Optical Switching and Networking 11 (2014) 92–104

Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

On joint topology design and load balancing in free-space optical networks ☆

In Keun Son [a], Shiwen Mao [b,*], Sajal K. Das [c]

[a] Defense Acquisition Program Administration (DAPA), Republic of Korea
[b] Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201, USA
[c] Department of Computer Science and Engineering, University Texas at Arlington, Arlington, TX 76019, USA

ARTICLE INFO

Available online 9 August 2013

Keywords:
Free-space optics
Load balancing
Multipath routing
Topology design
Optimization

ABSTRACT

Free-space optical networks have emerged as a viable technology for broadband wireless backbone networks of the next generation. In this paper, we investigate the challenging problem of joint topology design and load balancing in FSO networks. We consider FSO link characteristics, cost constraints, traffic characteristics, traffic demand, and QoS requirements in the formulation, along with various objective functions including network-wide average load and delay. We apply the reformulation–linearization technique (RLT) to obtain linear programming (LP) relaxations of the original complex problem, and then incorporate the LP relaxations into a branch-and-bound framework. The proposed algorithm can produce highly competitive solutions with the performance guarantees in the form of bounded optimality gap. For reducing computation complexity, we also develop a fast heuristic algorithm to provide highly competitive solutions. The heuristic algorithm iteratively perturbs the current topology and computes network flows for the new topology, thus progressively improving the configuration and load balancing of the FSO network. The proposed algorithms are complementary to each other, since jointly applying the algorithms can make the FSO network dynamically adaptive to events occurring at both large and small timescales. The proposed algorithms are evaluated with extensive simulations. Our simulation results show that the heuristic algorithm can achieve an optimality gap close to that of the branch-and-bound algorithm, with significantly reduced computation time.

1. Introduction

Free-space optics (FSO) have emerged as a promising technology for broadband wireless networks of the next generation [3]. FSOs are wireless systems that use free space as transmission medium to transmit optical data signals at high bit rates. FSOs have many advantages such as cost effectiveness, long transmission range, free license, interference immunity, and high-bandwidth, among others. In recent years, considerable advances have been made in understanding the FSO channel, and both experimental data and commercial FSO transceivers are now available [3]. For the widespread deployment of FSO networks, several important network problems should be addressed, such as how to design an FSO network topology with rich connectivity (making it robust to link failures) and how to accommodate traffic demands and QoS requirements of the underlying wired or wireless access network.

The FSO network topology design problem has been addressed in several prior works. In [4], a distributed minimum spanning tree (MST) algorithm was proposed

☆ Part of this work was conducted when In Keun Son was pursuing a doctoral degree at Auburn University. This work was presented in part at IEEE GLOBECOM 2010, Miami, FL, December 2010 [1,2].

* Corresponding author. Tel.: +1 334 844 1845; fax: +1 334 844 1809.
E-mail addresses: soninkeun@gmail.com (I.K. Son), smao@ieee.org (S. Mao), das@cse.uta.edu (S.K. Das).

1573-4277/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.osn.2013.08.001

to build degree-bounded tree topologies. In [5,6], algorithms were developed to maximize network connectivity and make a mesh topology. The load balancing problem was addressed in [7,8], where several topology design heuristics were developed to minimize network-wide average load. To maximize the potential of FSO networks, the unique characteristics of FSO links should be considered in the formulation, and the problems of topology design and routing of the traffic flows should be jointly considered and optimized [7,8]. Usually such problems are highly complex. Consequently, an approximation algorithm with the performance guarantees (i.e., in the form of a bounded *optimality gap*) would be highly appealing.

An FSO link is "pseudo-wired" in the sense that it has high bandwidth, narrow beam, and long distance as an optical fiber link. However, it is also like a radio frequency (RF) link that is flexibly steerable, in contrast to buried optical fiber links. Thus the network topology can be adaptively reconfigured on-the-fly in response to network dynamics. In this paper, we investigate the problem of joint topology design and load balancing in FSO networks, considering important design issues such as link reliability, cost constraints, traffic characteristics and demand, QoS requirements, routing policies, and network topology. We assume that a traffic matrix is known and the total number of edges used for building a topology is given a priori (due to some cost constraint). We then formulate the joint topology design and load balancing problem with objectives to minimize network-wide average load or network-wide average delay. Since for the same offered load, different traffic models will yield very different delay performance, we consider both short-range dependent (SRD) and long-range dependent (LRD) traffic models when deriving network delays.

With the objective function of average load, the formulated problem is a mixed integer linear programming (MILP) problem. With the objective function of an average delay, the formulated problem is a mixed integer nonlinear programming (MINLP) problem. These problems are NP-hard in general [7,8]. In prior work [7,8], effective heuristic algorithms are presented to minimize the network-wide average load. However, there was no guarantee on the optimality performance of the heuristic algorithms, and they do not apply to the more complex problem of minimizing network-wide average delay.

In this paper, we first develop a branch-and-bound algorithm incorporating the reformulation–linearization technique (RLT) that can produce highly competitive solutions with bounded performance. RLT is a useful technique that can be applied to derive linear programming (LP) relaxations for an underlying nonlinear non-polynomial programming problem [9]. We first adopt RLT to obtain LP relaxations for the complex MILP and MINLP problems. We then incorporate the LP relaxations into the branch-and-bound framework to compute (1-ϵ)-optimal solutions, where $0 \leq \epsilon \ll 1$ is a prescribed tolerance. When the algorithm terminates, it produces a feasible solution to the original MILP or MINLP problem, which is within the ϵ range of the global optimum.

Although highly appealing, the RLT-based branch-and-bound algorithm has a relatively high computation complexity. We next present a fast heuristic algorithm to the joint topology design and load balancing problem. The heuristic algorithm consists of three components: (i) initial topology design, (ii) multipath routing for load balancing, and (iii) topology perturbation. Starting from an initial topology that is designed to minimize the network-wide average load, the heuristic algorithm iteratively perturbs the current topology and computes network flows for the new topology, thus progressively improving the configuration and load balancing of the FSO network.

The proposed algorithms are evaluated with extensive simulations, and are shown to be highly suitable for joint topology and load balancing optimization in FSO networks. Our simulation results also show that the heuristic algorithm can achieve an optimality gap close to that of the branch-and-bound algorithm, but with significantly reduced computation time.

The proposed fast heuristic algorithm and the branch-and-bound algorithm are complementary to each other. The latter is suitable for optimizing the FSO network design and operation at large timescales with guaranteed optimality. The former is suitable for dynamic reconfiguration of the FSO network in response to small timescale events. We envision that the branch-and-bound algorithm will be executed at relatively large time intervals when significant changes occur in the FSO network, while the heuristic algorithm will be kept running to continuously optimize the operation and configuration of the FSO network in response to small timescale events such as bad weather conditions or fluctuations in traffic demand.

The remainder of this paper is organized as follows. In Section 2, we describe the system model and assumptions, and formulate the optimization problem. The RLT-based algorithm is presented in Section 3 and the fast heuristic algorithm is presented in Section 4. The algorithms are evaluated in Section 5. We discuss related work in Section 6, and Section 7 concludes this paper.

2. System model and problem statement

2.1. Network model

We consider an FSO network consisting of n base stations (BS), which provide mobile users with network access. Each BS could be the head of a cluster consisting of multiple access points. The aggregate traffic at the BS's will be relayed through wireless optical links. We assume that each BS has multiple sets of wireless optical devices in order to support the aggregate traffic load and provide a rich mesh connectivity. The FSO links are point-to-point connections with narrow beam divergence, and are immune to electromagnetic interference [3,6–8].

The FSO network can be modeled as a *simple* graph $G(V, E)$, where each vertex $v \in V$ represents a BS and each edge $e \in E$ is an FSO link. Let n and m denote the cardinality of V and E, respectively. We assume an $n \times n$ *traffic matrix* $\mathbf{F}$ that describes the traffic demand (measured, estimated, or projected) for the access network, where each element $f_{sd} = [\mathbf{F}]_{sd}$ represents the mean data rate between each source and destination BS pair s–d.

We characterize each FSO link $e=(i,j)\in E$ with two parameters: (i) link capacity c_{ij}; and (ii) link reliability γ_{ij}. As in prior work [8], we assume that each FSO channel is full duplex with symmetric capacity and a nominal data rate c is achievable within a predefined transmission range, i.e., $c_{ij}=c_{ji}=c$ for all $i\neq j$. We also assume symmetric link reliability, i.e., $\gamma_{ij}=\gamma_{ji}$ for all $i\neq j$, due to the line-of-sight transmissions with narrow beam divergence. There is connectivity between two BS's if the link reliability is larger than a threshold γ_{th}.

2.2. FSO channel model

We adopt the *log-normal* model to characterize FSO link reliability under turbulent atmosphere [10,11]. The marginal distribution of light intensity fading induced by atmospheric turbulence can be statistically modeled as [10]

$$f_I(I)=\frac{1}{2\sigma_X I\sqrt{2\pi}}\exp\left\{-\frac{[\ln(I)-\ln(I_0)]^2}{8\sigma_X^2}\right\}, \tag{1}$$

where σ_X^2 is the variance and I_0 is the received average intensity without turbulence. The standard deviation σ_X^2 can be approximated as $\sigma_X^2=0.30545(2\pi/\lambda)^{7/6}C_n^2(L)z^{11/6}$, where λ is the wavelength, $C_n^2(L)$ is the *index of refraction structure parameter* with constant altitude L, and z is the distance. It is shown that for atmospheric channels near the ground (i.e., $L<18.5$ m), C_n^2 ranges from 10^{-13} $m^{-2/3}$ to 10^{-17} $m^{-2/3}$ for strong and weak atmospheric turbulences, respectively. The common average is 10^{-15} $m^{-2/3}$.

The link reliability γ_{ij} is the probability that the intensity of received signal I exceeds a threshold I_{th}, which can be computed using the *error function* erf$(\cdot)$ as

$$\gamma_{ij}=P(I\geq I_{th})=\frac{1}{2}-\frac{1}{2}\mathrm{erf}\left(\frac{\ln(I_{th}/I_0)}{2\sqrt{2}\sigma_X}\right). \tag{2}$$

The ratio I_{th}/I_0 can be interpreted as *transmittance* according to Beer–Lambert Law [12], which is determined by the distance and absorption coefficient. For a fixed ratio I_{th}/I_0, γ_{ij} depends on the standard deviation σ_X, which is strongly influenced by weather conditions and transmission distance [13]. We assume that the set of edges satisfying $\gamma_{ij}\geq\gamma_{th}$ forms the *candidate link set* for constructing the FSO network topology.

2.3. Performance measures

2.3.1. Network-wide average load (L)

We adopt multipath routing for load balancing, where a flow f_{sd} may be split into multiple subflows. Let f_{ij}^{sd} be the subflow of f_{sd} passing through a link (i,j). We have the following flow-conservation condition:

$$\sum_{j=1}^{n}f_{ij}^{sd}-\sum_{j=1}^{n}f_{ji}^{sd}=\begin{cases}f_{sd}, & i=s,\ \forall i\in V\\ -f_{sd}, & i=d,\ \forall i\in V\\ 0 & \text{otherwise } \forall i\in V.\end{cases} \tag{3}$$

Considering all the s–d pairs, the average traffic load λ_{ij} and the link utilization ρ_{ij} are

$$\lambda_{ij}=\sum_{s,d\in V}f_{ij}^{sd}\quad\text{and}\quad\rho_{ij}=\lambda_{ij}/c_{ij}<1. \tag{4}$$

For the link to be stable, we have $\rho_{ij}<1$, for all $(i,j)\in E$. We define the *network-wide average load* L as

$$L\cong(1/\lambda)\cdot\sum_{(i,j)\in E}\lambda_{ij}, \tag{5}$$

where $\lambda=\sum_{s,d\in V}f_{sd}$ is the sum of all traffic demands. Note that when a packet is forwarded, it is counted multiple times in L. When all the s–d traffic are transmitted through direct links, L achieves its minimum value 1.

2.3.2. Network-wide average delay (T_1, T_2)

We model each link $(i,j)\in E$ as a general queueing system with an average input rate λ_{ij} and a service capacity c_{ij}. The average delay incurred at the link depends on the traffic auto-correlation structure. When the traffic constantly exhibits *short-range dependent* (SRD) characteristics (e.g., voice over IP (VoIP) traffic), we can model the link delay with an exponential distribution with parameter $c_{ij}-\lambda_{ij}$ [14]. Applying Little's formula, the network-wide average delay T_1 can be computed as

$$T_1\cong\sum_{(i,j)\in E}\left(\frac{1}{\lambda}\right)\left(\frac{\lambda_{ij}}{c_{ij}-\lambda_{ij}}\right). \tag{6}$$

When the traffic exhibits *long-range dependent* (LRD) characteristics (e.g., computer data or variable-bit-rate video traffic), we can model each link as a *fractional Brownian motion* (fBm) queueing system, where the queue length has a heavy-tailed *Weibull* distribution [15], i.e.,

$$\Pr\{Q_{ij}>q\}\approx\exp\left\{-\frac{(c_{ij}-\lambda_{ij})^{2H}}{2\kappa^2(H)a\lambda_{ij}}q^{2-2H}\right\}, \tag{7}$$

where $\kappa(H)=H^H(1-H)^{1-H}$, $H\in[0.5,1)$ is the Hurst parameter, and a is the index of dispersion. Applying Little's formula, the network-wide average delay T_2 is

$$T_2\cong\frac{1}{\lambda}\Gamma\left(1+\frac{1}{2-2H}\right)\sum_{(i,j)\in E}\left[\frac{(c_{ij}-\lambda_{ij})^{2H}}{2\kappa^2(H)a\lambda_{ij}}\right]^{-1/(2-2H)}. \tag{8}$$

Defining $\tau=\frac{1}{\lambda}\Gamma(1+(1/(2-2H)))[2\kappa^2(H)a]^{1/(2-2H)}$, we have

$$T_2=\tau\cdot\sum_{(i,j)\in E}\left[\frac{\lambda_{ij}}{(c-\lambda_{ij})^{2H}}\right]^{1/(2-2H)}. \tag{9}$$

2.4. Problem statement

Without the loss of generality, we consider the case of fixed BS's. The atmospheric condition can be known through weather forecast and past experiences. Then we can evaluate the edge reliabilities and determine the candidate edge set $\mathcal{E}_c$. The number of links in the FSO network is upper bounded by $m_c=|\mathcal{E}_c|$. The number of links is also lower bounded by $n-1$, the minimum number of links that is needed to construct a connected network (i.e, a minimum spanning tree).

The problem is to select m links from m_c candidate edges to form a mesh topology. In addition, we also determine multipath routing for the s–d flows, such that either the network-wide average load L or the network-wide average delay T_1 or T_2 is minimized. Define the

following index variables for each link $(i,j) \in V$ as:

$$x_{ij} = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{E}_c \\ 0 & \text{otherwise,} \end{cases}, \quad y_{ij} = \begin{cases} 1 & \text{if } (i,j) \text{ is chosen} \\ 0 & \text{otherwise.} \end{cases}$$

We have $x_{ij} \geq y_{ij}$ for all $(i,j) \in V$. The problem of joint topology design and load balancing, denoted as Problem OPT-TDLB, can be formulated as follows:

minimize $L/T_1/T_2$ (10)

subject to

$$\sum_{i=1}^{n}\sum_{j=i}^{n} x_{ij} = m_c, \sum_{i=1}^{n}\sum_{j=i}^{n} y_{ij} = m, \quad \text{for } i,j \in V \tag{11}$$

$$y_{ij} \leq x_{ij}, \quad x_{ij} = x_{ji}, \quad y_{ij} = y_{ji} \text{ for } i,j \in V \tag{12}$$

$$\theta_i^l \leq \sum_{j=1}^{n} y_{ij} \leq \theta_i^u \quad \text{for } i \in V \tag{13}$$

$$0 \leq f_{ij}^{sd} \leq y_{ij} f_{sd} \quad \text{for } i,j,s,d \in V \tag{14}$$

$$\lambda_{ij} = \sum_{s,d \in V} f_{ij}^{sd} \leq c_{ij} \quad \text{for } (i,j) \in \mathcal{E}_c \tag{15}$$

$$\text{flow conservation constraint} \quad (3) \tag{16}$$

$$x_{ij} \in \{0,1\}, \quad y_{ij} \in \{0,1\} \quad \text{for } i,j \in V. \tag{17}$$

In Problem OPT-TDLB, the optimization variables include binary variables $\mathbf{y} = \{y_{ij} | \forall i,j \in V\}$ and continuous variables $\mathbf{f} = \{f_{ij}^{sd} | \forall i,j,s,d \in V\}$. With objective function L, Problem OPT-TDLB(L) is an MILP problem. With objective function T_1 or T_2, the corresponding Problems OPT-TDLB (T_1) and OPT-TDLB(T_2) are MINLP problems. Constraints (11)–(13) are for edge selection and topology design, while constraints (14)–(16) are for multipath routing and load balancing. In (13), θ_i^u is the maximum degree (enforced by some cost constraints), and θ_i^l is the minimum degree for BS i, which is required to support the incoming and outgoing traffic from the BS and can be estimated as

$$\theta_i^l = \max\left\{ \left\lceil \sum_{d=1}^{n} \lambda_{id}/c \right\rceil, \left\lceil \sum_{s=1}^{n} \lambda_{si}/c \right\rceil \right\}. \tag{18}$$

3. RLT-based branch-and-bound algorithm

Our solution procedure for Problem OPT-TDLB is to incorporate an LP relaxation of the original problem into a branch-and-bound framework [9]. Problem OPT-TDLB(L) is an MILP. We can obtain its LP relaxation by allowing the binary variables y_{ij} to take real values in [0, 1]. In this section, we reformulate and linearize MINLP Problems OPT-TDLB(T_1) and OPT-TDLB(T_2). The LP relaxations will then be incorporated into a branch-and-bound algorithm that can compute $(1-\epsilon)$-optimal solutions.

3.1. Reformulation and linearization: OPT-TDLB(T_1)

To linearize the objective function T_1, we define substitution variables t_{ij}' as $t_{ij}' = \lambda_{ij}/(c_{ij} - \lambda_{ij})$. Then we obtain a linear objective function $\sum t_{ij}'$ and additional nonlinear constraints $t_{ij}' \cdot c_{ij} - t_{ij}' \cdot \lambda_{ij} - \lambda_{ij} = 0$.

We next linearize the nonlinear constraint by defining substitution variables for the quadratic term $\mu_{ij} = t_{ij}' \cdot \lambda_{ij}$. Substituting μ_{ij} into the additional nonlinear constraints, we make them linear, but with the additional *RLT bound-factor product constraints* for the new variable μ_{ij}. Since t_{ij}' and λ_{ij} are bounded by their respective lower and upper bounds as $0 \leq t_{ij}' \leq \bar{t}'$ and $0 \leq \lambda_{ij} \leq c_{ij}$, we have

$$\begin{cases} (t_{ij}' - 0) \cdot (\lambda_{ij} - 0) \geq 0 \\ (t_{ij}' - 0) \cdot (c_{ij} - \lambda_{ij}) \geq 0 \\ (\bar{t}' - t_{ij}') \cdot (\lambda_{ij} - 0) \geq 0 \\ (\bar{t}' - t_{ij}') \cdot (c_{ij} - \lambda_{ij}) \geq 0. \end{cases} \tag{19}$$

Expanding (19) and substituting $\mu_{ij} = t_{ij}' \cdot \lambda_{ij}$, we obtain the following RLT bound-factor product constraints:

$$\begin{cases} \mu_{ij} \geq 0 \\ c_{ij} \cdot t_{ij}' - \mu_{ij} \geq 0 \\ \bar{t}' \cdot \lambda_{ij} - \mu_{ij} \geq 0 \\ \bar{t}' \cdot c_{ij} - c_{ij} \cdot t_{ij}' - \bar{t}' \cdot \lambda_{ij} + \mu_{ij} \geq 0. \end{cases} \tag{20}$$

We then obtain an LP relaxation l-OPT-TDLB(T_1) as

minimize $(1/\lambda) \cdot \sum_{(i,j) \in E} t_{ij}'$ (21)

subject to

$$\text{constraints} \quad (11)\text{–}(16) \tag{22}$$

$$0 \leq y_{ij} \leq 1 \quad \text{for } (i,j) \in \mathcal{E}_c \tag{23}$$

$$c_{ij} \cdot t_{ij}' - \mu_{ij} - \lambda_{ij} = 0 \quad \text{for } (i,j) \in \mathcal{E}_c \tag{24}$$

$$\text{RLT bound-factor constraints (20) for } (i,j) \in \mathcal{E}_c. \tag{25}$$

3.2. Reformulation and linearization: OPT-TDLB(T_2)

The objective function T_2 consists of exponents. We define substitution variables for the exponents as $t_{ij}'' = [\lambda_{ij}/(c_{ij} - \lambda_{ij})^{2H}]^{1/(2-2H)}$. Substituting t_{ij}'' we obtain a linear objective function $\tau \sum t_{ij}''$, with additional nonlinear constraints: $t_{ij}'' \cdot (c_{ij} - \lambda_{ij})^{H/(1-H)} - \lambda_{ij}^{1/(2-2H)} = 0$. Letting $\nu_{ij} = c - \lambda_{ij}$, we have $t_{ij}'' \cdot (\nu_{ij})^{H/(1-H)} - \lambda_{ij}^{1/(2-2H)} = 0$. Finally, taking logarithms and defining substituting variables

$$\zeta_{t_{ij}'} = \log\,(t_{ij}''), \quad \zeta_{\lambda_{ij}} = \log\,(\lambda_{ij}), \quad \zeta_{\nu_{ij}} = \log\,(\nu_{ij}), \tag{26}$$

we obtain linear constraints

$$\zeta_{t_{ij}'} - \frac{1}{2-2H}\zeta_{\lambda_{ij}} + \frac{H}{1-H}\zeta_{\nu_{ij}} = 0.$$

We still need to linearize the new nonlinear constraints (26), which are all in the form of $y = \log\,(x)$. If x is bounded by $0 < x_l \leq x \leq x_u$, the logarithm relationship can be linearized using a *polyhedral outer approximation* comprises of a convex envelop in concert with several tangential

supports [9], i.e.,

$$\begin{cases} y \geq \dfrac{\log\,(x_u)-\log\,(x_l)}{x_u-x_l}\cdot(x-x_u)+\log\,(x_u) \\ y \leq x/x_k+\log\,(x_k)-1, \end{cases} \tag{27}$$

where $x_k = x_l + (k/(k_{max}-1))(x_u-x_l)$ for $k=0,\ldots,k_{max}-1$. The convex envelope consists of a chord connecting two end points and k_{max} supports each being tangent to the log (x) curve at x_k. Therefore, we generate $(k_{max}+1)$ new linear constraints for each logarithmic substitution variables in (26). Fig. 1 shows an example with the four-point tangential approximation.

Note that using a larger k_{max} can reduce the area above the log (x) curve. However, the reduction will not be significant since the area above the log (x) curve is much smaller than the area below the log (x) curve. However, using a larger k_{max} will produce more linear constraints in the relaxed problem, and cause the LP solver more time to solve the relaxed problem. Since the polyhedral outer approximation will be incorporated in the branch-and-bound framework, the infeasible area around the log (x) curve will be iteratively reduced as the range $[x_1, x_k]$ is iteratively reduced. Our experience show that a small k_{max} of 3 or 4 would suffice.

We thus obtain an LP relaxation l-OPT-TDLB(T_2) as

$$\textbf{minimize} \quad \tau \cdot \sum_{(i,j)\in E} t''_{ij} \tag{28}$$

subject to

$$\text{constraints} \quad (11)\text{–}(16) \tag{29}$$

$$0 \leq y_{ij} \leq 1 \quad \text{for } (i,j)\in\mathcal{E}_c \tag{30}$$

$$\zeta_{t'_{ij}} - \frac{1}{2-2H}\zeta_{\lambda_{ij}} + \frac{H}{1-H}\zeta_{\nu_{ij}} = 0 \quad \text{for } (i,j)\in\mathcal{E}_c \tag{31}$$

$$\nu_{ij} + \lambda_{ij} = c_{ij} \quad \text{for } (i,j)\in\mathcal{E}_c \tag{32}$$

$$\text{polyhedral outer approximations (27)} \quad \text{for } \zeta_{t'_{ij}}, \zeta_{\lambda_{ij}}, \zeta_{\nu_{ij}} \quad \text{given in (26) for } (i,j)\in\mathcal{E}_c. \tag{33}$$

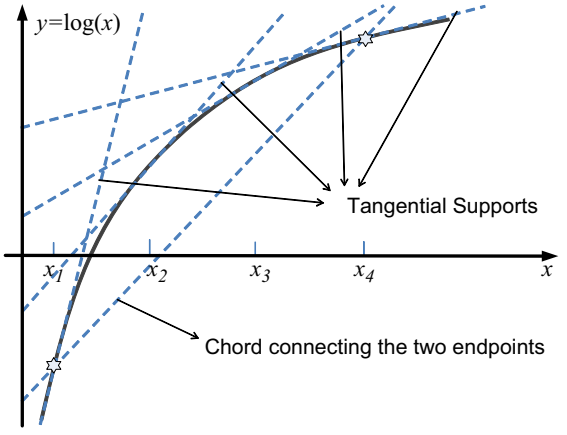


Fig. 1. An example of polyhedral outer approximation for $y=\log(x)$ where $0<x_l\leq x\leq x_u$ and $k_{max}=4$. The two endpoints are marked by the stars (corresponding to $x=x_1$ and $x=x_4$).

3.3. Branch-and-bound algorithm

In this section, we embedded the LP relaxations into the branch-and-bound framework to obtain solutions with bounded optimality gap. Branch-and-bound is an iterative optimization algorithm that is especially useful for solving discrete and combinatorial problems. It consists of two key components: (i) a strategy to split a problem into subproblems with smaller sizes, i.e., *branching*, and (ii) a fast way to obtain lower and upper bounds (LB, UB) for the subproblems, i.e., *bounding*. The resulting subproblems form a tree structure, while the set of leaf nodes is called the *problem list* $\mathcal{P}$. During the solution process, a branch of the tree may be deleted (or, *fathomed*) from future search, if all its solutions are dominated by some other subproblems, therefore reducing the computational cost.

Our RLT-based branch-and-bound algorithm is given in Algorithm 1 and the flow chart is given in Fig. 2. Solving the LP relaxation l-OPT-TDLB with an LP solver, we can obtain a solution $\hat{\delta}=(\hat{\mathbf{y}},\hat{\mathbf{f}})$, which is optimal to the LP relaxation. Note the relaxation is actually obtained by relaxing the constraints (i.e., augmenting the search region of the original problem). On one hand, the LP solution $\hat{\delta}$ may lie outside the feasible region of the original problem (e.g., a fractional x_{ij}, rather than binary) thus being infeasible to the original problem; on the other hand, if we substitute this possibly infeasible solution $\hat{\delta}$ into the objective function, we obtain an objective value that is a lower bound (i.e., LB) of the original problem. Then we apply a *local search* algorithm to derive a feasible solution δ in the neighborhood of $\hat{\delta}$, which provides an UB for the original problem. The global LB and UB are updated as follows:

$$\begin{cases} \text{LB} = \min\{\text{LB}_h : \text{all problems } h\in\mathcal{P}\} \\ \text{UB} = \min\{\text{UB}_h : \text{all problems } h\in\mathcal{P}\}. \end{cases} \tag{34}$$

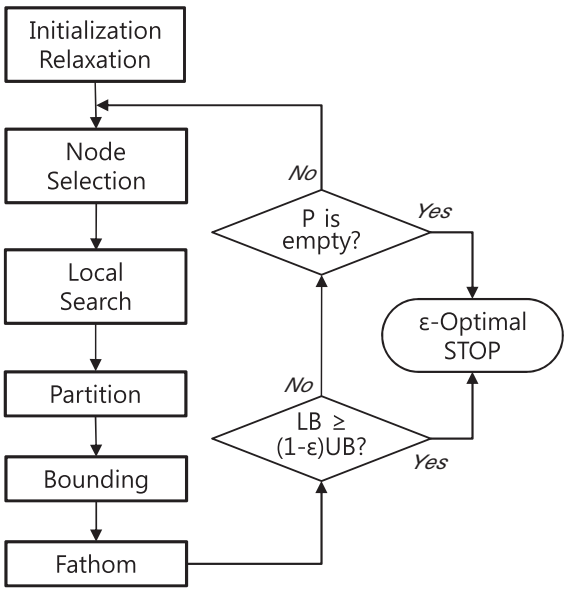


Fig. 2. Flowchart of the RLT-based branch and bound algorithm.

Algorithm 1. Branch-and-bound algorithm for Problem OPT-TDLB.

// Initialization
Initialize $\delta^* = \varnothing$ and $\mathrm{UB} = \infty$;
Initialize problem list $\mathcal{P}$ with the original Problem 1;
// Relaxation
Obtain l-OPT-TDLB for Problem 1;
Solve l-OPT-TDLB to obtain solution $\hat{\delta} = (\hat{\mathbf{y}}, \hat{\mathbf{f}})$ and the objective value as the lower bound LB_1;
Select problem h with the minimum LB_h from $\mathcal{P}$;
Set $\mathrm{LB} = \mathrm{LB}_h$ and let $\hat{\delta}$ be its relaxed solution;
// Local Search
Obtain feasible solution δ from $\hat{\delta}$ with the local search algorithm;
Compute UB_h from $(\mathbf{y}, \mathbf{f})$;
if $(\mathrm{UB}_h < \mathrm{UB})$ **then**
  Update $\delta^* = \delta$ and $\mathrm{UB} = \mathrm{UB}_h$;
  if $(\mathrm{LB} \geq (1-\epsilon) \cdot \mathrm{UB})$ **then**
    Stop with $(1-\epsilon)$−solution δ^*;
  else
    Discard all problemsh' from $\mathcal{P}$ satisfying$\mathrm{LB}_h \geq (1-\epsilon) \cdot \mathrm{UB}$;
  end
end
// Partition
Find $\tilde{y}_{ij}$ that is closest to 1 but not fixed yet;
Split problem h into h_1 and h_2 with respect to $\tilde{y}_{ij}$;
// Bounding
Solve the RLT relaxations of the two subproblems and obtain their lower bounds LB_{h_1} and LB_{h_2};
Remove problem h from $\mathcal{P}$;
if $(\mathrm{LB}_{h_1} < (1-\epsilon) \cdot \mathrm{UB})$ **then**
  Insert problem h_2 into $\mathcal{P}$;
end
if $(\mathrm{LB}_{h_2} < (1-\epsilon) \cdot \mathrm{UB})$ **then**
  Insert problem h_1 into $\mathcal{P}$;
end
if $(\mathcal{P} = \varnothing)$ **then**
  Stop with the current best solution, δ^*;
else
  Go to 5;
end

We adopt a simple local search algorithm as follows. First, we determine the network topology by fixing the y_{ij}'s to binaries. According to constraint (11), m links should be selected from the candidate set $\mathcal{E}_c$ to form a topology. Hence, we choose the m largest $\hat{y}_{ij}$'s in $\mathbf{y}$ and set them to 1; the rest smaller $\hat{y}_{ij}$'s are set to 0. When the topology is fixed, we then determine the optimal multipath routing by solving Problem l-OPT-TDLB again for the optimal f_{ij}^{sd}'s. The resulting solution $\delta = (\mathbf{y}, \mathbf{f})$ is thus feasible and provides an upper bound.

For branching, we choose the subproblem h with the smallest LB_h, which is indicative of the global optimal solutions. Subproblem h is then partitioned into two subproblems, h_1 and h_2, which replace the subproblem h in $\mathcal{P}$. The corresponding LB_{h_1}, LB_{h_2}, UB_{h_1}, and UB_{h_2} are calculated, and the global LB and UB are updated as in (34). The iteration procedure terminates when there is a feasible solution satisfying $\mathrm{LB} \geq (1-\epsilon) \cdot \mathrm{UB}$, or when the problem list $\mathcal{P}$ is empty.

For this minimization problem, the UB is from a feasible solution and the LB is from a possibly infeasible solution. If the global optimal solution is T^*, then we have $T^* \geq \mathrm{LB} \geq (1-\epsilon)\mathrm{UB}$, which leading to

$$\mathrm{UB} \leq (1 + \epsilon + o(\epsilon))T^*. \tag{35}$$

That is, the feasible solution produced by the proposed scheme will be within the ϵ−range of the global optimal.

4. Fast heuristic algorithm

Algorithm 2. Heuristic algorithm for Problem OPT-TDLB.

// Initialization
Read $\mathcal{N}$, $\mathbf{F}$, and C_n^2;
Derive the candidate edge matrix $\mathbf{X}$;
Derive the minimum multihop matrix $\mathbf{H}$;
Derive the minimum traffic demand matrix $\mathbf{F}'$;
// Select edges to form adjacency matrix $\mathbf{Y}(\subset \mathbf{X})$
Choose the largest unselected $f_{sd}^{'}$;
Find k multiple shortest paths for s–d;
Insert unselected edges in the shortest paths into $\mathbf{Y}$;
if (*m edges have been selected*) **then**
  Break with $\mathbf{Y}$;
else
  Go to Step5;
end
Solve l-OPT-TDLB($L/T_1/T_2$) for $\mathbf{f}$ with topology $\mathbf{Y}$;
// Perturb the current topology
Execute the branch exchange algorithm to get $\hat{\mathbf{Y}}$;
  Delete the link with the smallest load/delay;
  Connect the two nodes with the largest load/delay;
if (*no further branch exchange*) **then**
  Terminate with$(\mathbf{Y}, \mathbf{f})$;
end
// Multipath routing for load balancing
Solve l-OPT-TDLB($L/T_1/T_2$) for $\hat{\mathbf{f}}$ with topology $\hat{\mathbf{Y}}$;
if $((\hat{\mathbf{Y}}, \hat{\mathbf{f}}) \prec (\mathbf{Y}, \mathbf{f}))$**then**
  $(\mathbf{Y}, \mathbf{f}) = (\hat{\mathbf{Y}}, \hat{\mathbf{f}})$;
end
if (*less than maximum number of iterations*) **then**
  Go to Step 14;
else
  Terminate with $(\mathbf{Y}, \mathbf{f})$;
end

4.1. Overview

In this section, we present a fast heuristic algorithm for Problem OPT-TDLB. In Section 3, we observe that joint optimization with binary variables y_{ij}'s and continuous variables f_{ij}^{sd}'s requires long execution time. To speed up computation, we first determine the network topology to minimize average network load, and then solve the multipath routing problem for load balancing. To further improve the optimality, we iteratively perturb the topology and then compute new network flows $\mathbf{f}$, until some termination criterion is met.

The pseudo-code of the algorithm is given in Algorithm 2. As shown in the flowchart in Fig. 3, it consists of three parts: (i) initial topology design, (ii) multipath routing for load balancing, and (iii) topology perturbation. The algorithm iteratively perturbs the current topology by deleting and inserting FSO links, and computes the network flow for the s–d pairs based on the new topology. The network flow is derived by solving the LP relaxations l-OPT-TDLB ($L/T_1/T_2$) with fixed y_{ij} values, thus the computation is very fast. If the new topology and network flow produce a

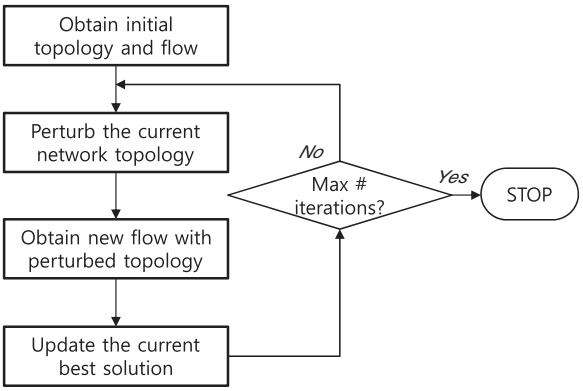


Fig. 3. Flowchart of the fast heuristic algorithm.

better objective value for the original problem, the new solution will replace the existing one. Thus the objective function will be progressively improved over iterations. The algorithm terminates when no further perturbation can be made or when a prescribed maximum number of iterations is reached. The three key components of the algorithm are described in detail in the remainder of this section.

4.2. Initial topology design

As discussed, the proposed algorithm iteratively improves the topology and network flow starting from an initial topology. A properly designed initial topology could speed up the convergence and achieve better solutions. In the following, we show how to choose m links from the candidate set $\mathcal{E}_c$ to form an initial topology. The initial topology is represented by an *adjacency matrix* $\mathbf{Y}=[y_{ij}]$. $\mathbf{Y}$ is a $0-1$ $n\times n$ matrix, where $y_{ij}=1$ if and only if link $(i,j)\in\mathcal{E}_c$ is included in the topology.

4.2.1. Prerequisite information

Assume the traffic matrix $\mathbf{F}$ is known. Let $\mathcal{N}$ denote node information, such as node ID and location, and C_n^2 represent FSO channel information. Then we can derive the reliability of the FSO links as in (2), and the set of candidate links $\mathcal{E}_c$. If $\gamma_{ij}\geq\gamma_{th}$, we have $x_{ij}=1$ and link $(i,j)\in\mathcal{E}_c$; otherwise $x_{ij}=0$. The candidate links form a topology with adjacency matrix $\mathbf{X}=[x_{ij}]$. The total number of candidate links is $m_c=|\mathcal{E}_c|=\sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}/2$. The minimum degree of each node can be derived as in (18) using traffic matrix $\mathbf{F}$ and link capacity c.

4.2.2. Minimum traffic matrix $\mathbf{F}'$

For a given topology, the traffic matrix $\mathbf{F}$ is usually supported with both single-hop and multihop flows. If nodes s and d are not one-hop neighbors, the flow f_{sd} will be forwarded multiple times through multihop paths. The network traffic load associated with f_{sd} is thus proportional to the hop-counts of the s–d paths. We aim to have small hop-counts for s–d pairs with high traffic demands when determining the initial topology.

We consider both single-hop and multihop traffic patterns by constructing a *minimum traffic matrix*, denoted as $\mathbf{F}'$. Assume K disjoint paths $\{P_{sd}^k\}_{k=1,\ldots,K}$ between nodes s and d. The s–d traffic subflow on path P_{sd}^k is denoted as f_{sd}^k, and the length (or, hop-count) of P_{sd}^k is denoted as l_{sd}^k, for $1\leq i\leq K$. We have $f_{sd}=\sum_{k=1}^{K}f_{sd}^k$. For the total traffic load associated with flow f_{sd}, denoted as $\lambda_{sd}^{'}$, we have

$$\lambda_{sd}^{'}=\sum_{k=1}^{K}l_{sd}^k\cdot f_{sd}^k \geq \min\{l_{sd}^k|k=1,2,\ldots,K\}\cdot f_{sd}\cong f_{sd}^{'}. \quad (36)$$

We next introduce a technique to compute $\min_k\{l_{sd}^k\}$, which is based on the adjacency matrix $\mathbf{X}$ defined earlier. We have the following from graph theory [16].

Fact 1. Let $\mathbf{A}$ be the adjacency matrix of graph G. The number of *walks* from vertex s to d in G with length l is $[\mathbf{A}^l]_{sd}$.

From Fact 1, the hop-count of the shortest path between nodes s and d can be found by identifying the smallest l, such that $[\mathbf{A}^l]_{sd}>0$. That is,

$$\min_k\{l_{sd}^k\}=l \quad \text{if } [\mathbf{A}^l]_{sd}>0 \quad \text{and} \quad [\mathbf{A}^h]_{sd}=0, \text{ for all } 1\leq h<l. \quad (37)$$

Once the hop-counts of the shortest paths are obtained, we can derive the minimum traffic matrix $\mathbf{F}'$, with elements

$$f_{sd}^{'}=\min_k\{l_{sd}^k\}\cdot f_{sd}. \quad (38)$$

4.2.3. Link selection

When the minimum traffic matrix is obtained, we next construct the initial topology based on $\mathbf{F}'$ and the adjacency matrix $\mathbf{X}$ representing all the candidate links. To minimize the network-wide average load L, we choose the links that are on the shortest paths for all s–d pairs. We examine the elements of $\mathbf{F}'$ in nonincreasing order, starting from the largest one, and choose links from $\mathcal{E}_c$ (or, elements in $\mathbf{X}$) to insert into $\mathbf{Y}$ until m edges are selected. If two nodes are adjacent according to $\mathbf{X}$, the direct link will be inserted. In the case of multihop flows, multiple paths will be selected for rich connectivity and load balancing.

4.3. Multipath routing for load balancing

When topology $\mathbf{Y}$ is fixed, the binary optimization variables $\mathbf{y}$ are all determined. The LP relaxations l-OPT-TDLB($L/T_1/T_2$) now only have continuous subflow variables $\mathbf{f}$. We then solve Problem l-OPT-TDLB($L/T_1/T_2$) again with all the y_{ij}'s determined, to obtain multipath routing of the s–d flows $\mathbf{f}$ for this given topology.

To obtain the LP relaxations l-OPT-TDLB($L/T_1/T_2$), the following two types of constraints are relaxed: (i) the binary variables y_{ij} are allowed to take real values in [0, 1], and (ii) the nonlinear objective functions T_1 and T_2 are linearized with additional RLT bound factor constraints and polyhedral out approximation constraints [9]. Except for these two, all the other constraints (11)–(16) are preserved during the procedure. When the y_{ij}'s are all fixed, the $\mathbf{f}$ solved from the LP relaxations are also feasible to the original problem OPT-TDLB($L/T_1/T_2$). That is, the solution $\{\mathbf{y},\mathbf{f}\}$ is feasible to both the LP relaxations and the

original problem. Substituting it into the objective function of OPT-TDLB($L/T_1/T_2$) yields an upper bound.

Clearly the optimality of the feasible solution $\{\mathbf{y}, \mathbf{f}\}$ depends on the topology from which the solution is computed. It is possible to obtain a better solution with a different topology. We next perturb the current topology $\mathbf{Y}$ to get a new topology $\hat{\mathbf{Y}}$ to further improve the current solution $\{\mathbf{y}, \mathbf{f}\}$, as discussed in the next subsection.

4.4. Topology perturbation

We adopt a *branch exchange* algorithm similar to that in [17] for topology perturbation. This algorithm deletes a link from the current topology $\mathbf{Y}$. It then chooses a remaining link from the candidate set $\mathcal{E}_c$ and inserts it into the reduced topology. When deleting and inserting links, the degree constraints (11) and (13) should always be satisfied. The performance of this algorithm depends on the decision rules for link deletion and insertion. We adopt the following strategies in our algorithm.

- Link deletion: for objective function L, we delete the link with the minimum load λ_{ij}. For objective functions T_1 and T_2, we delete the link with the minimum delay. That is, delete the least used link from the current topology.
- Link insertion: for objective function L, let the sum load at node i be $\lambda_i = \sum_{j=1}^{n} y_{ij} \cdot \lambda_{ij}$. We insert the link (i, j) with

$$\text{argmax}\{x_{ij} \cdot (\lambda_i + \lambda_j) | i \neq j, i, j \in V, y_{ij} = 0\}. \quad (39)$$

For objective functions T_1 and T_2, let the sum delay at node i be $t_i = \sum_{j=1}^{n} y_{ij} \cdot t_{ij}$. We insert the link (i,j) with

$$\text{argmax}\{x_{ij} \cdot (t_i + t_j) | i \neq j, i, j \in V, y_{ij} = 0\}. \quad (40)$$

If no improvement is achieved by inserting such a link (i, j), the link that achieves the second largest value will be inserted in the next iteration, and so forth.

The algorithm iteratively perturbs the topology and computes the network flows with the resulting new topology. It terminates with solution $\{\mathbf{y}, \mathbf{f}\}$ when the maximum number of iterations is achieved or when no further perturbation can be made.

4.5. Remarks

The heuristic algorithm can be executed independently to solve Problem OPT-TDLB for an underlying FSO network, as discussed above. In addition, the heuristic is complementary to the RLT-based branch-and-bound algorithm developed in Section 3. In practice, the RLT-based branch-and-bound algorithm can be executed at relatively large time intervals (e.g., when significant changes occur), to provide new topology $\mathbf{y}$ and network flows $\mathbf{f}$ with guaranteed optimality. On the other hand, the heuristic algorithm can be kept on running or executed at more frequent intervals. It keeps on perturbing the topology generated by the branch-and-bound algorithm and computing new network flows, thus making the FSO network dynamically reconfigurable and adaptive to small timescale changes such as bad weather or fluctuation in the traffic matrix $\mathbf{F}$.

5. Simulation studies

5.1. Simulation setting

In the simulations, n BS's are randomly deployed in a rectangular region. We present simulation results for cases $n=5$, $n=7$, and $n=15$. Assume that the traffic matrix $\mathbf{F}$ is

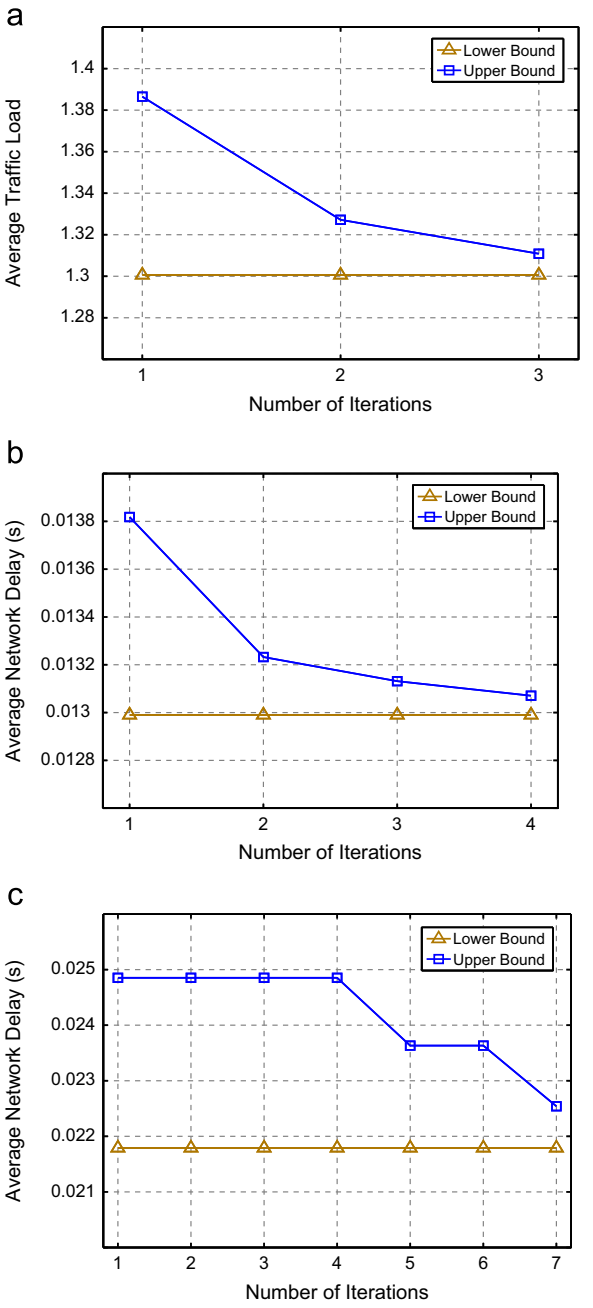


Fig. 4. Convergence of the RLT-based branch-and-bound algorithm for the 5-node network with optimality gap $\epsilon = 0.01, 0.01$, and 0.07, respectively. (a) OPT-TDLB(L) with SRD traffic, gap=0.78%; (b) OPT-TDLB(T_1), gap=0.62%; (c) OPT-TDLB(T_2), gap=3.33%.

given, which is randomly generated for the s–d pairs [8]. Each s–d flow f_{sd} ranges from 0 to 40% of the FSO link capacity. The constraints of minimum node degrees are determined from $\mathbf{F}$ as in (18).

Link connectivity is determined by link reliability, which is derived using the FSO channel model given in Section 2.2. Then x_{ij}'s are known and the candidate link set $\mathcal{E}_c$ is found. We use reliability threshold $\gamma_{th} = 99.99\%$ when determining the candidate set. For LRD traffic, the index of dispersion a is set to a half of the link capacity and Hurst parameter H is chosen to be 0.7.

The proposed algorithms are implemented in Matlab ver 7.4.0 for manipulating matrices and solving the LP relaxations. The codes are executed on a standard PC with a Core Duo 2.20 GHz processor and 2 GB memory.

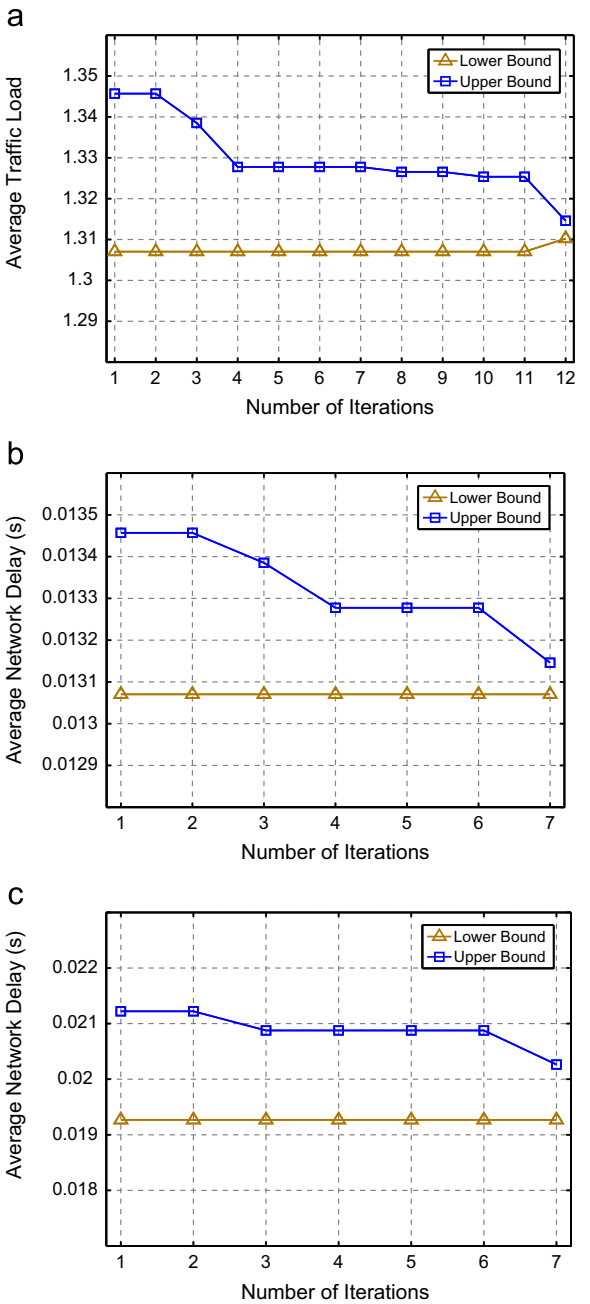


Fig. 5. Convergence of the RLT-based branch-and-bound algorithm for the 7-node network with optimality gap $\epsilon = 0.01, 0.01$, and 0.07, respectively. (a) OPT-TDLB(L) with SRD traffic, gap=0.33%; (b) OPT-TDLB(T_1), gap=0.58%; (c) OPT-TDLB(T_2), gap=4.91%.

5.2. RLT-based branch-and-bound algorithm

5.2.1. Convergence and optimality gap

The RLT-based branch-and-bound algorithm can provide $(1-\epsilon)$−optimal solutions. We first examine the convergence performance and the final optimality gap achieved by the proposed algorithm.

In Fig. 4, we plot the simulation results for the 5-node FSO network. We use the RLT-based branch-and-bound algorithm to solve the three variations of Problem OPT-TDLB, i.e., Problem OPT-TDLB(L) with SRD traffic, Problem OPT-TDLB(T_1), and Problem OPT-TDLB(T_2). The termination criteria are chosen to be $\epsilon = 0.01$, 0.01, 0.07 for the three problems, respectively. The simulations terminate when $\text{LB} \geq (1-\epsilon) \cdot \text{UB}$, and the upper bounding solution is a feasible one. The Problem OPT-TDLB(L) results are shown in Fig. 4(a), where the proposed algorithm terminates after three iterations and achieves an optimality gap of 0.78%. The Problem OPT-TDLB(T_1) results are shown in Fig. 4(b), where the algorithm terminates after four iterations and achieves a 0.62% optimal gap. In the case of Problem OPT-TDLB(T_2), the algorithm terminates after seven iterations (due to the high variation in LRD traffic and the need to handle logarithm functions) with a 3.33% optimality gap, as shown in Fig. 4(c).

In Fig. 5, we show convergence and optimality gap results for the 7-node FSO network. The proposed algorithm takes more iterations to finish, due to increased network size. For Problem OPT-TDLB(L), the algorithm takes 12 iterations to achieve an optimal gap of 0.33%, as shown in Fig. 5(a). For Problem OPT-TDLB(T_1), the algorithm takes seven iterations to achieve an optimality gap of 0.58%, as shown in Fig. 5(b). For Problem OPT-TDLB(T_2), the algorithm takes seven iterations to achieve an optimality gap of 4.91%, as shown in Fig. 5(c).

5.2.2. Computational cost

We next present the computation cost results in the form of execution time of the proposed algorithm, which largely depends on the efficiency of the underlying LP solver on handling large matrices. Since the algorithm is an iterative one, we focus on the execution time per iteration for clarity.

In Table 1, the average parameter building time (i.e., the time spent on obtaining the LP relaxations and assembling the input matrices to the LP solver), denoted by T_{build}, and the average execution time (i.e., the time it takes for the LP solver to solve the relaxed problem) for solving one subproblem in the Problem List $\mathcal{P}$ are listed, denoted by T_{exe}. In the table, $\mathbf{Q}$ and $\mathbf{Q}_{eq}$ are matrix inputs to the LP solver, whose sizes largely depend on the number of constraints of the LP relaxation l-OPT-TDLB.

We find that the building and execution times are proportional to the parameter matrix sizes. To compute

Table 1
Average building and execution time per subproblem for the branch-and-bound algorithm.

Problem type	Size of $\mathbf{Q}$	Size of $\mathbf{Q}_{eq}$	E[T_{build}] (s)	E[T_{exe}] (s)
OPT-TDLB(L), LB, 5 nodes	425×440	131×440	0.835	0.710
OPT-TDLB(L), UB, 5 nodes	N/a	112×252	0.018	0.036
l-OPT-TDLB(T_1), LB, 5 nodes	445×480	151×480	1.075	0.219
l-OPT-TDLB(T_1), UB, 5 nodes	12×276	124×276	0.024	0.042
l-OPT-TDLB(T_2), LB, 5 nodes	733×540	159×540	4.518	0.242
l-OPT-TDLB(T_2), UB, 5 nodes	216×312	136×312	0.140	0.091
l-OPT-TDLB(T_1), LB, 7 nodes	1813×1848	358×1848	85.067	7.995
l-OPT-TDLB(T_1), UB, 7 nodes	N/a	318×1032	1.296	0.081
l-OPT-TDLB(T_1), LB, 7 nodes	1855×1932	400×1932	93.516	8.963
l-OPT-TDLB(T_1), -UB, 7 nodes	24×1080	342×1080	1.613	0.089
l-OPT-TDLB(T_2), LB, 7 nodes	$2{,}543 \times 2058$	436×2058	184.305	14.110
l-OPT-TDLB(T_2), UB, 7 nodes	432×1152	366×1152	4.823	0.281

the LB for a subproblem, the corresponding LP relaxation, l-OPT-TDLB, should be solved to determine the topology design variables $\mathbf{y}$ and the multipath routing variables $\mathbf{f}$. The constraints for f_{ij}^{sd}'s have more impact on the matrix sizes, due to new linear constraints introduced during the reformulation and linearization procedure for the objective functions. Once the lower-bounding solution $\hat{\delta}$ is found, it takes negligible time for the local search algorithm to find an upper-bounding solution $\hat{\delta}$ in the neighborhood, since the topology is already given in $\hat{\delta}$. For example, the average execution time is approximately 14 s to get the LB for a subproblem of l-OPT-TDLB(T_2), while it only takes 0.3 s to obtain the UB for the subproblem.

From these simulation studies, we find that the branch-and-bound algorithm can produce feasible solutions that are highly competitive, as indicated by the very small optimality gaps when the algorithm terminates. Although the global optimum is unknown, the guarantee on the optimality gap ensures near-optimal solutions. The tolerance ϵ provides a convenient handle for the trade-off between computation time and optimality. These are useful features for the design and control of FSO networks.

5.3. Fast heuristic algorithm

In this section, we present simulation studies on the performance of the proposed heuristic algorithm. We first examine the optimality performance of the proposed heuristic algorithm. Since the parameters (such as traffic matrix and node location) are not provided in prior works [7,8], it is non-trivial to truthfully reproduce their results. Our strategy is to compare the heuristic algorithm with the RLT-based branch-and-bound algorithm developed in Section 3. Specifically, the lower and upper bounds provided by the branch-and-bound algorithm are good indicators of the global optimal for the joint topology design and load balancing problem. The gap between the heuristic solution and the branch-and-bound algorithm solution provides an upper bound on the optimality gap achieved by the heuristic algorithm.

In Fig. 6, we plot the network-wide average traffic load L and average delays T_1/T_2 for the 7-node network. In the figures, the markers correspond to the iterations for the algorithms. When the objective function is L, the RLT-based branch-and-bound algorithm terminates after 12 iterations, achieving an optimality gap of 0.33%. The heuristic algorithm also terminates after 12 iterations, achieving an optimality gap of 1.11%. When the objective function is T_1, the RLT-based branch-and-bound algorithm achieves an optimality gap 0.58% in seven iterations, while the heuristic achieves an optimality gap 1.12% in 12 iterations. When the objective function is T_2, the RLT-based branch-and-bound algorithm achieves an optimality gap 4.91% in seven iterations, while the heuristic achieves an optimality gap 4.33% in 16 iterations.

The convergence and optimality of the heuristic solutions are shown in Fig. 7 for the 15-node network. We also plot the lower bound for comparison purpose. It takes 76, 75, and 55 iterations, respectively, for the heuristic algorithm to achieve optimality gaps of 9.16%, 9.15%, and 8.16%, for the three problems, respectively.

We next compare the computation times of the proposed algorithms. In Fig. 6, the x-axis on the top marks the execution time of the heuristic algorithm, while the x-axis at the bottom is for the execution times of the RLT-based branch-and-bound algorithm. We find that the heuristic algorithm is considerably more computationally efficient than the RLT-based branch-and-bound algorithm. For example, in Fig. 6(a), the heuristic algorithm achieves an optimality gap of 1.11% in about 16 s, while the branch-and-bound algorithm achieves an optimal gap of 0.33% in about 2100 s. The same observation can be made in all the simulation studies.

Finally we present the execution time of each iteration for the heuristic algorithm in Table 2. It can be seen that the building time is in the range of a few seconds and the execution time is in the range from tens to a few hundreds of ms. The largest building time 27.538 s occurs for the 15-node network under LRD traffic, which is due to the extra constraints introduced by the polyhedral outer approximation given in (33).

6. Related work

This paper is closely related to the work on network planning [18–20]. Network planning problems usually belong to the class of combinatorial optimization problems. Since such problems are NP-hard, *metaheuristics*,

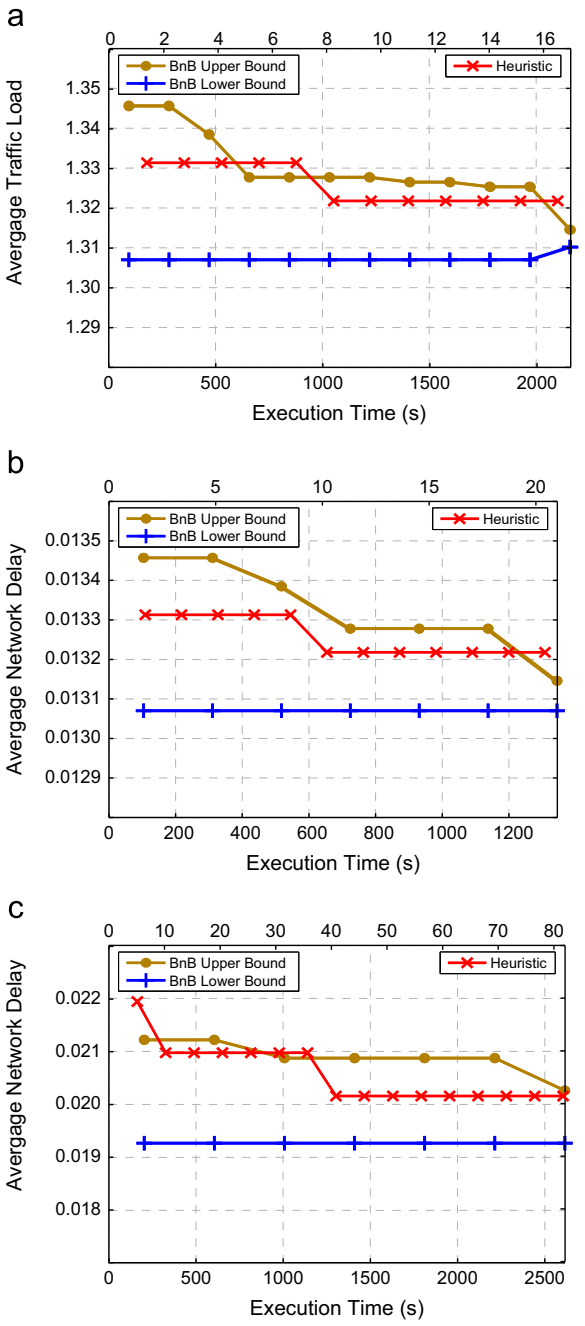


Fig. 6. Performance of the heuristic algorithm for the 7-node network, compared to the results of the RLT-based branch-and-bound algorithm. (a) OPT-TDLB(L) with SRD traffic, gap=1.11%; (b) OPT-TDLB(T_1), gap=1.12%; (c) OPT-TDLB(T_2), gap=4.33%.

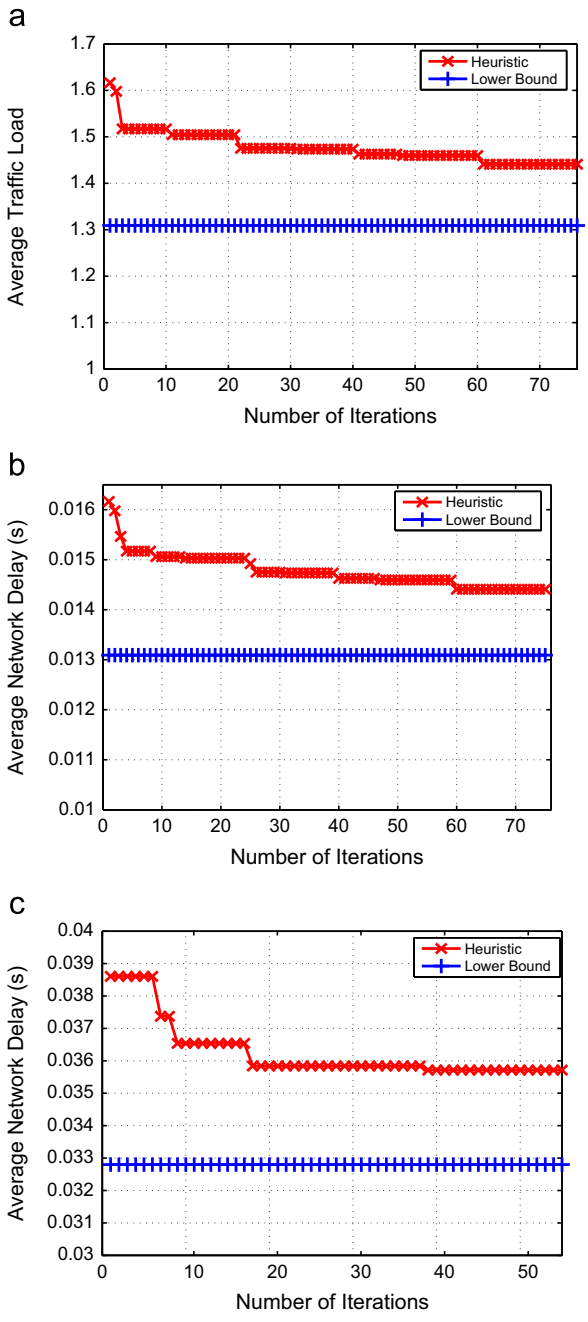


Fig. 7. Performance of the heuristic algorithm for the 15-node network. (a) OPT-TDLB(L) with SRD traffic, gap=9.16%; (b) OPT-TDLB(T_1), gap=9.15%; (c) OPT-TDLB(T_2), gap=8.16%.

e.g., simulated annealing [18] or genetic algorithms [20] are used to provide sub-optimal solutions. The main limitations of these approaches are the lack of performance guarantees and the relatively high complexity.

This work is also related to FSO research, which has attracted considered interest recently. See excellent surveys in [3,21] and references therein. Major FSO research has focused on the PHY, such as hardware architecture [3] and optical channel modeling [10,13]. There have been several work on the design [5,22] and configuration [4,7,21,23] of FSO networks. As in network planning, MILPs are usually formulated and various heuristic algorithms are proposed to provide sub-optimal solutions. Since such MILP-based solutions are usually centralized ones, a bootstrapping procedure is required to build a connected network from the set of initially isolated FSO nodes [4,24,25].

Table 2
Average building and execution times per iteration with the heuristic algorithm.

Problem type	Size of $\mathbf{Q}$	Size of $\mathbf{Q}_{eq}$	E[T_{build}] (s)	E[T_{exe}] (s)
l-OPT-TDLB(L), 7 nodes	N/a	318×1032	1.296	0.081
l-OPT-TDLB(T_1), 7 nodes	24×1080	342×1080	1.613	0.089
l-OPT-TDLB(T_2), 7 nodes	432×1152	366×1152	4.823	0.281
l-OPT-TDLB(L), 15 nodes	N/a	500×1550	5.350	0.131
l-OPT-TDLB(T_1), 15 nodes	50×1650	550×1650	6.452	0.175
l-OPT-TDLB(T_2), 15 nodes	900×1800	600×1800	27.538	0.476

FSO links and traditional RF links are complementary to each with respect to data rate, interference, robustness, and range. Several papers have investigated the hybrid RF/FSO networks for enhanced performance [6,20,23,26]. In [6], we investigate the design and optimization of a two-tier wireless network: the lower tier is a wireless mesh network that is partitioned into clusters based on traffic demand, while the upper tier is an FSO network with a mesh topology for which the algebraic connectivity is maximized. In [20], genetic algorithms are used to improve the capacity performance with a minimum number of hybrid FSO/RF gateways. In [23], the authors propose to adaptively adjust both transmission power (of RF and FSO transmitters) and the optical beamwidth, to meet prescribed QoS requirements. In [26], the authors investigate radio signal transmission over terrestrial optical wireless channels under a WiMAX network setting, and provide an outage probability analysis.

The "pseudo-wired" FSO links are highly desirable for interference management and security. As contrast to wired links, the FSO links also allows great flexibility for adaptive to network dynamics since they are steerable. In [27], the authors exploit slow-fading FSO channels and propose an adaptive transmission algorithm that can adjust transmit power and modulation according to channel status information feedback. In [28], the authors propose a fiber-bundle approach for beam steering to enhance the tolerance of optical link misalignment. In [29], we investigated the problem of maximizing the FSO network-wide throughput under the constraints of power budget and number of FSO transceivers, where cooperative relays are incorporated.

7. Conclusion

In this paper, we studied problem of joint topology design and load balancing in FSO networks. For given traffic demand, the objective is to design the topology and multi-path routing policies for end-to-end flows, such that network-wide average load or network-wide average delay can be minimized. We developed an effective branch-and-bound algorithm based on RLT for the formulated MILP and MINLP problems, which can provide highly competitive solutions with guaranteed performance. To reduce computation overhead, we also developed a fast heuristic algorithm with suboptimal solutions. The proposed algorithms are complementary to each other, for adaptation to network dynamics at both large and small timescales.

Acknowledgement

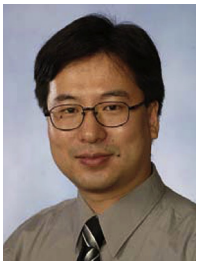
This work is supported in part by the National Science Foundation under Grant CNS-1145446, and through the Broadband Wireless Access & Application Center (BWAC) at Auburn University.

References

[1] I.K. Son, S. Mao, Fast heuristic algorithm for joint topology design and load balancing in fso networks, in: Proceedings of the IEEE GLOBECOM 2011, Miami, FL, 2011, pp. 1–5.
[2] I.K. Son, S. Mao, A reformulation–linearization technique-based approach to joint topology design and load balancing in FSO networks, in: Proceedings of the IEEE GLOBECOM 2011, Miami, FL, 2011, pp. 1–6.
[3] V.W. Chan, Free-space optical communications, IEEE/OSA Journal of Lightwave Technology 24 (12) (2006) 4750–4762.
[4] F. Liu, U. Vishkin, S. Milner, Bootstrapping free-space optical networks, IEEE Journal on Selected Areas in Communications 24 (12) (2006) 13–22.
[5] P.C. Gurumohan, J. Hui, Topology design for free space optical networks, in: Proceedings of the IEEE ICCCN'03, 2003, pp. 576–579.
[6] I.K. Son, S. Mao, Design and optimization of a tiered wireless access network, in: Proceedings of the IEEE INFOCOM'10, San Diego, CA, 2010, pp. 1–9.
[7] A. Desai, S. Milner, Autonomous reconfiguration in free-space optical sensor networks, IEEE Journal on Selected Areas in Communications 23 (8) (2005) 1556–1563.
[8] A. Kashyap, K. Lee, M. Kalantari, S. Khuller, M. Shayman, Integrated topology control and routing in wireless optical mesh networks, Computer Networks 51 (15) (2007) 4237–4251.
[9] S. Kompella, S. Mao, Y. Hou, H. Sherali, On path selection and rate allocation for video in wireless mesh networks, IEEE/ACM Transactions on Networking 17 (1) (2009) 212–224.
[10] X. Zhu, J.M. Kahn, Free-space optical communication through atmospheric turbulence channels, IEEE Transactions on Communications 50 (8) (2003) 1293–1300.
[11] M. Al-Habash, L. Andrews, R. Philips, Mathematical model for the irradiance probability density function of a laser beam propagating through turbulent media, Society of Photo-Optical Instrumentation Engineers 40 (8) (2001) 1554–1562.
[12] A. Prokes, Atmospheric effects on availability of free space optics systems, Optical Engineering 48 (6) (2009) 066001.
[13] A. Vavoulas, H. Sandalidis, D. Varoutas, Weather effects on fso network connectivity, IEEE/OSA Journal of Optical Communications and Networking 4 (10) (2012) 734–740.
[14] D. Bertsekas, R. Gallager, Data Networks, Prentice-Hall, 1992.
[15] I. Norros, On the use of fractional Brownian motion in the theory of connectionless networks, IEEE Journal on Selected Areas in Communications 13 (6) (1995) 953–962.
[16] F.R.K. Chung, Spectral Graph Theory, American Mathematical Society, 1997.
[17] E. Miguez, J. Cidras, E. Diaz-Dorado, J.L. Garcia-Dornelas, An improved branch-exchange algorithm for large-scale distribution network planning, IEEE Transactions on Power Systems 17 (4) (2002) 931–936.
[18] C. Ersoy, S. Panwar, Topological design of interconnected LAN-MAN networks, IEEE Journal on Selected Areas in Communications 11 (8) (1993) 1172–1182.

[19] R.M. Krishnaswamy, K. Sivarajan, Design of logical topologies: A linear formulation for wavelength-routed optical networks with no wavelength changers, IEEE/ACM Transactions on Networking 9 (2) (2001) 186–198.
[20] M.N. Smadi, S.C. Ghosh, A.A. Farid, T.D. Todd, S. Hranilovic, Free-space optical gateway placement in hybrid wireless mesh networks, IEEE/OSA Journal of Lightwave Technology 27 (14) (2009) 2688–2697.
[21] C.C. Davis, I.I. Smolyaninov, S.D. Milner, Flexible optical wireless links and networks, IEEE Communications Magazine 41 (3) (2003) 51–57.
[22] S. Ghosh, K. Basu, S. Das, Meshup: self-organizing mesh-based topologies for next generation radio access networks, Ad Hoc Networks Journal 5 (6) (2007) 652–679.
[23] O. Awwad, A. Al-Fuqaha, B. Khan, G. Brahim, Topology control schema for better QoS in hybrid rf/fso mesh networks, IEEE Transactions on Communications 60 (5) (2012) 1398–1406.
[24] I.K. Son, S. Kim, S. Mao, Building robust spanning trees for free space optical networks, in: Proceedings of the IEEE MILCOM 2010, San Jose, CA, 2010, pp. 1397–1402.
[25] H. Zhou, A. Babaei, S. Mao, P. Agrawal, Algebraic connectivity of degree constrained spanning trees for FSO networks, in: Proceedings of the IEEE ICC 2013, Budapest, Hungary, 2013, pp. 1–6.
[26] N. Vaiopoulos, H. Sandalidis, D. Varoutas, WiMAX on FSO: outage probability analysis, IEEE Transactions on Communications 60 (10) (2012) 2789–2795.
[27] M. Karimi, M. Uysal, Novel adaptive transmission algorithms for free-space optical links, IEEE Transactions on Communications 60 (12) (2012) 3808–3815.
[28] D. Zhou, P. LoPresti, H. Refai, Enlargement of beam coverage in FSO mobile network, IEEE/OSA Journal of Lightwave Technology 29 (10) (2011) 1583–1589.
[29] H. Zhou, D. Hu, S. Mao, P. Agrawal, Joint relay selection and power allocation in cooperative FSO networks, in: Proceedings of the IEEE GLOBECOM 2013, Atlanta, GA, 2013, pp. 1–6.

In Keun Son received his B.E. degree in Information Engineering from Korea Military Academy, Seoul, South Korea in 1997, M.S. degree in Computer Science from KAIST (Korea Advanced Institute of Science and Technology), Daejun in 2005, and Ph.D. degree in Electrical and Computer Engineering from Auburn University, Alabama, USA in 2010. His research interests include network topology design and performance optimization in Wireless Radio Networks and Wireless Optical Networks. He is currently serving as an information technology specialist in the military of South Korea. Since 2001, he has been working on developing military C2 (Command and Control) systems as an intelligence system developer, an interoperation coordinator, and a system integrator. He became a manager of Korea Army C2 System Development Program in DAPA (Defense Acquisition Program Administration) in 2012.



Shiwen Mao received Ph.D. in electrical and computer engineering from Polytechnic University, Brooklyn, NY, USA (now Polytechnic Institute of New York University) in 2004. He was a research staff member with IBM China Research Lab from 1997 to 1998. He was a Postdoctoral Research Fellow/Research Scientist in the Bradley Department of Electrical and Computer Engineering at Virginia Polytechnic Institute and State University (Virginia Tech), Blacksburg, VA, USA from 2003 to 2006. Currently, he is the McWane Associate Professor in the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL, USA.

His research interests include cross-layer optimization of wireless networks and multimedia communications, with current focus on cognitive radios, femtocell networks, 60 GHz mmWave networks, and smart grid. He is on the Editorial Board of IEEE Transactions on Wireless Communications, IEEE Internet of Things Journal, IEEE Communications Surveys and Tutorials, Elsevier Ad Hoc Networks Journal, Wiley International Journal of Communication Systems, and ICST Transactions on Mobile Communications and Applications. He serves as the Director of E-Letter of the Multimedia Communications Technical Committee (MMTC), IEEE Communications Society for 2012–2014.

Dr. Mao is a coauthor of TCP/IP Essentials: A Lab-Based Approach (Cambridge University Press, 2004). He is a co-recipient of The IEEE ICC 2013 Best Paper Award. He received the 2013 IEEE ComSoc MMTC Outstanding Leadership Award and was named the 2012 Exemplary Editor of IEEE Communications Surveys and Tutorials. He was awarded the McWane Endowed Professorship in the Samuel Ginn College of Engineering for the Department of Electrical and Computer Engineering, Auburn University in August 2012. He received the US National Science Foundation Faculty Early Career Development Award (CAREER) in 2010. He is a co-recipient of The 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems and The Best Paper Runner-up Award at QShine 2008. He also received Auburn Alumni Council Research Awards for Excellence-Junior Award and two Auburn Author Awards in 2011. Dr. Mao holds one US patent.

Sajal K. Das is a University Distinguished Scholar Professor of Computer Science and Engineering and the Founding Director of the Center for Research in Wireless Mobility and Networking (CReWMaN) at the University of Texas at Arlington (UTA). During 2008-2011, he served the US National Science Foundation (NSF) as a Program Director in the Division of Computer Networks and Systems. His research interests include wireless and sensor networks, mobile and pervasive computing, cyber-physical systems and smart environments, smart grids and sustainability, security and privacy, biological and social networks, applied graph theory and game theory. He has published extensively in these areas, including 3 books, and holds five US patents. A recipient of the IEEE Computer Society Technical Achievement Award in 2009 for pioneering contributions to sensor networks, Dr. Das is the founding editor-in-chief of Elsevier's Pervasive and Mobile Computing journal.