Cross-layer multipath congestion control, routing and scheduling design in ad hoc wireless networks

Mohammed Aljubayri | Zhaohui Yang | Mohammad Shikh-Bahaei

Abstract
Demands for stable and efficient transmission are increasing due to the ongoing expansion of global wireless data traffic. One of the most promising ways to moderate burden of traffic load and decrease the queuing delay is to design cross-layer protocols. This paper considers a utility maximization problem for multipath ad hoc wireless networks via joint cross-layer congestion control, routing and scheduling design. In the formulated utility maximization problem, rate, scheduling and queuing delay constraints are involved under the condition of fixed channels and time varying channels. To solve this utility maximization problem, two scheduling methods are proposed, that is, perfect scheduling and distributed scheduling. In perfect scheduling, all nodes in the network contribute to the scheduling process, while distributed method only takes into account adjacent nodes. For both methods, the global convergence is proved. A comprehensive simulation evaluation is performed which shows that this proposed algorithm outperforms existing cross-layer protocols in terms of increasing source rate and reducing congestion price.

1 | INTRODUCTION

With the recent advancements in wireless communication technologies and developments of internet of things (IoT), efficient multipath transmission protocols are gaining more interest [1]. A common example is the ubiquitous mobile communication terminals which support wireless communications, data processing, and rich multimedia [2]. These devices, which support multiple wireless networks such as WiFi, 5G cellular, WiMAX [3] are very common and suitable benchmarks for cross layer interaction and multipath congestion control. Another potential area is smart city which has played an important role in academic and industry fields (e.g. see [4]) in the last few years. It involves the development and deployment of different IoT-based infrastructures and wireless sensor networks with available multipath connection [5].

In an effort to address such demands, there is a growing interest on efficient cross-layer designs. Cross layer protocols have emerged to provide an integrated scheme to enable interaction, coordination and joint optimization among different layers [6]. For instance, the cross layer design in order to maximize user utility for internet congestion control in the wireless networks were studied in the literature [7–9]. In [10], physical (PHY) and medium access control (MAC) layers were taken into consideration to provide stable wireless connection among moving, portable and fixed stations. In [11], Rajesh and Gnanasekar studied congestion control in the heterogeneous wireless ad hoc networks by considering data priorities. In [12, 13], the authors studied joint routing, power allocation policy and the influence of interference on multi-hop wireless network performance. Similarly, in [14, 15], joint routing and scheduling problem were studied by excluding transport layer. In [16], Chen et al. proposed a joint design of congestion control, routing and scheduling for ad hoc wireless networks by considering network utility maximization.

The above works considered only single path connection, but ignored the availability of multi-path in the network. Multipath congestion control methods are usually developed as a joint design of transport layer and multipath routing at network layer [17]. In [18], Mishra et al. proposed a packet scheduling algorithm and formulated the multipath TCP problem, but they neglected the routing problem of the network layer.

In addition, it is practical to consider queuing delay when designing cross-layer protocols. Queuing delay is defined as the waiting time in a queue for a packet to be served [19]. The number of waiting packets in the buffer are counted by each
separate node. This will allow an estimate of the expected queuing delay [20]. As a result, fewer packets in the queues lead to smaller end-to-end delays [21]. Reducing such delay can also be beneficial in load-balancing schemes to prevent congestion [22]. Thus, end-to-end queuing delay in multipath protocols plays an important role to provide a robust connection and improve the network performance.

In this paper, a utility maximization problem is considered for multipath ad hoc wireless networks via joint cross layer congestion control, routing and scheduling. We extend the work in [16] to adapt multipath transmission and queuing efficiency. The formulation of the optimisation problem with respect to source rate, scheduling and queuing delay constraints in fixed and time varying channels are presented. Duality theory is employed in this study to solve the optimisation problem and two methods of scheduling namely, Perfect Scheduling and Distributed Scheduling are proposed. All nodes in perfect scheduling algorithm participate in the scheduling process. In distributed scheduling, neighbor nodes only contribute to the scheduling. Moreover, the global convergence is mathematically proved for the above methods. Simulation results show that our algorithms outperform existing schemes in terms of enhancing source rate and decreasing congestion price.

The remainder of this paper is organized as follows: Section 2 reviews some background and related works available in the literature. In Section 3, we describe our system model and its mathematical description. In Section 4, we propose our cross-layer design in fixed channels and derive the algorithm convergence. In Section 5, we further analyze the algorithm convergence based on our proposed cross-layer design in time varying channels. Section 6 reports our simulation results. Finally, we conclude in Section 7.

2 BACKGROUND AND RELATED WORKS

After introducing Internet Protocol (IP) and Transmission Control Protocol (TCP) in the 1970s, they have been investigated in the literature due to their major functionality of exchanging data packets and providing a reliable byte stream service in [23] [24]. However, single-path transport protocols such as TCP suffer from several efficiency-related issues, for example, failure to support multihomeing and bandwidth aggregation [17]. Multihoming is the procedure of exploiting multiple hardware facilities, such as a mobile phone with both WiFi and 5G interfaces, to transmit data simultaneously. Due to this motivation, Multipath TCP (MPTCP) were proposed to provide efficient data transmission in such environments [25]. Also, Stream Control Transport Protocol (SCTP) was developed to adapt multipath transmission [26]. Multipath protocols also adapt fault tolerance, that is, different paths can be assigned for different tasks. For instance, the path with the highest throughput can be used for data transmission while the slow paths could be utilised for backup process [27]. Moreover, it has been shown that the connection-level throughput region of multi-path routing/congestion control algorithms can be larger than that of a single-path congestion control scheme [28].

Various researches have been proposed to develop multi-path routing/congestion control problem, including Host Identity Protocol and Mobile IP [29]. However, these solutions usually have negative impacts on the congestion price, scheduling and the possible conflict with regular TCP for existing applications. The approaches to address the multipath congestion control issues can be divided into two major categories. First, the application based approach where software is developed and tested on an applied benchmark while neglecting the mathematical formulation, convergence and stability analysis of the algorithm. For instance, a Linux-based software was developed in [25]. As another example, in [30], Cui et al. proposed a fountain code based multipath TCP algorithm and they report the validation of the algorithm through simulations. The main idea behind this algorithm is to use the random nature of fountain code in transmitting data through different subflows. There are also some experimental studies associated with multipath Transmission Control Protocols in [31] which can be considered within this category.

The second approach utilizes the mathematical formulation, graph theory and optimization problems for a given network topology. As a result, the stability of the algorithm and its convergence to the optimal solution are mathematically analyzed and discussed. For example, in [16], a jointly optimal design of cross layer congestion control, routing and scheduling was proposed for single path connection in the context of ad-hoc wireless networks. In [32], Eryilmaz and Srikan analysed the stability of the congestion control by using LaSalle’s invariant principle. In [33], Zhou et al. proposed a cross-layer design considering the rate control, the medium access control and the routing problem for the cooperative vehicular ad hoc network. Considering the flow rate allocation [34], Ploumidis et al. proposed a distributed scheme to maximize the aggregate flow throughput. The multi-hop networks utility maximization was studied in [19][35][36]. Nevertheless, those literature are mainly limited to single rate congestion control. To the best of our knowledge, there are few works focusing on multipath congestion control using the second approach. In [37], Vo et al. studied the congestion prices of a multipath TCP network; however, the joint congestion control, routing and scheduling is not considered. This paper follows the second approach and consider joint congestion control, routing and scheduling design for multipath ad hoc wireless networks.

3 SYSTEM MODEL

We consider an ad hoc wireless network including a set of $N$ nodes and a set of $L$ links. We assume that these links are directed with symmetric connectivity, that is, link $(i, j) \in L$ if and only if $(i, j) \in L$. Each link $l \in L$ has a fixed link capacity $c_l^i$. We further assume that the nodes are randomly distributed in the network area while each node has $\zeta$ number of available channels for transmission.

Due to scarce resource in wireless spectrum, connection links compete with each other to obtain access to the channel. We

1 This assumption will be relaxed in Section 5.
The physical meaning of such constraint is the capacity allocated to the flows through all links \((i, j)\) and available channels \(z\) to destination \(k\) should meet the feasible rate region's requirements at the MAC layer. Let \(x^k_i\) indicate the network flow created at node \(i\) to the destination \(k\). The difference between the aggregate capacity for all its outgoing flows and the aggregate capacity for all its incoming flows should be greater or equal than \(x^k_i\). Mathematically speaking, this constraint can be expressed as

\[
x^k_i \leq \sum_{z \in \mathcal{Z}} \sum_{j : (i, j) \in L} f^k_{i, j, z} - \sum_{z \in \mathcal{Z}} \sum_{j : (j, i) \in L} f^k_{j, i, z}
\]

We consider multipath transmission and M/M/m queuing system as the queuing delay model (see Figure 2). In M/M/m queuing system, \(m\) represents the number of servers allocated for customer service. This should satisfy the condition of \(m \geq 1\). We assume the number of servers \(m\) denotes the number of transmitters in each node. The packets arrival in M/M/m follows a Poisson process with mean arrival rate of \(\lambda_0\). In our model, \(\lambda_0 = \sum_i x^i_j\) which is the sum of incoming flow rate to node \(i\). The service time \(\mu_0\) is exponentially distributed with mean value of \(1/\mu_0\). Hence, the mean stay time for a packet can be expressed as [20]

\[
t_i = \frac{1}{\sum_j x^i_j} (L_q + m \rho)
\]

\[
= \frac{1}{\sum_j x^i_j} \left( \frac{(mp)^m \rho}{m!l(1-\rho)^2} \right) \mu_0 + m \rho \right), \rho < 1,
\]

where

\[
\rho = \sum_j x^i_j \frac{1}{m \mu_0},
\]

and \(L_q\) denotes the queue size. The stay time includes waiting and service times. As mentioned earlier, nodes have number of channels available for transmission. Since we are considering multipath transmission, the number of channels \(\mathcal{Z}\) is the number of all paths \(\mathcal{Z}\). Let \(s \in \mathcal{S}\) indicate the set of source nodes. Every node can be whether a traffic source node \(s\) or a relay node \(r\).
that forwards packets to the required destination. The number of traffic flows that pass the node is defined as Pass Route Number (PRN). Thus, the mean value of PRN can be defined as \[ sr = \frac{Zn(\tilde{m} - 1)}{n} = Z(\tilde{m} - 1), \] where \( \tilde{m} \) and \( n \) are the average number of hops from source to destination and the total number of nodes in the network, respectively. Furthermore, concurrent transmission ratio \( v \) is defined as the concurrent transmission capacity \( G \) over the total number of nodes in the network (i.e. \( n \)) and can be expressed as \[ v = \frac{G}{n}. \] Therefore, the packets’ arrival follows a Poisson process at each relay node with arrival rate of \( sr \sum_{j} x_j^f \). However, if the node is a destination node, the arrival of packets also follows a Poisson process but with arrival rate of \( Z \sum_{j} x_j^f \) since they can be separated. The maximization over \( x_j^f \) and \( f^k \) can be performed independently in the primal problem (i.e. (11)).

\[ D_j = \frac{1}{Z\sum_j x_j^f}\left(\frac{m(1-\rho)}{n}\rho \right) Z_0 + \frac{m\rho}{\rho - 1}, \quad \rho < 1, \] where \( \tilde{m} = Z \sum_j x_j^f \). Since \( \tilde{m} \) denotes the average number of hops from source to destination, the end-to-end queuing delay can be expressed as \( \sum D_j \).

The delay constraint in (11) can be shown to be equivalent to a linear constraint, that is, \( \sum x_j^f \leq E_j^k \), where \( E_j^k \) is a function of \( t_{jd} \). Note that all the constraints in (11) are linear. Further, the utility function is concave, thus, the objective function in (11) is concave as well. As a result, problem (11) is a convex function, which can be solved to its optimality via the dual theory.

4 | CROSS-LAYER DESIGN IN FIXED CHANNELS

The congestion control problem plays a critical role in the multipath transport protocols. The optimality and stability of the proposed algorithm for solving the congestion control problem are an important issue that should be considered. It should be noted that the design of the multipath congestion control algorithm depends on the problem under consideration. In fact, the utility function \( U_i(x_i) \) is not a function of the flows therefore, they can be separated. The maximization over \( x_i \) and \( f^k \) can be performed independently in the primal problem (i.e. (11)). Thus, the problem with dual configuration can be expressed as

\[ \min_{\rho \geq 0} Du(\rho), \] where \( \rho \) denotes a set of consensus variables and \( p_i^k \) is Lagrange multiplier for node \( i (i \in N) \) and destination \( k (k \in D) \).

To perform the congestion control and joint routing and scheduling, we consider the following dual function:

\[ Du(\rho) = \max_{x_i \geq 0, f^k \geq 0} \left\{ \sum_{i} U_i(x_i) - \sum_{i \in N, j \in D, k} p_i^k \cdot \left( x_j^k - \sum_{\xi \in E \cap (i,j) \in E} f^k_{i,j,\xi} - \sum_{\xi \in E \cap (j,k) \in E} f^k_{j,k,\xi} \right) \right\} \] subject to \( f \in R \).

Considering that a flow generated in the system can only be emitted from components of source nodes set, (13) can be written as

\[ Du_1(\rho) = \max_{x_i \geq 0} \left\{ \sum_{i} U_i(x_i) - \sum_{i \in N, j \in D, k} p_i^k \cdot \left( x_j^k - \sum_{\xi \in E \cap (i,j) \in E} f^k_{i,j,\xi} - \sum_{\xi \in E \cap (j,k) \in E} f^k_{j,k,\xi} \right) \right\}, \] considering that a flow generated in the system can only be emitted from components of source nodes set, (13) can be written as
where $\rho = \{p, \Sigma q^k\}$ and $p_i$ denotes the multipliers $p^k_i$ if $[i, k] \in S \times D$, and

$$D_{U_2}(\rho) = \max_{f_{i,j}, k \geq 0} \left\{ \sum_{i \in \mathbb{X}} \sum_{j \in \mathbb{Y}} f_{i,j,\zeta}(p^k_i - p^k_j) \right\}$$

s.t. $f \in \mathbb{R}$.

This paper employs the dual decomposition to perform cross-layer optimization. Therefore, according to (14), the congestion control can be expressed as

$$x_i(p) = U^{-1}_r(p, \sum q^k_i),$$

in which a unique maximizer regarding the queuing delay is subjected. Now, we first elaborate on the scheduling, routing and then adjust the congestion objective based on the local evacuation at each node. After that, the transport layer sources will be configured. It should be noted that in (15), $(i, j)$ is a bilateral node pair in $L$. In other words, if $(i, j) \in L$, consequently $(j, i) \in L$ and vice versa. Further, each link capacity is assigned if and only if sequence of $(i, j)$ and $\zeta$ are specified. Hence, (15) can be written as

$$D_{U_2}(\rho) = \max_{f_{i,j}, k \geq 0} \left\{ \sum_{i \in \mathbb{X}} \sum_{j \in \mathbb{Y}} f_{i,j,\zeta}(p^k_i - p^k_j) \right\}$$

s.t. $f \in \mathbb{R}$.

In order to undertake scheduling, the first step is to search through destinations. The optimal destination for each joint scheduling can be written as $k^* \in \arg \max_k \{E \sum \Sigma q_j^k - \rho_i^k\}$. Let $d_{i,j}$ denote the destination $k$ differential congestion price over nodes $i, j$. Considering $k^*$, we have $d_{i,j} = \rho_i^k - \rho_j^k$. Therefore, the scheduling demands can be expressed as

$$j \in \arg \max_{j \in \mathbb{R}} \sum_{i \in \mathbb{X}} \sum_{(i,j) \in \mathbb{Y}} d_{i,j} f_{i,j,\zeta}$$

Choosing an extreme point maximizer, each routing link launches packets toward the optimal destination as follows:

$$f_{i,j,\zeta}^k = \begin{cases} j \in \mathbb{X} & \text{if } k = k^* \\ 0 & \text{if } k \neq k^* \end{cases}$$

Noting that (17) is a piece-wise linear and not differentiable, hence, the dual function in (13) is not differentiable either. By

### Algorithm 1: Perfect scheduling over network with fixed channels

1: for each time $t$ do
2: Update price for each node $i$ to destination $k$ according to (20).
3: Each node sends its price information to its neighbors for each source $i$ do
4: $x_i(p) = U^{-1}_r(p, \sum q^k_i)$
5: end for
6: Assign respective capacity to each link according to $\gamma \in \arg \max_j \sum_{i \in \mathbb{X}} \sum_{(i,j) \in \mathbb{Y}} d_{i,j} f_{i,j,\zeta}$
7: According to assigned capacities, generate the routing at each link toward the considered destination.
8: end for

Introducing the following function:

$$\Theta_{i,j,\zeta}(p) = \left( \sum_{i \in \mathbb{X}} \sum_{j \in \mathbb{Y}} f_{i,j,\zeta}(p) - \sum_{i \in \mathbb{X}} \sum_{j \in \mathbb{Y}} f_{i,j,\zeta}(p) - x_i^k(p) \right)$$

$$- \sum_{\in \mathbb{X}, \in \mathbb{Y}, \in \mathbb{Y} \neq k} q_j^k \left( \sum_{i \in \mathbb{X}} z_i - E_i^k \right)$$

as a subgradient of the dual function and utilizing subgradient method [39], the congestion price of each node can be expressed as a non-negative real value of

$$p^k_i(t + 1) = \left[ \left( p^k_i(t) + \gamma \left( x_i^k(p(t)) \right) \right) - \sum_{(i,j) \in \mathbb{Y}} f_{i,j,\zeta}(p(t)) \right]_+$$

where $\gamma$ is a positive scalar step size and "+" is the projection of $\mathbb{R}^+$ set [16]. Therefore, we solved the dual problem for the joint optimal design of cross-layer congestion control, scheduling and routing. The pseudo-code of the proposed solution for perfect scheduling is outlined in Algorithm 1.

#### 4.1 Algorithm convergence

In this section, we investigate the convergence of our proposed algorithm. Suppose $\tilde{p}(t) := \frac{\sum_{i=1}^n p_i(t)}{n}$ is the average congestion
price at time $t$. The convergence can be obtained by proving

$$\limsup_{t \to \infty} Du(\bar{p}(t)) - Du^*(\bar{p}^*) < \varepsilon : \text{for any positive } \varepsilon,$$

(21)

real value of $\varepsilon$,

where $Du^*$ and $\bar{p}^*$ denote the optimal target and optimal price, respectively. Recalling the subgradient method, for a small enough step size $\gamma$, we have

$$\|p(t+1) - \bar{p}^*\|_2^2 \leq \|p(t) - \bar{p}^*\|_2^2 - 2\gamma \bar{\vartheta}(\bar{p}(t))^T \bar{\vartheta}(p(t) - \bar{p}^*) + \gamma^2 \|g(p(t))\|_2^2$$

we can rewrite (22) as

$$\|p(t+1) - \bar{p}^*\|_2^2 \leq \|p(t) - \bar{p}^*\|_2^2 - 2\gamma (Du(p(t)) - Du^*) \|p(t) - \bar{p}^*\| + \gamma^2 \|g(p(t))\|_2^2$$

(23)

As $\|p(t+1) - \bar{p}^*\|_2^2 > 0$ and according to subspace in each recursive duration along the corresponding dual space, (23) can lead to

$$Du(p(t)) - Du^*(\bar{p}^*) \leq \frac{\|p(t) - \bar{p}^*\|_2^2}{2\gamma t} + \frac{\gamma \sum_{n=1}^{t} \|\bar{\vartheta}(p(n))\|_2^2}{2t}.$$  

(24)

If $\|\bar{\vartheta}(p(n))\|_2 \leq Q$, and since $Du$ is a convex function, by Jensen’s inequality, we can conclude that $Du(p(t)) - Du^*(\bar{p}^*) \leq \frac{\gamma Q^2}{2}$. In other words, we proved that the algorithm converges statistically to within $\frac{\gamma Q^2}{2}$ of the optimal value. Hence, according to (16), convergence of the congestion price is guaranteed.

4.2 Distributed scheduling

Distributed scheduling in a multipath network can play a significant role in the robustness of the wired or wireless network, specifically when traffic shifting to a better path is required [40]. For example, a flow which operates along multiple paths at the same time could be designed using traffic scheduling mechanisms which allow the shifting of traffic to those paths with good delivery conditions (e.g., high bandwidth, low delay and packet loss rate) [17].

The scheduling demands in (18) is to maximize the weighted sum of the capacity of links. Equivalently, this problem can be solved by weighted matching problem rather than finding the maximum weighted of all independent set in the conflict graph. In a graph, a subset of links which do not share common nodes is referred to matching set. Maximum weighted matching problems have polynomial time complexity (see, e.g., [41]) with centralized implementation requirement, that is, each node should announce its weight to a central node to construct the weighted graph of the network topology. As an alternative method, we utilize a sequential greedy algorithm introduced in [42]. In order to convert (18) to a maximum weighted matching problem, we consider $d_{i,j} > 0$ if $d_{i,j} < 0$ for all $z \in Z$. From each directed bilateral link pair, we extract an undirected link $\langle i, j \rangle$ and its weight can be expressed as

$$d_{i,j} = \max \{ w_{i,j}(i,j), w_{j,i}(j,i) \}.$$  

Let $L^{und}$ and $D^{und}$ denote the set of undirected links and its set of weight, respectively. Now, this problem is is equivalent to maximum weighted matching problem over the graph $G^{und} = (N, L^{und}, D^{und})$ [19].

While the scheduling process in ad hoc networks usually proceed in each time step, the time complexity and generation of multi-hop scheduling for a deterministic solution are controversial [43]. On the other hand, the greedy behavior of wide domination of wireless interactions requires the study of detection in compromised nodes and overall utilization of advanced statistical methods [44][45]. Moreover, developing dynamic of channel residual reaction in steady state modes should be considered to expand efficiency of ad hoc networks scheduling [46][47]. This paper aims to take advantage of the concepts represented in [16][47][48] for distribution scheduling algorithm, which is further detailed in Algorithm 2.

Remark: The scheduling algorithm needs to cope with the hierarchy of search along with the difference of node to node price in a region starting from source set and ending up in the destination set. In addition, seeking an optimal solution would be done according to the greediness of the algorithm and the bearable congestion threshold. Furthermore, being a subject to such optimization problem, it is necessary to make a balance between greediness of the algorithm and absolute optimality of the system with respect to computational complexity. When the primal problem is broken down into aforementioned subproblems in the cross-layer design which is a generic sequence throughout requisite solutions for (17) and (14), it is important to avoid an exponential increase of iteration steps. Hence, in this scheduling problem, it is recommended to make a balance between the greedy algorithm and the dynamic programming presented in [49].

5 CROSS-LAYER DESIGN IN TIME-VARYING CHANNELS

Multichannel networks involve multi-rate devices. In such cases, the allocated capacity for each link is time variant. We assume the state of the channels is fixed within a time slot. We also assume that the channel states change independently among different time slots. Let $e(t)$ denote the state of the channel in time slot $t$. The capacity of link $l$, at each time slot $t$ is denoted by
Algorithm 2 Distributed scheduling over network with fixed channels

1: for each time $t$ do
2: \hspace{1em} Calculate $d_{(i,j)}$ for all edges where $d_{(i,j)} = \max\{w_{(i,j)}(q_{(i,j)}), w_{(i,j)}(t_{(i,j)})\}$
3: \hspace{1em} for all $\xi \in Z$ do
4: \hspace{2em} find node $j^*$ with free neighbour $j$.
5: \hspace{1em} end for
6: \hspace{1em} if each $i$ received a matching request from corresponding $j^*$, then
7: \hspace{2em} $(i, j^*)$ is a matched link and node $i$ sends a matched reply to $j^*$. Consequently, $i$ will drop all other messages.
8: \hspace{1em} end if
9: \hspace{1em} if $i$ did not receive any message then
10: \hspace{2em} send a matching request to $j^*$
11: \hspace{1em} end if
12: for all $\xi \in Z$ when receiving a matching request from neighbour $j$ do
13: \hspace{2em} choosing the corresponding link as a matched link and node $i$ sends a matched reply to $j^*$. Consequently, $i$ will drop all other messages.
14: else
15: \hspace{2em} saves the message.
16: end if
17: end for
18: for all $\xi \in Z$ do
19: \hspace{2em} when receiving a matched reply from neighbour $j$, link $(i, j^*)$ is confirmed by node $i$ to be a matched link and a drop message is sent to all free neighbours.
20: end for
21: for all $\xi \in Z$ do
22: \hspace{2em} when receiving a matched reply from neighbour $j$, node $i$ removes node $j$ form the list of free neighbours.
23: end for
24: Repeat steps 3 to 8 unless node $i$ is matched or no free neighbors are available.
25: All matched links are ready for transmission. The corresponding nodes will use $d_{(i,j)}$ to schedule the directed links for transmission.
26: end for

$c_i(e)$, therefore, (1) can be revised as
\[
b_f^j(t) := \begin{cases} c_i(e) : & \text{if } l \in b \\ 0 : & \text{otherwise} \end{cases}
\]

(25)

Recalling our earlier discussion in Section 3, the feasible rate region at link layer is identified as a convex hull of the rate vectors. Let $R(e)$ denote the feasible rate region. Thus, using a finite channel state assumption and a finite channel process in the identical distribution $u(e)$, the mean rate region can be defined as
\[
R := \left\{ \bar{b} : \bar{b} = \sum_e u(e) \cdot b(e), b(e) \in R(e) \right\}.
\]

(26)

Following the steps provided in Section 4, we can state a dual parallel to problem statement in (11) and (12) by
\[
D_n(p) = \max_{x_i \geq 0, f^k_{(i,j)} \geq 0} \left\{ \sum_i U_i(x_i) - \sum_{\xi \in Z, j \in [j(i)]} p^k_{(i,j)} \right. \\
\left. \times \left( x_i - \sum_{\xi \in Z, j : (j(i))} f^k_{(i,j)} - \sum_{\xi \in Z, j : (j(i))} f^k_{(i,j)} \right) - \sum_{\xi \in Z, j \neq k} q^k_{(i,j)} \left( \sum_i x_i - E^k_i \right) \right\}.
\]

(27)

5.1 Prefect scheduling over time varying channels

The generated stream from the source set will comply accordingly from the utility function as congestion price measures linked sets of channels. This time variation in the system model require stability evaluation for the proposed congestion control method. Here, for simplicity, we employ the floor function which returns the largest integer less than or equal to the actual value of the price.

For each node, it is necessary to have the information of its neighbor nodes. In a long enough coherence time, with the small time step specified by simulation, channel states can be considered as fixed during the time slot.

In our algorithm, at time $t$, each node $i$ updates its price with respect to destination $k$ and channel $\xi$ as follows:
\[
p^k_{(i,k)}(t+1) = \left[ \left( p^k_{(i,k)}(t) + \gamma \left( x_i^k(p(t)) - \sum_{\xi \in Z, j : (j(i))} f^k_{(i,j)}(p(t)) \right) \right) \\
- \sum_{\xi \in Z, j \neq k} q^k_{(i,j)} \left( \sum_i x_i^k - E^k_i \right) \right] +
\]

and announces the price $p^k_{(i,k)}$ to all of its neighbor nodes. Here, $\lfloor \cdot \rfloor$ denotes the floor function. In the congestion control, each source node $s$ modifies its sending rate for $i$ based on the local congestion price which can be written as
\[
x_i^k(p) = U_i^{-1} \left( p + \sum_{\xi} q^k_{(i,j)} \right).
\]

Node $i$, after collecting the congestion price information from its neighbor node $j$, identifies the destination such
that \( k^* \in \arg \max_k \sum_{i \in \mathcal{E}} (q^k_i - \rho^k_j) \) and obtain \( d_{i,j} = \sum_{i \in \mathcal{E}} (q^k_i - \rho^k_j) \). Then, it will assign respective capacity to each link according to \( j^* = \arg \max_{j \in \mathcal{E}(i)} \sum_{i \in \mathcal{E}} d_{i,j} f_i(t) \) for each node according to the congestion process, recursive time variant scheduling can be generalised. The configuration of the network which forms the congestion process, recursive time variant scheduling can be summarized in Algorithm 3.

### Stability and convergence

Since the global parameters are fixed, it is necessary to point out the stability. Considering the dual optimization problem in (27) and using Markov chain as the stochastic model for \( p \), the price can be considered as a random variable. Our aim is to prove that the Markov chain is non-explosive and stable. Consider the following function of \( V \) at each time slot where the conditional expected value by any Lyapunov function \( V \) is given by [50]

\[
E[V(p(t+1)) - V(p(t))|p(t)]
\]

In addition, for each \( p(t) \) during the time slot \( t \) is constant, so we should consider \( p(t+1) = p(t) - \gamma \hat{\vartheta}(p(t)) \). The conditional expected value can be written as:

\[
E[V(p(t+1)) - V(p(t))|p(t)] = E[V((p(t)) - \gamma \hat{\vartheta}(p(t)))]
\]

\[
= V(p(t))|p(t)| - V(p(t))|p(t)|
\]

\[
\leq E[V(p(t)) - \gamma \hat{\vartheta}(p(t))]
\]

\[
- V(p(t))|p(t)|. \tag{29}
\]

Consider the Lyapunov function \( V(p(t)) = \| p(t) - \rho^* \|_2 \) and also define \( \hat{\vartheta} \) as the subgradient of (28), thus

\[
E[V(p(t+1)) - V(p(t))|p(t)]
\]

\[
= E[-\gamma \hat{\vartheta}(p(t))^T (2(p(t) - \rho^*)) - \gamma \hat{\vartheta}(p(t))]
\]

\[
- \gamma \hat{\vartheta}(p(t))^T (2(p(t) - \rho^*))
\]

\[
+ \gamma^2 E[\| \hat{\vartheta}(p(t)) \|_2^2 |p(t)|] \tag{30}
\]

Since, we have already proved that the stated subgradient is bounded, we can also say that \( \hat{\vartheta} \) is also bounded. Consequently, \( E[V(p(t+1)) - V(p(t))|p(t)] \) is bounded and the stability based on Foster’s criterion can be obtained [51]. In an infinite approach of \( p \) during time, \( D_u(E[V(p(t+1)) - D_u(p(t))]) \) is also bounded or, in other words, the referred Markov chain guarantees non-explosive behaviour and convergence to a boundary near the optimal value [52]. Therefore, along with the congestion process, recursive time variant scheduling can be generalised. The configuration of the network which forms our method can maintain the system with a practical evaluation and command. Moreover, the implemented effect of delay and considering the ambient space in both (27) and (13) would provide a more stable model as shown in the simulation results.

### 6 Simulation results

In this section, we evaluate the performance of our proposed cross layer design on a simple ad hoc network shown in Figure 3. We assume that there are two network layer flows, \( A \) to \( F \) and \( B \) to \( E \) with same utility function \( U_S(x_S) = \log(x_S) \). We set the step size of the presented algorithms as \( \gamma = 0.1 \). As we mentioned in our system model, we assume that all the links are bidirectional and able to carry the traffic in opposite directions. For the convenience of the reader, the network parameters are summarized in Table 1. As a bench mark, our simulation

2 To have a fair comparison, the ad hoc network under consideration is the same as the one in [10].
results include the performance of algorithms introduced in [16] and [37].

6.1 Prefect scheduling over fixed channels

In this sub-section, we study the performance of our proposed perfect scheduling algorithm (i.e. Algorithm 1) over fixed time channels condition. We assume that the capacity of links $C \rightarrow E$, $E \rightarrow C$, $B \rightarrow F$, and $F \rightarrow B$ are one unit and the rest of links have two units of capacity. We consider two distinct flows, AF and BE. Figure 4 illustrates the performance of our algorithm compared with the existing schemes in terms of normalised source rate and normalised congestion price. It is worth mentioning that the oscillating behaviour is caused by the non-differentiability of the dual function. The algorithm introduced in [37] always makes an effort to balance the congestion prices among its paths. In case of high congestion in one path, the allocated rate to this path will be reduced to minimise the congestion. However, it can be observed from Figure 4 that our approach outperforms the existing methods (i.e. [16] and [37]) for both flows in terms of source rate. The reason for such behaviour lies on the fact that our algorithm takes into account the queuing delay and multipath transmission in the communication system. Furthermore, as it is shown in Figure 4, our algorithm presents more stability and it quickly converges to the optimal value in comparison with other schemes. Besides, our proposed algorithm demonstrates lower normalised congestion price comparing with other methods. This is due to the fact that our algorithm benefits from the available multi-path in the network during the scheduling process.

In an effort to have an insight into average stream rate, Tables 2 and 3 presents the average stream rate of flows AF and BE employing our proposed algorithm. The first column represents the source nodes and the first row represents the destination nodes. These tables provide the information about the paths that have been used for each flow when employing our perfect scheduling algorithm.

6.2 Distributed scheduling over fixed channels

In this scenario, we investigate the performance of our proposed distributed scheduling algorithm (i.e. Algorithm 2) over the network under consideration with fixed time channels. Similar to the first scenario, we assume that the capacity of links $C \rightarrow E$, $E \rightarrow C$, $B \rightarrow F$, and $F \rightarrow B$ are one unit and the rest of links have two units of capacity. Figure 5 illustrates the performance of our algorithm compared with the existing schemes in terms of normalised source rate and normalised congestion price. It can be observed from Figure 5 that the fluctuation pattern is lower than prefect scheduling algorithm. However, comparing Figures 4 and 5 demonstrates that the distributed algorithm achieve less normalised source rate. This comes from the fact that the feasible rate region shrinks to a smaller set under the approximate scheduling. It should be noted that in the source rate comparison, the algorithm proposed in [37] performs better than our algorithm at the beginning of the simulation; nevertheless, our algorithm outperforms it over time. Moreover, the proposed algorithm presents more stability compared to [37]. In general, our proposed distributed scheduling algorithm shows higher source rate and lower congestion price compared to both [16] and [37] methods. The main reason behind the satisfactory performance is exploiting multipath transmission and considering an efficient queuing system in our cross-layer design. We further summarize the average link rates.
6.3 Perfect scheduling over time varying channels

In this sub-section, we explore the performance of our proposed perfect scheduling algorithm (i.e. Algorithm 3) over time varying channels condition. Here, we assume that the capacity of links $C \rightarrow E$, $E \rightarrow C$, $B \rightarrow F$, and $F \rightarrow B$ are uniformly distributed with 0.5, 1 and 1.5 units, whereas the rest of links have uniformly distributed over 1, 2 and 3 units of capacity. Figure 6 shows the performance of our algorithm with regard to the normalised source rate and normalised congestion price. As it can be observed, considering time varying condition leads to more fluctuations for both $BE$ and $AF$ flows. This is due to variation of channel states in this condition. However, our algorithm demonstrates more stable pattern and lower congestion price. In addition, the performance of our algorithm is better for $AF$ flow in comparison with the other schemes. Although $BE$ flow in [16] achieved higher source rate, ours algorithm shows lower oscillating behaviour. Further, the proposed algorithm outperforms the algorithm in [37] in both source rate and congestion price. Note that the average links capacity in this scenario is the same as fixed channel scenario; that is, the links’ capacity for $CE$, $EC$, $BF$, and $FB$ are one unit while the other links’ capacity are two units. It should be noted that benefit of available multi-path becomes more obvious in time varying condition. As it can be observed, the achievable sending rate for each flow is larger compared to the fixed channel condition. The corresponding average stream rate of $AF$ and $BE$ flows for perfect scheduling are summarized in Tables 6 and 7.

![Figure 5](image1.png)  
Normalised source rate and normalised congestion price for distributed scheduling over fixed time channels using Algorithm 2

![Figure 6](image2.png)  
Normalised source rate and normalised congestion price for perfect scheduling over time varying channels using algorithm 3

### Table 4: Average stream rate of AF flow using distributed scheduling algorithm

|   | A   | B   | C   | D   | E   | F   |
|---|-----|-----|-----|-----|-----|-----|
| A | 0   | 0.274 | 0.396 | 0 | 0 | 0 |
| B | 0   | 0 | 0 | 0 | 0.269 |
| C | 0   | 0 | 0 | 0.391 | 0 | 0 |
| D | 0   | 0 | 0 | 0 | 0 | 0.391 |
| E | 0   | 0 | 0 | 0 | 0 | 0 |
| F | 0   | 0 | 0 | 0 | 0 | 0 |

### Table 5: Average stream rate of BE flow using distributed scheduling algorithm

|   | A   | B   | C   | D   | E   | F   |
|---|-----|-----|-----|-----|-----|-----|
| A | 0   | 0 | 0 | 0 | 0 | 0 |
| B | 0 | 0 | 0 | 0.140 | 0.531 | 0 | 0.132 |
| C | 0 | 0 | 0 | 0 | 0 | 0.139 |
| D | 0 | 0 | 0 | 0 | 0.531 | 0 |
| E | 0 | 0 | 0 | 0 | 0 | 0.130 |
| F | 0 | 0 | 0 | 0 | 0 | 0 |
TABLE 7: Average stream rate of BE flow using perfect scheduling algorithm

|   | A  | B  | C  | D  | E  | F  |
|---|----|----|----|----|----|----|
| A | 0  | 0  | 0.241 | 0  | 0  | 0  |
| B | 0.241 | 0  | 0.241 | 0.245 | 0  | 0.081 |
| C | 0  | 0  | 0  | 0.433 | 0.056 | 0  |
| D | 0  | 0  | 0  | 0  | 0.673 | 0  |
| E | 0  | 0  | 0  | 0  | 0  | 0  |
| F | 0  | 0  | 0  | 0  | 0.081 | 0  |

FIGURE 7: Normalised source rate and normalised congestion price for distributed scheduling over time varying channels using algorithm 3.

6.4 Distributed scheduling over time varying channels

In this scenario, we study the performance of our proposed algorithm for distributed scheduling (i.e. Algorithm 3 with distributed scheduling Algorithm 2) under time varying channel condition. Similar to the previous scenario, we assume that the links’ capacity for $C \rightarrow E$, $E \rightarrow C$, $B \rightarrow F$, and $F \rightarrow B$ are uniformly distributed with $0.5$, $1$ and $1.5$ units, whereas the rest of links have uniformly distributed over 1, 2 and 3 units of capacity. Figure 7 illustrates the performance of our algorithm compared with the existing schemes in terms of normalised source rate and normalised congestion price. The method in [16] is better than our method as well as [37] in BE flow. However, our algorithm outperforms the existing methods in AF flow as well as congestion price measurement. Also, it shows more stability pattern. It can be observed that the source rate of our distributed scheduling algorithm in time varying channels is larger than the source rate in fixed channels. This is due to variation of channel capacities. We summarize the average link rates allocated to each flow with our distributed scheduling algorithm in Tables 8 and 9.

TABLE 8: Average stream rate of AF flow using distributed scheduling algorithm

|   | A  | B  | C  | D  | E  | F  |
|---|----|----|----|----|----|----|
| A | 0  | 0.467 | 0.194 | 0  | 0  | 0  |
| B | 0  | 0  | 0  | 0.466 | 0  | 0  |
| C | 0  | 0  | 0  | 0.190 | 0  | 0  |
| D | 0  | 0  | 0  | 0  | 0.800 | 0  |
| E | 0  | 0  | 0  | 0  | 0  | 0  |
| F | 0  | 0  | 0  | 0  | 0.010 | 0  |

TABLE 9: Average stream rate of BE flow using distributed scheduling algorithm

|   | A  | B  | C  | D  | E  | F  |
|---|----|----|----|----|----|----|
| A | 0  | 0  | 0  | 0  | 0  | 0  |
| B | 0  | 0  | 0  | 0.800 | 0  | 0.010 |
| C | 0  | 0  | 0  | 0  | 0  | 0  |
| D | 0  | 0  | 0  | 0  | 0.800 | 0  |
| E | 0  | 0  | 0  | 0  | 0  | 0  |
| F | 0  | 0  | 0  | 0  | 0.010 | 0  |

7 CONCLUSIONS AND FUTURE WORK

In this paper, we presented a model for a joint congestion control, routing and scheduling design for multipath ad hoc wireless networks. A utility maximization problem was considered with rate, scheduling and the end-to-end queuing delay constraints. We solved this joint optimization problem by dual decomposition approach. Two different algorithms, called perfect scheduling and distributed scheduling, were developed over network with fixed and time varying channels. We further analyzed and proved the convergence and the stability of these algorithms. A comprehensive simulation was performed for four different scenarios to evaluate the performance of each algorithm in terms of source rate and congestion price. The results were compared with previously proposed algorithms available in the literature. Simulation results revealed that our algorithm can be beneficial since it considers both the available multi-path connections and the queuing delay in its joint optimal design. Our approach provides higher source rates, faster convergence and lower congestion prices. Moreover, our approach leads to lower time complexity due to allocation of queuing delay as one of the constraints. Therefore, the presented design can be convenient for a wide range of ad hoc networks, for example, vehicular ad hoc network (VANET).

In the future, our aim is to investigate the number of channels for each node as a design parameter and set another constraint on the number of available channels. We expect to observe an improvement in the network performance and the user utility especially in wireless networks.
REFERENCES

1. Xu, C., et al.: Socially aware mobile peer-to-peer communications for community multimedia streaming services. IEEE Commun. Mag. 53(10), 150–156 (2015)
2. Setton, E., et al.: Cross-layer design of ad hoc networks for real-time video streaming. IEEE Wirel. Commun. 12(4), 59–65 (2005)
3. Cao, Y., Liu, Z., Yang, Y.: A centralized scheduling algorithm based on multi-path routing in winmax mesh network. In: 2006 International Conference on Wireless Communications, Networking and Mobile Computing, Wuhan, China, 2006, pp. 1–4
4. Alsamhi, S.H., et al.: Survey on collaborative smart drones and internet of things for improving smartness of smart cities. IEEE Access 7, 128125–128152 (2019)
5. Hosahalli, D., Srinivas, K.G.: Cross-layer routing protocol for event-driven M2M communication in IoT-assisted smart city planning and management: CWSN-eCOPM. IET Wireless Sens. Syst. 10(1), 1–12 (2019)
6. Zorba, N., Skianis, C., Verikoukis, C.: Cross Layer Designs in WLAN Systems. Troubador Publishing, UK (2011)
7. Arun, V., Balakrishnan, H.: Copa: Practical delay-based congestion control for the internet. In: 15th IONUSIM Symposium on Networked Systems Design and Implementation (NSDI) 18, pp. 329–342 (2018)
8. Goyal, P., et al.: Elasticity detection: A building block for internet congestion control. arXiv preprint arXiv:180208730 (2018)
9. Wang, Z., et al.: Tcp congestion control algorithm for heterogeneous internet. J. Netw. Comput. Appl. 68, 56–64 (2016)
10. IEEE Std. 802-11: IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification. IEEE, Piscataway, NJ (1997)
11. Rajesh, M., Gnanasekar, J.: Congestion control scheme for heterogeneous wireless ad hoc networks using self-adjust hybrid model. Int. J. Pure Appl. Math. 116(21), 519–536 (2017)
12. Askarzadeh, A., Gurbuz, O., Unliyurt, T.: Power allocation and routing for full-duplex multi hop wireless networks under full interference. Ad Hoc Netw. 82, 91–99 (2019)
13. Kim, H.W., Kachroo, A.: Low power routing and channel allocation of wireless video sensor networks using wireless link utilization. Ad Hoc Sens. Wireless Netw. 30 (2016)
14. Hajek, B., Sasaki, G.: Link scheduling in polynomial time. IEEE Trans. Inf. Theory 34(5), 910–917 (1988)
15. Kodialam, M., Nagan sekar, J.: Congestion control scheme for heterogeneous wireless ad hoc networks using self-adjust hybrid model. Int. J. Pure Appl. Math. 116(21), 519–536 (2017)
16. Akgun, K., Gurbuz, O., Unliyurt, T.: Power allocation and routing for full-duplex multi hop wireless networks under full interference. Ad Hoc Netw. 82, 91–99 (2019)
17. Kim, H.W., Kachroo, A.: Low power routing and channel allocation of wireless video sensor networks using wireless link utilization. Ad Hoc Sens. Wireless Netw. 30 (2016)
18. Hajek, B., Sasaki, G.: Link scheduling in polynomial time. IEEE Trans. Inf. Theory 34(5), 910–917 (1988)
19. Kodialam, M., Nagan sekar, J.: Congestion control scheme for heterogeneous wireless ad hoc networks using self-adjust hybrid model. Int. J. Pure Appl. Math. 116(21), 519–536 (2017)
20. Akgun, K., Gurbuz, O., Unliyurt, T.: Power allocation and routing for full-duplex multi hop wireless networks under full interference. Ad Hoc Netw. 82, 91–99 (2019)
21. Kim, H.W., Kachroo, A.: Low power routing and channel allocation of wireless video sensor networks using wireless link utilization. Ad Hoc Sens. Wireless Netw. 30 (2016)
22. Xu, C., Xu, M., Fu, X.: Delay-based congestion control for multipath TCP. In: 2012 20th IEEE International Conference on Network Protocols (ICNP), pp. 1–10. IEEE, Piscataway, NJ (2012)
23. Barre, S., Pasch, C., Bonaventure, O.: Multipath tcp: from theory to practice. In: International Conference on Research in Networking, pp. 444–457. Springer, New York (2011)
24. Pasch, C., Bonaventure, O.: Decoupled from tp, tcp is at last able to support multihomed hosts. ACM Queue 12(2), (2014)
25. Zhu, M., et al.: Belia: bandwidth estimate-based link increase algorithm for MPTCP. IET Netw. 6(5), 94–101 (2017)
26. Adhart, H., Dreibholz, T., Becke, M.: Sctp socket api extensions for concurrent multipath transfer. Draft-dreibholz-tsvwg-sctp-multipath-01 (2012)
27. Yang, J., et al.: Architectural Guidelines for Multipath TCP Development. (RFC Editor, 2011. RFC 6182). https://rfc-editor.org/rfc/rfc6182.txt
28. Han, H., et al.: Multi-path tcp: a joint congestion control and routing scheme to exploit path diversity in the internet. IEEE/ACM Trans. Netw. 14(6), 1260–1271 (2006)
29. Moskowitz, R., et al.: Host identity protocol (HIP) architecture. RFC 4423 (2006)
30. Cui, Y., et al.: Fm tcp: A fountain code-based multipath transmission control protocol. IEEE/ACM Trans. Netw. 23(2), 465–478 (2014)
31. Pasch, C., et al.: Experimental evaluation of multipath TCP schedulers. In: Proceedings of the 2014 ACM SIGCOMM Workshop on Capacity Sharing Workshop, pp. 27–32. ACM (2014)
32. Eryilmaz, A., Srikant, R.: Joint congestion control, routing, and MAC for stability and fairness in wireless networks. IEEE J. Sel. Areas Commun. 24(8), 1514–1524 (2006)
33. Zhou, L., et al.: Cross-layer rate control, medium access control and routing design in cooperative VANET. Comput. Commun. 31(12), 2870–2882 (2008)
34. Ploumidis, M., Pappas, N., Tragantitis, A.: Flow allocation for maximum throughput and bounded delay on multiple disjoint paths for random access wireless multihop networks. IEEE Trans. Veh. Technol. 66(1), 720–733 (2016)
35. Chiang, M.: Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control. IEEE J. Sel. Areas Commun. 23(1), 104–116 (2005)
36. Lauffer, R., et al.: A cross-layer backpressure architecture for wireless multihop networks. IEEE/ACM Trans. Netw. 22(2), 363–376 (2013)
37. Vo, P.L., et al.: Multi-path utility maximization and multi-path TCP design. J. Parallel Distrib. Comput. 74(1), 1848–1857 (2014)
38. Luo, H., Lu, S., Bhanghavan, V.: A new model for packet scheduling in multihop wireless networks. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp. 76–86. ACM, New York (2000)
39. Kuhn, H.W., Tucker, A.W.: Nonlinear programming. In: Traces and emergence of nonlinear programming, pp. 247–258. Springer (2014)
40. Raiciu, C., Handley, M., Wischik, D.: Coupled congestion control for multipath transport protocols. IETF RFC 6356 (2011)
41. Kleinberg, J., Tardos, E.: Introduction to algorithms. Cornell University Course Notes, COMS 482, (2003)
42. Preis, R.: Linear time 1/2-approximation algorithm for maximum weighted matching in general graphs. In: Meinel C., Tison S. (eds) STACS 99. STACS 1999. Lecture Notes in Computer Science, vol. 1563. Springer, Berlin, Heidelberg (1999). https://doi.org/10.1007/3-540-49116-3_24
43. Djahel, S., Naitabdessalam, F., Turgut, D.: Characterizing the greedy behavior in wireless ad hoc networks. Secur. Commun. Netw. 4(3), 284–298 (2011)
44. Meij, M.N., Ben.Othman, J.: Detecting greedy behavior by linear regression and watchdog in vehicular ad hoc networks. In: 2014 IEEE Global Communications Conference, 5032–5037. IEEE, Piscataway, NJ (2014)
45. Aloisio, A., Izzo, V., Ramponi, S.: Vlsi implementation of greedy-based distributed routing schemes for ad hoc networks. Soft Comput. 11(9), 865–872 (2007)
46. Lee, S.J., Gerla, M.: Dynamic load-aware routing in ad hoc networks. ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01CH37240), vol. 10, pp. 3206–3210. IEEE, Piscataway, NJ (2001)

47. Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. In: Mobile Computing, pp. 153–181. Springer, Berlin (1996)

48. Kanellopoulos, D.N.: Recent progress on qos scheduling for mobile ad hoc networks. J. Org. End User Comput. (JOEUC) 31(3), 37–66 (2019)

49. Kirk, D.E.: Optimal Control Theory: An Introduction. Courier Corporation, North Chelmsford, MA (2012)

50. Asmussen, S.: Applied probability and queues, vol. 51. Springer Science & Business Media, Berlin (2008)

51. Meyn, S.P., Tweedie, R.L.: Stability of markovian processes iii: Foster-lyapunov criteria for continuous-time processes. Adv. Appl. Probab. 518–548 (1993)

52. Tassiulas, L., Ephremides, A.: Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multi-hop radio networks. In: 29th IEEE Conference on Decision and Control, pp. 2130–2132. IEEE, Piscataway, NJ (1990)

How to cite this article: Aljubayri M, Yang Z, Shikh-Bahaei M: Cross-layer multipath congestion control, routing and scheduling design in ad hoc wireless networks. IET Commun. 2021;15:1096–1108 https://doi.org/10.1049/cmu2.12145