipartimento di Elettronica e Informazione (DEI), *Politecnico di Milano*, Italy. L.G. Gianoli is with both DEI, *Politecnico di Milano* and the Département de Génie Electrique, *École Polytechnique de Montreal*, Canada.

terms of quality and computing time.

## II. MMF-constrained Traffic Engineering

Let $G = (V, A)$ be a directed graph representing the network topology with a capacity $c_{ij} \geq 0$ for each arc $(i, j) \in A$ and let $K$ be a set of communications, specified by the corresponding origin-destination pairs $(s, t)$, with $(s, t) \in K$.

Let $\phi^{st}$ denote the amount of bandwidth allocated to each origin-destination pair $(s, t) \in K$. As utility function, we consider either the total weighted throughput

$$\max \sum_{(s,t)\in K} w_{st}\phi^{st}, \qquad (1)$$

which is a simple weighted sum of the allocated bandwidths with real weights $w_{st}$, or

$$\max \sum_{(s,t)\in K} w_{st}\alpha(1 - e^{\frac{1}{\beta}\phi^{st}}), \qquad (2)$$

for suitable $\alpha, \beta > 0$, which favors the increase of a small bandwidth rather than of a large one. The latter function is linearized by means of the standard linear programming piecewise-affine approximation of concave functions, see for instance [9], with 6 pieces. This amounts to introducing a single continuous variable per origin-destination pair and a linear constraint per piece.

A MMF flow (bandwidth) allocation is formally defined as follows. Let $\underline{\phi} \in \mathbb{R}_+^{|K|}$ be the vector of flows, with one component per origin-destination pair. Let $\sigma$ be the sorting operator that permutes the components of a vector in nondecreasing order. $\underline{\phi}$ is MMF if and only if, for any other vector $\underline{\phi}' \in \mathbb{R}_+^{|K|}$, $\sigma(\underline{\phi})$ lexicographically dominates $\sigma(\underline{\phi}')$, that is, either $\sigma(\underline{\phi}) = \sigma(\underline{\phi}')$ or there exists an integer $l$, with $1 \leq l \leq |K|$, such that $\sigma(\underline{\phi})_l > \sigma(\underline{\phi}')_l$ and $\sigma(\underline{\phi})_k = \sigma(\underline{\phi}')_k$ for all $\leq k < l$. In other words, if any other flow vector allocates more flow to an origin-destination pair, then it allocates less to another origin-destination pair receiving a smaller flow. The reader is referred to [1].

The following example illustrates the substantial difference between our MMF-constrained traffic engineering problem where we look for a MMF solution w.r.t. a set of (single) routing paths that optimizes the objective function (1) with uniform weights $w_{st}$, and the standard single path routing problem where we look for a solution which is overall MMF. The example can also be adapted to the objective function (2).

Example: Consider the network in Figure 1 with origin-destination pairs $(s_1, t_1), \ldots, (s_5, t_5)$, where the dashed lines represent paths with a capacity of at least 5, the link $(a_1, b_1)$ has a capacity of 10, and the links $(a_i, b_i)$, with $i = 2, \ldots, 5$, have a capacity of 1. We consider two solutions a) and b) where the communications are routed along different paths, with a MMF bandwidth allocation. In a), each communication $i$, with $1 \leq i \leq 5$, is routed along the path containing the nodes $s_i$, $a_i$, $b_i$ and $t_i$. Since these paths are disjoint, the MMF bandwidth allocation trivially amounts to sending the maximum flow on each path. Thus, we have $\sigma(\underline{\phi}) = (1, 1, 1, 1, 10)$ and a total throughput of 14. In b), all the communications $i$, with $1 \leq i \leq 5$, are routed through the link $(a_1, b_1)$, i.e., using the paths $s_i$, $a_1$, $b_1$, $t_i$. Then $\sigma(\underline{\phi}) = (2, 2, 2, 2, 2)$ with a total throughput of 10. Any other solution is obtained by routing any strict subset of the communications in $\{2, \ldots, 5\}$ through the link $(a_1, b_1)$. For any subset of cardinality $h$, with $1 \leq h \leq 3$, we clearly obtain $\sigma(\underline{\phi})_l = \frac{10}{h+1}$ for the communications routed over $(a_1, b_1)$ and $\sigma(\underline{\phi})_l = 1$ for all the other ones. As to the MMF bandwidth allocation, any of those solutions is clearly dominated by b) and provides a total throughput of $10 + 4 - h$, which is inferior to that in a). Therefore, a) is an optimal solution to MMF-CTE, whereas b) is an optimal solution to the MMF single path routing problem.

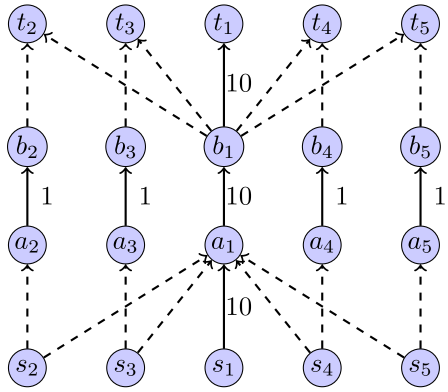


Figure 1. Representation of the capacited network and five origin-destination pairs discussed in the example.

## III. MILP formulation

For each origin-destination pair $(s, t) \in K$, we define the flow variables $\phi^{st}$ and $f_{ij}^{st}$, which represent, respectively, the total flow assigned to the $(s, t)$ pair and the amount of flow on link $(i, j) \in A$. We also introduce the binary variables $x_{ij}^{st}$ that are equal to 1 if $f_{ij}^{st} > 0$, and 0 otherwise.

Besides the usual flow conservation and capacity constraints

$$\sum_{(i,j)\in A} f_{ij}^{st} - \sum_{(j,i)\in A} f_{ji}^{st} = \begin{cases} \phi^{st} & \text{if } i = s \\ -\phi_{st} & \text{if } i = t \\ 0 & \text{else} \end{cases} \quad \forall i \in V, \forall (s,t) \in K \qquad (3)$$

$$f_{ij}^{st} \leq c_{ij} x_{ij}^{st} \qquad \forall (i,j) \in A, \forall (s,t) \in K \qquad (4)$$

we introduce the following degree constraints on the outgoing star of each node $i$:

$$\sum_{(i,j)\in A} x_{ij}^{st} \leq 1 \quad \forall i \in V, \forall (s,t) \in K \qquad (5)$$

to ensure that each communication is routed along a single path.

Let us now explain how to impose a MMF bandwidth allocation. If a routing path were already known for each $(s, t)$ pair, a MMF bandwidth allocation could be determined via the Water Filling algorithm [2], which we briefly summarize. Starting from the solution with $\phi^{st} = 0$ for all

$(s,t) \in K$, all the flows are simultaneously increased until one or more arcs are (simultaneously) saturated. We then remove such *bottleneck arcs* and all the communications which saturated them, update each capacity to its residual value, and iterate until no communications or no arcs are left.

For each arc $(i,j) \in A$ and origin-destination pair $(s,t) \in K$, we thus introduce the binary variable $y_{ij}^{st}$, which is equal to 1 if $(i,j)$ is a bottleneck arc for the pair $(s,t)$, and 0 otherwise. Due to the correctness of the above algorithm, a flow vector $\underline{\phi}$ is MMF if and only if it satisfies the following constraints which are slightly restated version of those in [6]:

$$\sum_{(i,j)\in A} y_{ij}^{st} \geq 1 \qquad \forall (s,t) \in K \qquad (6)$$

$$\sum_{(o,d)\in K} f_{ij}^{od} \geq c_{ij} y_{ij}^{st} \qquad \forall (i,j) \in A, \forall (s,t) \in K \quad (7)$$

$$f_{ij}^{st} \geq f_{ij}^{od} - c_{ij}(1 - y_{ij}^{st}) \qquad \forall (i,j) \in A, \forall (s,t) \in K, \ \forall (o,d) \in K \qquad (8)$$

Constraints (6) ensure that we have at least a bottleneck arc for each $(s,t)$ pair, while Constraints (7) guarantee that the bottleneck arcs are saturated. Constraints (8) impose that the flow through a bottleneck arc $(i,j)$ for a pair $(s,t)$ be at least as large as the flow through $(i,j)$ for all the other origin-destination pairs.

Unfortunately, without further constraints, subtours may arise. To see this, consider a communication $(s,t)$ with a pre-defined $s-t$ routing path. Note that the introduction of a subtour does not change the value of $\phi^{st}$, due to the balance constraints. Consider now a subtour defined by a subset of arcs $S \subset A$ not involving the $s-t$ path and an arc $(v,w) \in S$ such that $c_{vw} = \min_{(i,j)\in S}\{c_{ij}\}$. Then, letting $y_{vw}^{st} = 1$, $f_{ij}^{st} = c_{vw}$ for all $(i,j) \in S$, and $f_{ij}^{od} = 0$ for any $(i,j) \in S$ and $(o,d) \in K \setminus \{(s,t)\}$, the MMF constraints in (6)-(8) are trivially satisfied on the subtour, allowing for a non MMF flow on the actual $s-t$ path.

To avoid this drawback, we can introduce the standard subtour elimination constraints which are employed when formulating the Traveling Salesman Problem (TSP) as an Integer Program. Since there are exponentially many constraints of such type and we aim at a MILP formulation of compact (polynomial) size, we eliminate the occurrence of subtours by introducing a modified version of the constraints and continuous variables adopted in the compact (extended) formulation proposed in [10] for the TSP. For the lack of space, we do not report them here.

## IV. Enhanced formulation

We also consider two simple valid inequalities which yield tighter LP relaxations and, overall, accelerate the convergence of the Branch-and-Bound method. First, we introduce

$$y_{ij}^{st} \leq x_{ij}^{st} \quad \forall (i,j) \in A, \forall (s,t) \in K \qquad (9)$$

which adds a trivial valid connection between variables $y_{ij}^{st}$ and $x_{ij}^{st}$. Then, we add

$$\phi^{st} \geq \frac{\min_{(i,j)\in A}\{c_{ij}\}}{|K|} \quad \forall (s,t) \in K \qquad (10)$$

which is valid since any MMF flow for an origin-destination pair $(s,t)$ will saturate at least a link and, due to the MMF allocation of bandwidth, the smallest quantity of bandwidth will be allocated when the link is simultaneously shared by all the communications.

Finally, a more compact formulation is obtained by introducing the nonnegative variables $u_{ij}$ for each $(i,j) \in A$ and replacing Constraints (8) with:

$$u_{ij} \geq f_{ij}^{st} \qquad \forall (i,j) \in A, (s,t) \in K \ (11)$$

$$f_{ij}^{st} \geq u_{ij} - c_{ij}(1 - y_{ij}^{st}) \quad \forall (i,j) \in A, (s,t) \in K \ (12)$$

To summarize, we maximize either (1) or (2) subject to Constraints (3), (4), (5), (6), (7), (11), (12), (9), (10), as well as the extended formulation for the subtour elimination constraints in [10], properly modified to suit our problem.

## V. Computational results

Computational experiments have been carried out on a set of network topologies taken from the SND library [11]. We consider 4 networks (`polska`, $n = 12, m = 36$, `abilene`, $n = 12, m = 30$, `nobel-us`, $n = 14, m = 42$, `atlanta`, $n = 15, m = 44$) and generate, for each of them, four instances by adopting a different set of origin-destination pairs, for a total of 20 instances. Since capacities are not provided for SND library instances, we assume, for simplicity, that every link has the same capacity of 1000 Mbps. At the end of this section, we also comment on the results obtained for random capacities.

Our MILP formulations are solved with CPLEX 12.3 (with parameter `mipemphasis=4`) using the AMPL modeling language. The computational experiments are carried out on a machine equipped with 4 Intel i7 processors and 8 GB of RAM. A time limit of 3600 seconds is imposed. For simplicity, we assume that all the communications have the same weight $w^{st} = 1$. In the objective function (2), we let $\alpha = 1000$ and $\beta = 200$.

To speed up the computations, even though at the cost of possibly obtaining suboptimal solutions, we also consider restricted path models where routing is restricted to a set of 10 or 20 predetermined paths for each origin-destination pair, which are randomly generated in a preprocessing phase.

Table I reports the results obtained when optimizing the linear sum of the throughputs, namely objective function (1).

Within the time limit, 10 out of 20 instances are solved to optimality. Overall, the integrality gap is quite small, below 3% on average. The two restricted path formulations with predefined random paths are much easier to solve, as shown by the substantially smaller integrality gap ($< 1\%$ and $< 1.6\%$, respectively). Notably, the model with 10 (respectively 20) random paths is solved, on average, in less than 4% (6%) of the computing time needed to tackle the complete MILP model. The geometric mean of the ratios between the objective function value of the solutions found with the restricted path models and the exact one show a loss of less than 20% in solution quality for the restricted model with 10 paths per origin-destination pair and, most interestingly, an improvement of 2% for that with 20 paths.

Table I
RESULTS WITH THE EXACT AND THE RESTRICTED PATHS FORMULATIONS WITH OBJECTIVE FUNCTION (1).

| | | *Exact* | | | *Restr. 10 paths* | | | *Restr. 20 paths* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| ***Net*** | $\|K\|$ | ***Thr*** | ***Gap*** | ***Time*** | ***Thr*** | ***Gap*** | ***Time*** | ***Thr*** | ***Gap*** | ***Time*** |
| pol | 6 | **5000.0** | 0.0 | 0.4 | **5000.0** | 0.0 | 0.1 | **5000.0** | 0.0 | 0.1 |
| pol | 10 | **7666.6** | 0.0 | 45.7 | **7666.7** | 0.0 | 0.1 | **7666.7** | 0.0 | 0.1 |
| pol | 21 | **10000.0** | 0.0 | 38.3 | 9875.0 | 0.0 | 3.9 | **10000.0** | 0.0 | 1.1 |
| pol | 28 | **11800.0** | 1.7 | 3600.0 | 11259.3 | 0.0 | 16.8 | 11454.5 | 0.0 | 86.8 |
| pol | 36 | **12800.0** | 9.3 | 3600.0 | 12666.7 | 0.0 | 998.7 | 12772.7 | 0.0 | 2312.3 |
| abi | 12 | **8000.0** | 0.0 | 0.1 | **8000.0** | 0.0 | 0.0 | **8000.0** | 0.0 | 0.0 |
| abi | 20 | **9000.0** | 0.0 | 2.0 | **9000.0** | 0.0 | 0.1 | **9000.0** | 0.0 | 0.2 |
| abi | 30 | **12000.0** | 0.0 | 394.3 | 11928.6 | 0.0 | 58.3 | **12000.0** | 0.0 | 78.0 |
| abi | 42 | 13166.7 | 13.4 | 3600.0 | **13277.8** | 2.7 | 3600.0 | 13166.7 | 7.8 | 3600.0 |
| abi | 56 | **15404.8** | 10.2 | 3600.0 | 14381.4 | 13.4 | 3600.0 | 14484.8 | 14.3 | 3600.0 |
| n-u | 6 | **6000.0** | 0.0 | 0.5 | **6000.0** | 0.0 | 0.1 | **6000.0** | 0.0 | 0.1 |
| n-u | 10 | **8000.0** | 0.0 | 4.0 | **8000.0** | 0.0 | 0.1 | **8000.0** | 0.0 | 0.1 |
| n-u | 15 | **10333.3** | 1.6 | 3600.0 | 9750.0 | 0.0 | 0.4 | 10125.0 | 0.0 | 2.0 |
| n-u | 21 | **11933.3** | 0.6 | 3600.0 | 11500.0 | 0.0 | 0.9 | 11750.0 | 0.0 | 1.3 |
| n-u | 28 | **13333.3** | 5.0 | 3600.0 | 13066.7 | 0.0 | 8.7 | 13166.7 | 0.0 | 10.9 |
| atl | 6 | **6000.0** | 0.0 | 2.3 | **10000.0** | 0.0 | 0.0 | **10000.0** | 0.0 | 0.1 |
| atl | 12 | **6000.0** | 0.0 | 0.4 | **6000.0** | 0.0 | 0.0 | **6000.0** | 0.0 | 0.0 |
| atl | 20 | **12666.7** | 2.6 | 3600.0 | **12666.7** | 0.0 | 2.6 | **12666.7** | 0.0 | 152.2 |
| atl | 30 | 14214.3 | 5.5 | 3600.0 | 14133.3 | 0.8 | 3600.0 | **14533.3** | 3.1 | 3600.0 |
| atl | 42 | 15800.0 | 7.6 | 3600.0 | 15937.5 | 1.7 | 3600.0 | **16166.7** | 5.1 | 3600.0 |
| Aggreg. | | | 2.871 | | 0.805 | 0.925 | 0.039 | 1.021 | 1.515 | 0.054 |

Table II reports the results obtained when optimizing the piecewise affine utility function (2).

Table II
RESULTS WITH THE EXACT AND THE RESTRICTED PATHS FORMULATIONS WITH OBJECTIVE FUNCTION (2).

| | | *Exact* | | | *Restr. 10 paths* | | | *Restr. 20 paths* | | |
|---|---|---|---|---|---|---|---|---|---|---|
| ***Net*** | $\|K\|$ | ***Thr*** | ***Gap*** | ***Time*** | ***Thr*** | ***Gap*** | ***Time*** | ***Thr*** | ***Gap*** | ***Time*** |
| pol | 6 | **5787.9** | 0.0 | 0.7 | **5787.9** | 0.0 | 0.1 | **5787.9** | 0.0 | 0.1 |
| pol | 10 | **9503.5** | 0.0 | 1.4 | **9503.5** | 0.0 | 0.0 | **9503.5** | 0.0 | 0.1 |
| pol | 21 | **17394.2** | 1.1 | 3593.3 | 17324.3 | 0.0 | 74.8 | 17338.6 | 0.0 | 1151.4 |
| pol | 28 | **22068.6** | 0.5 | 3593.3 | 21932.8 | 0.0 | 47.9 | 21997.1 | 0.0 | 186.0 |
| pol | 36 | **26681.6** | 0.0 | 2797.8 | **26681.6** | 0.0 | 12.4 | 26624.5 | 0.0 | 66.2 |
| abi | 12 | **11232.5** | 0.0 | 0.3 | **11232.5** | 0.0 | 0.0 | **11232.5** | 0.0 | 0.0 |
| abi | 20 | **16179.9** | 0.0 | 0.6 | **16179.9** | 0.0 | 0.1 | **16179.9** | 0.0 | 0.1 |
| abi | 30 | **22188.7** | 0.0 | 0.2 | **22188.7** | 0.0 | 1.0 | **22188.7** | 0.0 | 1.8 |
| abi | 42 | 27303.9 | 1.4 | 3593.4 | 27226.0 | 1.2 | 3593.5 | **27315.5** | 1.0 | 3593.5 |
| abi | 56 | 31968.2 | 3.0 | 3593.8 | **32055.8** | 2.4 | 3593.5 | 32020.2 | 2.6 | 3593.5 |
| n-u | 6 | **5959.6** | 0.0 | 0.3 | **5959.6** | 0.0 | 0.0 | **5959.6** | 0.0 | 0.0 |
| n-u | 10 | **9589.3** | 0.0 | 294.1 | **9589.3** | 0.0 | 0.1 | **9589.3** | 0.0 | 0.5 |
| n-u | 15 | **14040.7** | 0.0 | 1154.4 | 13954.9 | 0.0 | 0.7 | **14040.7** | 0.0 | 1.3 |
| n-u | 21 | **18867.0** | 0.2 | 3593.3 | **18867.0** | 0.0 | 1.4 | **18867.0** | 0.0 | 13.1 |
| n-u | 28 | **23924.0** | 0.9 | 3593.3 | 23874.1 | 0.0 | 42.9 | 23874.1 | 0.0 | 47.3 |
| atl | 6 | **11575.8** | 0.0 | 5.0 | **11575.8** | 0.0 | 0.2 | **11575.8** | 0.0 | 0.2 |
| atl | 12 | **5959.6** | 0.0 | 0.4 | **5959.6** | 0.0 | 0.0 | **5959.6** | 0.0 | 0.0 |
| atl | 20 | **18320.4** | 0.9 | 3593.4 | **18320.4** | 0.0 | 59.7 | **18320.4** | 0.4 | 3593.4 |
| atl | 30 | 25305.9 | 0.8 | 3593.3 | 25055.5 | 0.0 | 501.0 | **25311.0** | 0.6 | 3593.4 |
| atl | 42 | **31298.0** | 0.0 | 176.3 | 31012.4 | 0.0 | 3.7 | 31276.4 | 0.0 | 10.9 |
| Aggreg. | | | 0.439 | | 0.998 | 0.179 | 0.031 | 1.000 | 0.232 | 0.078 |

Interestingly, 12 out of 20 instances are solved to optimality within the time limit. Since this utility function is not as flat as the linear sum of throughputs, it may yield solutions with more varied bounds allowing for a heavier pruning in the branch-and-bound tree, possibly explaining the faster convergence that we observe. Indeed, on average, the integrality gap is smaller that 0.5%. Even for the 8 instances for which we did not find an optimal solution, the best solution found is still very close to be optimal. The two restricted path models with 10 (respectively 20) paths are solved within a gap smaller than 0.2% (0.24%), in 3% (8%) of the time needed to solve the exact formulation. Quite interestingly, they yield very high quality solutions which are almost equivalent, in terms of the piecewise-affine utility, to those obtained with the exact formulation.

Due to the lack of space, we do not report the detailed results obtained for the instances with nonuniform capacities. We just mention that our exact MILP formulation with objective function (1) could, in some cases, be solved to optimality substantially faster (possibly due to the reduced problem symmetry) and that the restricted path models tend to provide slightly worse quality solutions.

## VI. CONCLUSIONS AND FURTHER RESEARCH

We have proposed a traffic engineering problem where, given a network topology with link capacities and a set of communications, we must select a single path for each communication so as to maximize a network utility function, assuming a MMF bandwidth allocation. We have shown that this problem can be cast as a single mixed-integer linear program with a polynomial number of variables and constraints, which is solvable in a reasonable amount of computing time for medium-size networks. Motivated by the very good computational results obtained with the restricted path models, future algorithmic work includes the development of a column generation method. From a networking point of view, our approach of considering max-min fairness as a constraint on the flow allocation, rather than an objective, can also be adapted and extended to network design problems, such as energy-aware traffic engineering.

## REFERENCES


[1] D. Nace and M. Pióro. Max-min fairness and its applications to routing and load-balancing in communication networks: a tutorial. *Communications Surveys & Tutorials, IEEE*, 10(4):5–17, 2008.

[2] D. Bertsekas and R. Gallager. *Data Networks, 1992.* Prentice-Hall, 1992.

[3] M. Pióro and D. Medhi. *Routing, Flow and Capacity Design in Communication and Computer Networks.* Morgan Kaufman, 2004.

[4] P. Nilsson. *Fairness in Communication and Computer Network Design.* PhD thesis, Lund University, Sweden, 2006.

[5] W. Ogryczak, M. Pióro, and A. Tomaszewski. Telecommunications network design and max-min optimization problem. *Journal of Telecommunications and Information Technology*, 3:1–14, 2005.

[6] A. Tomaszewski. A polynomial algorithm for solving a general max-min fairness problem. *European transactions on telecommunications*, 16(3):233–240, 2005.

[7] B. Radunovic and J.-Y. Le Boudec. A unified framework for max-min and min-max fairness with applications. *Networking, IEEE/ACM Transactions on*, 15(5):1073 –1083, 2007.

[8] L. Massoulié and J. Roberts. Bandwidth sharing: objectives and algorithms. *Networking, IEEE/ACM Transactions on*, 10(3):320–328, 2002.

[9] H.P. Williams. *Model building in mathematical programming.* John Wiley and Sons, 1999.

[10] R.T. Wong. Integer programming formulations of the travelling salesman problem. In *Proc. IEEE Conf. on Circuits and Computers*, pages 149–152, 1980.

[11] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski. SNDlib 1.0 - survivable network design library. *Networks*, 55(3):276–286, 2010.