24th Mediterranean Conference on Control and Automation (MED)
June 21-24, 2016, Athens, Greece

Trust-Aware Network Utility Optimization in Multihop Wireless Networks with Delay Constraints

Evripidis Paraskevas[1], Tao Jiang[2] and John S. Baras[1]

***Abstract*— Many resource allocation problems can be formulated as a constrained maximization of a utility function. Network Utility Maximization (NUM) applies optimization techniques to achieve decomposition by duality or the primal-dual method. Several important problems, for example joint source rate control, routing, and scheduling design, can be optimized by using this framework. In this work, we introduce an important network security concept, "trust", into the NUM formulation and we integrate nodes' trust values in the optimization framework. These trust values are based on the interaction history between network entities and community based monitoring. Our objective is to avoid routing packets though paths with large percentage of malicious nodes. We also add end-to-end delay constraints for each of the traffic flows. The delay constraints are introduced to capture the imposed quality of service (QoS) requirements for each traffic flow.**

***Index Terms*— cross-layer optimization, trust, source rate control, multipath routing, scheduling**

I. Introduction

The problem of resource allocation in wireless networks has been a growing area of research. Recent advances in the area of network utility maximization (NUM) driven cross-layer design [1], [2], [3] have led to efforts on top-down development of next generation wireless network architectures. By linking decomposition of the NUM problem to different layers of the network stack, we are able to design protocols, based on the optimal NUM derived algorithms [4], which provide much better performance gain over the current network protocols.

In recent years, network security has become increasingly important in the context of wireless multihop networks. Different types of network attacks can be released and affect significantly their performance. In our work, we consider that the adversary is capable of releasing some form of *denial of service* (DoS) attack. To capture the notion of security, we use "trust weights" [5] in the network utility optimization process. These weights indicate whether a network entity is malicious or not, based on its interactions with the other network entities. Trust weights are developed by our network community based on monitoring and are disseminated via efficient methods so that they are timely available to all nodes [6].

Work partially supported by DARPA under the Wireless Network Defense (WND) program, Contract No. FA8750-15-C-0038 and by AFOSR MURI grant FA9550-10-1-0573.

[1]Evripidis Paraskevas and John S. Baras are with the Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, USA. (email: {evripar, baras}@umd.edu)

[2]Tao Jiang is with Samsung Research America-Dallas, Richardson, TX, USA. (email: t.jiang@sta.samsung.com)

End-to-end delay is a critical *quality of service* (QoS) requirement for resource-constrained wireless networks. Network applications have different delay requirements. Hence, it is crucial to take into account these delay constraints, corresponding to different classes of traffic flows, to our trust-aware NUM problem.

In this paper, we incorporate the notion of security into the NUM problem, by using the trust values of the network nodes. Users get higher utility, when they relay packets though trusted paths. Hence, our proposed *trust-aware NUM* process ensures that untrusted paths (with malicious entities) will not receive high traffic rate. We also add end-to-end delay constraints in the NUM problem based on [7]. These delay constraints indicate the QoS requirements of the different traffic flows. The notion of *link capacity margin* [7] is used to control the end-to-end delay. Finally, we propose a distributed cross-layer optimization algorithm for the trust-aware NUM problem with delay constraints. The distributed algorithm is based on the dual decomposition into source rate control, average end-to-end delay control and scheduling subproblems.

The rest of this paper is organized as follows. Section II reviews the related work in the literature on network utility maximization (NUM) problem formulation and its security considerations. Section III introduces the system model that we consider in this paper, including the network model, the adversary model and the trust values estimation. Section IV outlines the optimization constraints, which include link capacity, average end-to-end delay, and scheduling constraints, as well as the primal optimization problem. The dual function and its decomposition into different subproblems is studied in Section V. Section V-D discusses the distributed algorithm for solving the network utility maximization (NUM) problem. The simulation results for our proposed trust-aware NUM problem with delay constraints are shown in Section VI. Section VII concludes this paper and discusses future work.

II. Related Work

Network utility maximization (NUM) problems have been investigated widely during recent years. Most of works [1], [2], [3], [4] focus on using NUM for cross-layer optimization. Chiang et. al [1] introduced a methodology for optimizing functional modules of the network, such as congestion control, routing and scheduling, through optimization decomposition. Chen et. al [2] proposed a subgradient algorithm for cross-layer optimization and its extension to time-varying channels and adaptive multi-rate devices. Decomposition

978-1-4673-8345-5/16/$31.00 ©2016 IEEE

methods for the solution of the NUM problem are proposed in [8].

Several works have introduced delay considerations for the traffic flows into the NUM problem formulation. Trichakis et. al [9] proposed a dynamic NUM formulation with delivery contracts for the different traffic flows. One other concept for delay, used for the NUM problem, is the *link capacity margins*. These margins were introduced in [7] and [10] to control the average end-end delay. Link margins represent the estimated delay of the link, because higher link margin indicates lower link congestion and thus less delay.

As far as we are concerned, there are not a lot of works that relate security with the NUM problem [11], [12]. Tague et. al [12] proposed a jamming-aware throughput maximization approach. The authors use the jamming estimates in the NUM problem to allocate data traffic appropriately in order to achieve throughput maximization. They adopt an objective function, based on portfolio selection theory to maximize throughput for the different source nodes. Our work is one of the first to study trust-aware network utility maximization problems. Trust values affect the outcome of the NUM process and make it resilient to malicious nodes' behavior.

III. System Model

A. Network Model

We consider a multihop wireless network that can be defined by a graph $G(\mathcal{N}, \mathcal{L})$. The vertex set $\mathcal{N}$ represents the wireless network nodes. The edge set $\mathcal{L}$ represents the wireless links. An ordered pair of nodes (i, j) belongs to the edge set $\mathcal{L}$ if and only if node j can receive data packets directly from node i. For simplicity, we also use the symbol ℓ to denote a wireless link. We assume that all node-to-node communication is unicast, i.e. each packet transmitted by a node $i \in \mathcal{N}$ is intended for a unique $j \in \mathcal{N}$ with $(i, j) \in \mathcal{L}$. Each of the wireless links has a maximum capacity $c_{i,j}$.

There is a set $\mathcal{F}$ of network traffic flows that share the wireless network resources and each flow $f \in \mathcal{F}$ is associated with a source node s. Each source node s in a subset $\mathcal{S} \subseteq \mathcal{N}$ generates data packets for a single destination node $d_s \in \mathcal{N}$. We assume that each source node s constructs multiple routing paths with multiple hops to d_s in order to distribute the traffic demand and satisfy the flow related QoS requirements. We denote as $\mathcal{P}_s = \{p_{s1}, \ldots, p_{sP_s}\}$ the collection of the alternative paths P_s that can be used to route packets from s to d_s. Each path $p_{sk} \in \mathcal{P}_s$ is specified by a subset of wireless links and is assumed to be loop-free.

Let $\mathbf{x_s}$ denote the $P_s \times 1$ traffic rate vector with which data packets are sent from s to d_s over multiple paths $p_{sk} \in \mathcal{P}_s$, and multiple hops. Each component of the vector x_{sk} denotes the proportion of traffic rate allocated to the corresponding path p_{sk}, which routes data packets from source node s to destination d_s. The total data rate of the source s is given by the summation of x_{sk} over $k = 1, \ldots, P_s$.

We assume that the traffic rate vector $\mathbf{x_s}$ of each flow is constrained to a non-negative orthant. The traffic rate allocated to each traffic flow should also not exceed a maximum data rate $\mathcal{R}_s$. Therefore, each of the traffic rate vectors $\mathbf{x_s}$ should satisfy the following constraints

$$\mathbf{x_s} \geq \mathbf{0} \quad , \forall s \in \mathcal{S} \tag{1}$$

$$\mathbf{1}^T \mathbf{x_s} \leq \mathcal{R}_s \quad , \forall s \in \mathcal{S} \tag{2}$$

In Eq. (1) each component of the vector is nonnegative.

We denote by $\mathbf{R_s} = [(R_s)_{k(i,j)}]_{(P_s \times |\mathcal{L}|)}$ the routing matrix that indicates the different paths from source node s to destination d_s. Element $(R_s)_{k(i,j)}$ of the routing matrix is defined as follows

$$(R_s)_{k(i,j)} = \begin{cases} 1, & \text{if } p_{sk} \text{ passes through link } (i, j). \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

B. Security Considerations: Adversary Model and Trust

In this paper, we study the network utility optimization problem with considerations of network security. All previous works on the network utility maximization (NUM) problems assume that nodes operate correctly. However, nodes may be compromised by attackers, their communication may be blocked or interfered by attackers, or they may just be misconfigured. Hence, we believe it is crucial to take the security aspect into consideration in the NUM problems.

Adversary Model: We assume that the adversarial node is not following the network protocol and attempts to disrupt communication by dropping or modifying data packets. In this work, we mainly consider that the adversary is capable of dropping data packets in a deterministic or probabilistic way. This type of attack leads to lack of **availability** of the network and constitutes a *denial of service* (DoS) attack. The DoS attack affects significantly some QoS requirements, such as end-to-end delay and packet delivery ratio. Thus, in order to support *time-critical* applications, the traffic allocation mechanisms should be resilient to these types of attacks. In general, the notion of trust can also address different types of attacks. In this case, trust evaluation should incorporate authentication or inspection mechanisms at the receiver and intermediate nodes to define the trustworthiness of a node.

Trust Estimates: The concept of security, which we adopt to distinguish misbehaving nodes in this work is **trust**. Trust is a very critical concept not only in computer networks, but also in various other networks that involve intelligent decisions, such as social networks. All the connections and communications in these networks imply the existence of trust. We assume that there are mechanisms to efficiently distribute trust evidence [13], such that duplicates of evidence documents are stored in places where they are most needed. Once the trust evidence is in hand, nodes could evaluate the trustworthiness of other nodes. We define the trust estimated value of node i as ν_i. In this work, we follow the definition of trust in [14] and the trust values take continuous numerical values in $[0, 1]$.

We define an *update period* of the trust estimates denoted by T_{update}. During the update period, represented by the time interval $[t - T_{update}, t]$, the trust evaluation mechanism provides fresh estimates of the trust values for nodes $i \in \mathcal{S}$, based on the interaction between network entities.

In order to prevent significant variation in the trust estimate ν_i of node i and to include memory of the trust evaluation, we suggest using an exponential weighted moving average (EWMA) [15] to update the trust estimate as a function of the previous estimate, as indicated in [12]. Hence, the trust value of node i at time t is given by

$$\nu_i(t) = (1-\alpha)\nu_i(t - T_{update}) + \alpha\nu_i^{new}, \tag{4}$$

where $\alpha \in [0,1]$ is a constant weight indicating the relative preference between updated and historic samples of trust values and ν_i^{new} is the fresh estimate of the trust value for node i, given from the trust evaluation mechanism.

Given the trust values for the intermediate nodes across a path p_{sk}, the source node s evaluates the updated *aggregate trust value* for the path $p_{sk} \in \mathcal{P}_s$. The *aggregate trust value* of the path p_{sk} is denoted by t_{sk} and can be expressed as the product of the corresponding trust values along the path as

$$t_{sk} = \prod_{j:(i,j)\in p_{sk}} \nu_j \tag{5}$$

One additional parameter that we should consider in the data traffic allocation process is *path reliability* [16]. In our work, path reliability is indicated by the corresponding aggregate trust value t_{sk} over the path p_{sk}, which denotes the proportion of the allocated traffic flow that is actually received at destination node d_s. Hence, in order to maintain the reliability of the network the received traffic rate for each traffic flow should exceed a certain threshold. We denote this threshold for each source node s as $\mathcal{R}_s^{thres}$, which is proportional to the maximum allowable rate $\mathcal{R}_s$. Thus, our allocated traffic rate for each source node should satisfy the reliability constraint

$$\sum_{p_{sk}\in\mathcal{P}_s} t_{sk}x_{sk} \geq \mathcal{R}_s^{thres}, \quad \forall s \in \mathcal{S} \tag{6}$$

IV. Network Utility Maximization Formulation

In this section, we present the optimization framework for trust-aware network utility maximization (NUM). We first develop a set of optimization constraints and then we formulate the trust-aware utility optimization problem.

A. Optimization Constraints

Link Capacity constraint: To define capacity constraints we first introduce the *link capacity margin* optimization variables, which were initially introduced in [7] and [10], in order to capture the imposed delay constraints. We denote by $\sigma_{i,j}$ the link capacity margin of link $(i,j) \in \mathcal{L}$. Link capacity margin is defined as the difference between scheduled capacity of a wireless link and the maximum allowable traffic flow passing though it and it is used to control link delay and therefore the average end-to-end delay.

We also need to take into account our trust estimates for the capacity constraints of each link $(i,j) \in \mathcal{L}$. Based on the capabilities of the malicious nodes, described in Section III-B, the initially allocated traffic rate x_{sk} can be significantly reduced at malicious intermediate nodes because of dropping attacks. The decrease of the traffic rate is proportional to the *aggregate trust value* of the selected path. To be more specific, the decrease of the rate observed at an intermediate node is proportional to the aggregate trust value up to this intermediate node. Let $p_{sk}^{(i,j)}$ denote the sub-path of p_{sk} from source node s to the intermediate node j through link $(i,j) \in p_{sk}$. Then the traffic rate forwarded by intermediate node $j \in \mathcal{N}$ is computed by $t_{sk}^{(i,j)}x_{sk}$, where $t_{sk}^{(i,j)}$ is evaluated as the product of trust estimates over the sub-path $p_{sk}^{(i,j)}$, given by Eq. (5).

Hence, the capacity constraint associated with each wireless link $(i,j) \in \mathcal{L}$ is formulated as follows

$$\sum_{s\in\mathcal{S}} \sum_{k:(i,j)\in p_{sk}} t_{sk}^{(i,j)} x_{sk} \leq \hat{c}_{i,j} - \sigma_{i,j}, \quad \forall (i,j) \in \mathcal{L}, \tag{7}$$

where $\hat{c}_{i,j}$ is the capacity allocated to the wireless link $(i,j) \in \mathcal{L}$.

To define the different sub-paths' aggregate trust values, we denote by $\mathcal{T}_s$ the $P_s \times |\mathcal{L}|$ *aggregate trust incidence matrix* for source s, with rows indexed by the alternative paths p_{sk} and columns indexed by links (i,j). If a link (i,j) does not belong to any of the possible paths p_{sk} for source s, then the corresponding entry of the incidence matrix is equal to 0. The element $t(p_{sk},(i,j))$ or otherwise $t_{sk}^{(i,j)}$ for row p_{sk} and column (i,j) of $\mathcal{T}_s$ denotes the aggregate trust value of a possible sub-path $p_{sk}^{(i,j)}$ of path p_{sk} and is given by

$$t(p_{sk},(i,j)) = \begin{cases} \prod_{j':(i',j')\in p_{sk}^{(i,j)}} \nu_{j'} & \text{, if } (i,j) \in p_{sk} \\ 0 & \text{, otherwise} \end{cases} \tag{8}$$

Average end-to-end delay constraint: By using the link margin variables $\sigma_{i,j}$, we define as $\phi(\sigma_{i,j})$ the delay of link $(i,j) \in \mathcal{L}$. The function $\phi(\cdot)$ is typically a strictly convex, nonnegative valued, function of σ. The packet arrival process model determines the way that $\phi(\cdot)$ depends on $\sigma_{i,j}$. As described in [7] and [10], for Poisson process arrival, we have $\phi(\sigma_{i,j}) = \phi_{i,j} = 1/\sigma_{i,j}$. We define by $\phi(\sigma)$ the vector that has components the delay of all links of the network.

Delay constraints indicate the QoS requirements imposed to a specific traffic flow. The end-to-end delay is expressed by adding the link delays for each of the links over path p_{sk} of source node s. We denote the upper bound average delay constraint for each of the multiple paths of the source node s as $\mathcal{D}_s > 0$. Hence, the average end-to-end delay constraint for every source node s is given by

$$\mathbf{R_s}\phi(\sigma) \leq \mathbf{1}\mathcal{D}_s, \quad \forall s \in \mathcal{S} \tag{9}$$

Scheduling constraint: The capacity $\hat{c}_{i,j}$ allocated to the wireless link (i,j) should lie on the capacity region defined by Λ, described in [2] and [17]. Hence, our scheduling constraint is expressed as $\hat{\mathbf{c}} \in \Lambda$.

B. Utility Optimization

Each source s chooses a utility function $\mathcal{U}_s(\cdot)$ that evaluates the total data rate delivered to the destination d_s. Utility functions $\mathcal{U}_s(\cdot)$ are chosen to be strictly concave, continuous, monotonically increasing and twice differentiable.

Trust estimates for the different paths p_{sk} of a source node (defined in Sec. III-B) should be incorporated to the selected utility function $\mathcal{U}_s(\cdot)$. Source nodes should obtain greater utility when they decide to allocate higher traffic rate through routing paths with higher aggregate trust value t_{sk}. Hence, the utility function for each source node $s \in \mathcal{S}$ is defined as

$$\mathcal{U}_s(\mathbf{x}_s) = \sum_{p_{sk} \in \mathcal{P}_s} \Big(t_{sk} \log (x_{sk}) \Big) \tag{10}$$

The *primal utility optimization problem* formulation, based on the capacity, average end-to-end delay and scheduling constraints described in Sec. IV-A, is given by

$$\begin{aligned}
\max_{\mathbf{x},\sigma,\hat{\mathbf{c}}} \quad & \sum_{s\in\mathcal{S}} \mathcal{U}_s(\mathbf{x}_s) && \text{(11a)}\\
\text{s. t.} \quad & \sum_{s\in\mathcal{S}} \sum_{k:(i,j)\in p_{sk}} t_{sk}^{(i,j)} x_{sk} \le \hat{c}_{i,j} - \sigma_{i,j}, \forall (i,j) && \text{(11b)}\\
& \mathbf{R_s}\phi(\sigma) \le \mathbf{1}\mathcal{D}_s, \qquad \forall s \in \mathcal{S} && \text{(11c)}\\
& 0 \le \mathbf{1}^T\mathbf{x_s} \le \mathcal{R}_s, \qquad \forall s \in \mathcal{S} && \text{(11d)}\\
& \sum_{p_{sk}\in\mathcal{P}_s} t_{sk} x_{sk} \ge \mathcal{R}_s^{thres}, \quad \forall s \in \mathcal{S} && \text{(11e)}\\
& \hat{\mathbf{c}} \in \Lambda && \text{(11f)}
\end{aligned}$$

The trust-aware utility optimization problem is a strongly convex optimization problem, due to the strict concavity assumption of $U_s(\cdot)$ and the convexity of the capacity region. Therefore, there exists a unique optimal solution for the above primal problem, which we refer to as $(\mathbf{x}^*, \sigma^*, \hat{\mathbf{c}}^*)$.

V. Dual Decomposition Algorithm

In this section, we solve the utility optimization problem described in Eq. (11a) by applying dual decomposition [8], [18]. The decomposition of the optimization problem provides distributed algorithms, which solve the underlying optimization problem.

We define the Lagrange multipliers (dual variables) associated with the capacity and average end-to-end delay constraints. Let λ denote the $|\mathcal{L}| \times 1$ vector of *link prices* (dual variables) $\lambda_{i,j}$ associated with the capacity constraints for each wireless link. Also, let μ_s denote the $P_s \times 1$ vector of dual variables μ_{sk} associated with the average end-to-end delay constraints imposed to every traffic flow $s \in \mathcal{S}$.

To introduce the dual problem, we define the partial Lagrangian $L(\mathbf{x}, \sigma, \hat{\mathbf{c}}, \lambda, \mu)$ of the optimization problem by using the inequality constraints given from Eq. (11b) and (11c)

$$\begin{aligned}
&L(\mathbf{x}, \sigma, \hat{\mathbf{c}}, \lambda, \mu) = \\
&\sum_{s\in\mathcal{S}} \Big(\mathcal{U}_s(\mathbf{x}_s) - (\lambda^s)^T \mathcal{T}_s^T \mathbf{x}_s \Big) - \sum_{(i,j)\in\mathcal{L}} \Big(\phi_{i,j}\mu^{(i,j)} + \lambda_{i,j}\sigma_{i,j} \Big) \\
&\qquad + \lambda^T \hat{\mathbf{c}} + \sum_{s\in\mathcal{S}} \mu_s^T \mathbf{1}\mathcal{D}_s, \quad (12)
\end{aligned}$$

where λ^s is a sub-vector of the λ dual variable and is associated with the constraint in Eq. (11b). It defines the $|\mathcal{L}| \times 1$ column link price vector related to the links that belong to any of the paths $p_{sk} \in \mathcal{P}_s$ of a particular source node s and is given by

$$\lambda_{i,j}^s = \begin{cases} \lambda_{i,j} & \text{, if } (i,j) \in \bigcup_{p_{sk}\in\mathcal{P}_s} p_{sk} \\ 0 & \text{, otherwise} \end{cases} \tag{13}$$

and $\mu^{(i,j)} = \sum_{s\in\mathcal{S}} \sum_{k:(i,j)\in p_{sk}} \mu_{sk}[(R_s)_{k(i,j)}]$ denotes the combination of dual variables μ, which are related to a specific link (i,j) and is associated with the constraint (11c).

The dual objective function $h(\cdot)$ is then expressed as

$$\begin{aligned}
h(\lambda, \mu) \quad &= \sup_{\mathbf{x}\in\mathcal{X}} \Big\{ \sum_{s\in\mathcal{S}} \Big(\mathcal{U}_s(\mathbf{x}_s) - (\lambda^s)^T \mathcal{T}_s^T \mathbf{x}_s \Big) \Big\} && \text{(14a)}\\
&+ \sup_{\sigma\ge\mathbf{0}} \Big\{ - \sum_{(i,j)\in\mathcal{L}} \Big(\phi_{i,j}\mu^{(i,j)} + \lambda_{i,j}\sigma_{i,j} \Big) \Big\} && \text{(14b)}\\
&+ \sup_{\hat{c}\in\Lambda} \Big\{ \lambda^T \hat{\mathbf{c}} \Big\} && \text{(14c)}\\
&+ \sum_{s\in\mathcal{S}} \mu_s^T \mathbf{1}\mathcal{D}_s && \text{(14d)}
\end{aligned}$$

The dual optimization problem is defined by minimizing the dual objective function [19] over the dual vector variables λ and μ.

$$\min_{\lambda\ge\mathbf{0},\mu\ge\mathbf{0}} \quad h(\lambda, \mu) \tag{15}$$

For given dual variables λ and μ, we can identify in the above equation of $h(\lambda, \mu)$ three decoupled maximization problems which we can solve separately. These three problems correspond to source rate control in Eq. (14a), average end-to-end delay control in Eq. (14b), and scheduling in Eq. (14c) respectively.

By solving these three independent optimization problems we can derive the optimal values for the primal optimization problem $\mathbf{x}^*(\lambda, \nu)$, $\sigma^*(\lambda, \mu)$ and $\hat{\mathbf{c}}^*(\lambda, \mu)$ (described in Eq. (11a)). Given these values, we can then solve the dual problem by minimizing $h(\lambda, \mu)$ over $\lambda, \mu \ge \mathbf{0}$. There is no duality gap between the primal and the dual, because the capacity region Λ [2], [17] is a convex set.

In the following subsections, we describe the decomposition of the dual objective function that leads to the cross-layer optimization problem and we specify the optimal solutions by solving these independent subproblems.

A. Source rate control

Based on the dual decomposition the traffic rate vector of source node s is determined by the first maximization subproblem in Eq. (14a). $U_s(\cdot)$ is a strictly concave scalar function of the rate vector variable x_s. The maximization problem in (14a) is maximization of a concave function subject to the convex constraints (11d) and (11e). Thus, it has a unique solution. $U_s(\cdot)$ is continuously differentiable. Thus, the maximum will be given by the numerical solution of the equation

$$\nabla \mathcal{U}_s(x_s^*) = \mathcal{T}_s \lambda^s, \quad \forall s \in \mathcal{S} \tag{16}$$

as long as the resulting solution for x_s^* is in the interior of the constraint set defined by (11d) and (11e). Otherwise the solution will lie at the corners of the constrained set defined by (11d) and (11e).

B. Average End-to-End Delay Control

The second subproblem of the dual decomposition described in Eq. (14b) is related to average end-to-end delay control based on the optimal values for the link capacity margin $\sigma_{i,j}$. Eq. (14b) is a strictly convex, minimization problem, subject to the constraint that all sigma are nonnegative. Thus, it has a unique solution. Function $\phi(\cdot)$ is a continuously differentiable function. Hence, the optimal values of $\sigma^*_{i,j}$ are obtained by solving the equations numerically

$$\frac{d\phi}{d\sigma}(\sigma^*_{\mathbf{i},\mathbf{j}})\mu^{(i,j)} = -\lambda_{i,j}, \quad \forall (i,j) \in \mathcal{L} \tag{17}$$

C. Scheduling policy

The third problem of the dual decomposition determines the scheduling policy. The optimal value for the allocated link capacity $\hat{c}^*_{i,j}$ is given by the third term of Eq. (14c)

$$\hat{c}^*_{i,j} = \underset{\hat{c}_{i,j} \in \Lambda}{\text{argmax}} \sum_{(i,j)\in\mathcal{L}} \lambda_{i,j}\hat{c}_{i,j} \tag{18}$$

This scheduling subproblem is based on the *maximum weight* scheduling policy introduced in [7] and described in the extended version of the paper [20].

D. Distributed Algorithm

In this section, we describe the distributed algorithm that solves the network utility optimization problem. In order to solve the dual problem of Eq. (15), we use a subgradient descent iteration method [19] to update at each iteration n the dual variables (Lagrangian multipliers) as follows

$$\lambda^{(n+1)}_{i,j} = \left\{ \lambda^{(n)}_{i,j} - \gamma\left(\hat{c}^{(n)}_{i,j} - \right.\right. \\ \left.\left. - \sum_{s\in\mathcal{S}} \sum_{k:(i,j)\in p_{sk}} \left(t^{(i,j)}_{sk} x^{(n)}_{sk} \right) - \sigma^{(n)}_{i,j} \right) \right\}^+ \tag{19}$$

$$\mu^{(n+1)}_{sk} = \left\{ \mu^{(n)}_{sk} - \gamma\left(\mathcal{D}_s - \sum_{(i,j)\in\mathcal{L}} \phi(\sigma^{(n)}_{i,j})(R_s)_{k(i,j)} \right) \right\}^+, \tag{20}$$

where γ is a positive step-size that ensures convergence of the iterative solution (e.g. $\gamma = 0.01$) and $(v)^+ = \max(0, v)$ is the projection to the non-negative value.

Based on the primal and dual variable updates of Eq. (16), (17), (18), (19) and (20), we propose a distributed optimization algorithm, described below in Alg. 1.

VI. Simulation Results

In this section, we present simulation results for our trust-aware network utility maximization problem. Fig. 1 represents the sample wireless network scenario. The wireless network contains $\mathcal{N} = 8$ nodes and $\mathcal{L} = 11$ links, with maximum allowable capacity $c_{i,j}$ chosen in $[9, 11]$ *kbps*. There is one traffic flow from s to d_s, which allocates traffic to different routing paths. Our simulation time is $T = 160$ time slots. The end-to-end delay constraint for the traffic flow is $\mathcal{D}_s = 2$ *msec*. There are five different paths p_{sk}, where

Algorithm 1 Distributed Cross-Layer Optimization

```
INITIALIZE primal and dual variables
while 1^T |x_s^(n) − x_s^(n−1)| ≤ ε do
    Dual Variables Update
    Each link (i, j) updates its dual variable λ_{i,j} (Eq. (19)).
    Each source s updates the dual variables μ_{sk} (Eq. (20)).
    Sources exchange dual variables
    Each source s evaluates λ^{s,(n)}.
    Each source s computes its traffic rate vector x_s^(n) by solving Eq. (16)
    Each link (i, j) evaluates μ^{(i,j),(n)}.
    Each link (i, j) computes its σ_{i,j}^(n) by solving Eq. (17).
    Each node performs scheduling via Eq. (18) as in [7].
end while
```

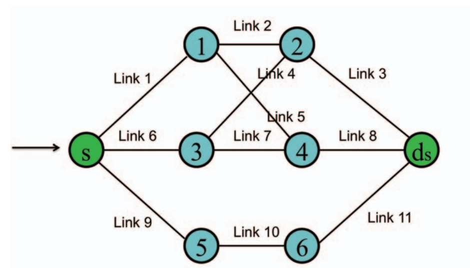


Fig. 1: Wireless Network Scenario

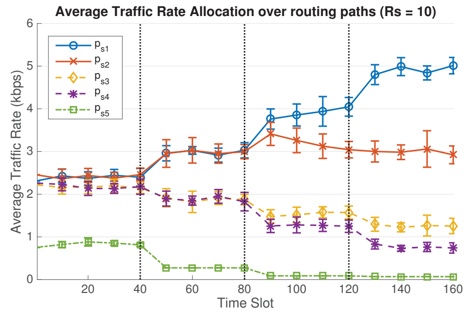
the source node s can allocate data traffic.

We define four trust *update periods* (each period is defined every T_{update} time slots), in order to show the behavior of our approach for different trust values. For the simulations, we define $T_{update} = T/4 = 40$ time slots. Node trust estimates ν_i change dynamically at every update period, based on the trust evaluation mechanism. The different node trust values from the trust evaluation mechanism for each of the four update periods are shown at the matrix below

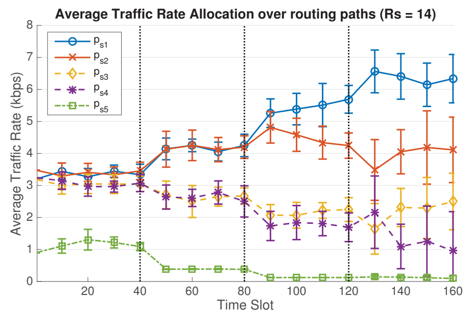
$$\nu = \begin{array}{c} \begin{matrix} s & 1 & 2 & 3 & 4 & 5 & 6 & d_s \end{matrix} \\ \begin{pmatrix} 1 & 1 & 1 & 0.7 & 1 & 0.7 & 0.5 & 1 \\ 1 & 0.9 & 0.9 & 0.5 & 0.9 & 0.2 & 0.2 & 1 \\ 1 & 0.9 & 0.9 & 0.3 & 0.7 & 0.1 & 0.1 & 1 \\ 1 & 0.9 & 0.9 & 0.2 & 0.5 & 0.1 & 0.1 & 1 \end{pmatrix} \end{array} \tag{21}$$

Trust values are adjusted using the EWMA algorithm expressed in Eq. (4), in order to prevent significant variations in the trust estimates over subsequent trust update periods. For our simulation, the EWMA algorithm uses $\alpha = 0.8$ to give more significance to the latest update.

Given the trust values estimates in Matrix (21), we can notice that path p_{s5} contains untrusted (malicious) nodes and should ideally be excluded from the traffic rate assignment. In addition, node 3 is detected to be malicious and hence our mechanism should ideally assign significantly less traffic to the paths p_{s3} and p_{s4} that contain this node. Finally, node 4 obtains a low trust value estimate at the last update period, which should lead to decrease in the traffic rate assignment



(a) Average Traffic Rate with $\mathcal{R}_s = 10$



(b) Average Traffic Rate with $\mathcal{R}_s = 14$

Fig. 2: Average Traffic Rate over paths for different maximum rates $\mathcal{R}_s$

even for path p_{s2} that contains this node.

Figures 2a and 2b present the numerical results of the average traffic rate allocation for two different cases of maximum allowable traffic rate $\mathcal{R}_s$ (with the corresponding error bars). In the case of $\mathcal{R}_s = 10$ *kbps*, the maximum traffic rate is close to the maximum allowable capacity of the wireless links, while in the case of $\mathcal{R}_s = 14$ *kbps*, the maximum traffic rate is greater than the maximum capacity of the links. We observe that in both cases the traffic rate assigned to each routing path changes at every update period based on the trust estimates. Our algorithm assigns to the path p_{s1} the maximum traffic rate, since it contains trusted nodes and to the path p_{s5} the lowest traffic rate, because it consists of untrusted nodes. For the rest of the paths, the traffic rate is being adjusted according to trust estimates. We also observe that in the case of $\mathcal{R}_s = 14$, more traffic rate is allocated to untrusted paths to cover the demand. Some numerical results for the link capacity margins are presented in the extended version [20].

VII. Conclusion and Future Work

In this paper, we investigated an important application of performance and security tradeoff by introducing security considerations in the cross layer design of network protocols via network utility maximization (NUM). The specific concept of security we used is *trust*. Users get higher utility by transmitting data through nodes of higher trust values. Thus, trust values should be taken into account as parameters in the optimization problem, so that the resulting trust-aware protocols are resilient to network failures and to possible attacks. We also incorporated delay constraints in the utility optimization problem to capture QoS requirements. Finally, we proposed a distributed algorithm that achieves network utility maximization. As part of future work, we plan to investigate scenarios with dynamic changes in trust values, and to evaluate our approach in large scale scenarios.

References

[1] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, “Layering as optimization decomposition: A mathematical theory of network architectures,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 255–312, January 2007.

[2] L. Chen, S. Low, M. Chiang, and J. Doyle, “Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks,” in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, April 2006, pp. 1–13.

[3] E. Stai, S. Papavassiliou, and J. Baras, “Performance-aware cross-layer design in wireless multihop networks via a weighted backpressure approach,” *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–1, 2014.

[4] X. Lin, N. B. Shroff, and R. Srikant, “A tutorial on cross layer optimization in wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1452–1463, August 2006.

[5] G. Theodorakopoulos and J. Baras, “On trust models and trust evaluation metrics for ad hoc networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 318–328, Feb 2006.

[6] S. Marti, T. J. Giuli, K. Lai, and M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks,” in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*. Boston, Massachusetts, United States: ACM Press, 2000, pp. 255–265.

[7] F. Qiu, J. Bai, and Y. Xue, “Towards optimal rate allocation in multi-hop wireless networks with delay constraints: A double-price approach,” in *Proceedings of the IEEE International Conference on Communications (ICC)*, June 2012, pp. 5280–5285.

[8] D. P. Palomar and M. Chiang, “A tutorial on decomposition methods for network utility maximization,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1439–1451, August 2006.

[9] N. Trichakis, A. Zymnis, and S. Boyd, “Dynamic network utility maximization with delivery contracts,” in *in Proceedings of the IFAC World Congress*, 2008, pp. 2907–2912.

[10] M. H. Hajiesmaili, M. S. Talebi, and A. Khonsari, “Utility-optimal dynamic rate allocation under average end-to-end delay requirements,” *CoRR*, vol. abs/1509.03374, 2015.

[11] J. Baras, T. Jiang, and P. Purkayastha, “Constrained coalitional games and networks of autonomous agents,” in *Proceedings of the 3rd International Symposium on Communications, Control and Signal Processing (ISCCSP)*, March 2008, pp. 972–979.

[12] P. Tague, S. Nabar, J. Ritcey, and R. Poovendran, “Jamming-aware traffic allocation for multiple-path routing using portfolio selection,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 1, pp. 184–194, Feb 2011.

[13] T. Jiang and J. Baras, “Trust credential distribution in autonomic networks,” in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM)*, Nov 2008, pp. 1–5.

[14] U. M. Maurer, “Modelling a public-key infrastructure,” in *Proceedings of the 4th European Symposium on Research in Computer Security: Computer Security (ESORICS)*, London, UK, 1996, pp. 325–350.

[15] S. W. Roberts, “Control chart tests based on geometric moving averages,” *Technometrics*, vol. 42, no. 1, pp. 97–101, 2000.

[16] J.-W. Lee, M. Chiang, and A. Calderbank, “Price-based distributed algorithms for rate-reliability tradeoff in network utility maximization,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 962–976, May 2006.

[17] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu, “Impact of interference on multi-hop wireless network performance,” in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom)*, New York, NY, USA, 2003, pp. 66–80.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[19] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1999.

[20] E. Paraskevas, T. Jiang, and J. S. Baras, “Trust-aware network utility optimization in multihop wireless networks with delay constraints,” *CoRR*, vol. abs/1603.09278, 2016.