An Approach to Minimal Power Routing and Scheduling in Wireless Ad Hoc Networks

Sugata Hazarika* Jim Kurose* Dennis Goeckel†
*Department of Computer Science †Department. of Electrical Engineering
University of Massachusetts, Amherst
*{hazarika, kurose}@cs.umass.edu †goeckel@ecs.umass.edu

***Abstract*— In this paper we present a distributed routing algorithm that minimizes total power spent by all the nodes in the network in routing packets from source to destination over wireless hops. Power spent by each node is a function of the data rate at that node and the interference experienced by the node. Nodes are capable of tuning their transmission power to support the incoming data rate in the presence of interference. The key idea of the algorithm is to route flows so that links with lower interference and hence lower power requirements are better utilized.**

The algorithm incrementally adjusts its routing of flows by iteratively transferring flow from the links with high marginal cost to links with low marginal costs. Every node does so locally in a distributed manner, when determining flow rates. The flow rates themselves are constrained by a link level schedulability requirement that ensures all packets received within a time frame can be scheduled for forwarding within one time frame. The algorithm converges to a theoretical minimal power for the network. Simulation results show that the minimal power is achievable in a relativelly small number of iterations.

I. Introduction

Power is a precious resource in wireless ad-hoc nodes because of constrained battery lifetime. Power control has been the focus of much research in the wireless community. In most wireless nodes, the communication device, such as the wireless card operates at a fixed power level. Research on power control emphasizes the ability of nodes to tune their transmission power level as a function of the incoming data rate. Typically a wireless node spends a certain fixed amount of power to support an incoming data rate. Frequently, the amount of power required to support the data rate is less than the actual amount of power spent, which leads to power wastage. The data rate may also require more power than the node's fixed power level that is used, leading to data loss.

The problem that we address is primarily a routing problem - how to route data in such a way that the total power spent in the network in routing packets from sources to destinations over wireless hops is minimized. Power spent by every node is a function of the data rate and interference experienced by the node. Nodes experience interference from nearby nodes as well as external noise present in the environment. Every node is capable of tuning its transmission power level to support the data rate. The key idea is to route more data over nodes that experience minimal interference. We present a distributed algorithm that iteratively minimizes the total amount of power spent in the network. We quantify the power spent in the network by all the nodes by a global objective function which is the sum of all link powers. The algorithm to minimize power iteratively balances the marginal derivative of the global objective function with respect to the routing fractions at each of a node's outgoing links.

Our work combines physical layer channel characteristics of network traffic flows with a distributed hill-climbing approach to power optimization and scheduling. The distinctiveness of the work arises from the fact that to the best of our knowledge it is the first attempt to reach a theoretical optimal power while considering routing and schedulability of flows. We distinguish our work from related work in section II.

We simulate our power-aware routing algorithm in a wireless ad-hoc network with 50 nodes with omni-directional antennas. The network setting that we consider is a multipath network in which a node may have multiple paths to every destination. In our evaluation, we consider two different scenarios. In the first case there is a single source and a single destination. In the second case there are 3 sources and 3 destinations, with each source sending data to all destinations.

The primary contribution of our work is that we develop a fully distributed algorithm that finds the minimum power routing configuration. Our solution follows from a distributed optimization framework for minimizing total delay in a wired network [1]. Although the objective function in [1] is delay, a similar approach can be taken for minimizing any convex global objective function including our case. We verify the quality of the solution for different network settings. In all cases the algorithm converges to a minima. In the case of a single source and single destination the algorithm converges very quickly and the minima is reached in less than 10 iterations of the algorithm. In case of 3 sources and 3 destinations, reaching the absolute minima takes many more iterations; however the total power decreases very quickly and the algorithm reaches within $4\ \%$ of the minima in 200 iterations.

Schedulability is also a concern when considering flow routing in wireless networks. This is because a node can be transmitting to, or receiving from, only one of its neighbors during a particular time slot. We thus incorporate schedulability considerations in the objective function, adapting the ideas from [3]. The minima achieved is affected by schedulability. However, when comparing the the actual minimum to an ideal case where there are no scheduling considerations, we find that the effect of scheduling on the minimum power achieved is minimal.

We would like to point that our work is an elegant synthesis of three different areas. The routing flow optimization which determines the routing fractions at every node is handled at the network layer. Schedulability is a link layer consideration; as mentioned above we handle schedulability of flows by incorporating it in the objective function. Note the relevance of the combination of schedulability with the optimization, this validates that the algorithm is implementable in a real wireless ad-hoc network. Finally, we would like to point that the objective function depicts the physical layer channel characteristics without any approximation. In power control literature, it is seen that the channel characteristics are often an approximation of the actual values. This greatly reduces the accuracy and validity of the solution, which is not the case in our work.

The rest of the paper is organized as follows. We present related work in section II. We derive the condition for schedulability of flows in III. In IV, we develop the objective function to be minimized. In V, we develop the mathematical conditions on the routing fractions that form the basis of the distributed routing algorithm. In VI we present the distributed routing algorithm itself. We simulate the algorithm for a variety of scenarios and present the results in VII. We finally conclude the paper in VIII.

II. Related Work

Our work is conceptually similar to [1], where the authors present a distributed optimization approach towards minimizing delay routing in a wired network. The theoretical basis for the minimization of a convex function that we apply is quite similar to that in [1]. However, we consider a different network setting and objective function. The objective function in our case is the amount of transmission power expended (network-wide) in a wireless ad-hoc network in order to route a given set of flows with given rates. A number of challenges arise in adapting the conditions in [1] to our case, primarily because link level schedulability must be taken in consideration in a wireless setting.

In [6] the authors consider the joint scheduling and power control problem. They do not consider routing and do not tackle the problem of finding valid routes. Also the combined algorithm is computationally complex since it must find a valid schedulable set at the begining of *each* slot. On the other hand, since we have schedulability constraints as part of the objective function itself, the scheduling algorithm is run only once, after the optimal flow rates have been obtained.

One problem considered in the power control literature is to determine the optimal transmission radius at each node in a distributed manner. One example of that is [4], in which the authors develop a minimum energy topology from the original topology by only using those links that are within the optimal transmission radius of each node. Another approach to achieve a minimum energy graph is to have link weights reflect link transmission power and then run shortest path routing between sources and destinations to find the lowest power path. Neither of these approaches take either the input flow rates or the problem of routing such input flows into account.

In [9] the authors find energy-efficient routes in a wireless ad-hoc network. They form relay regions from sources to destinations to find minimum power paths between them. They do not consider the issue of scheduling, and make simplifying assumptions about link transmission parameters, such as a constant channel fading parameter. We consider a more dynamic scenario here.

The problem of finding an optimal schedule of transmissions has been considered in [2] and [3]. Both formulate the problem as the max flow problem with optimal scheduling constraints. In [2] Jain et. al prove that the max flow problem is NP complete for a TDMA network. In our case we solve the scheduling issues in a TDMA network by absorbing the interference constraints in the link layer power equation, a different approach towards tackling scheduling constraints.

In [3] the authors look at the combined problem of achieving maximum flow and providing an optimal transmission schedule in a CDMA network, which has more relaxed interference constraints than a TDMA network in that the only constraint is that a node cannot be receiving and transmitting simultaneously. Under these conditions they present a centralized algorithm to calculate the optimal schedule and also give scheduling constraints for this optimal schedule.

III. Scheduling Considerations

We consider a time-slotted system, where time is divided into equal length frames. In each frame there are T time slots, which are numbered as $1, 2, ..., T$. Every time slot is of τ seconds. We take τ to be fixed.

Consider a node i. Let N_i be the set of neighbors of node i and M_i be the set of nodes for which i is a neighbor. Then $\aleph_i = N_i \cup M_i$ gives the set of incoming and outgoing links on i.

In [3] the authors develop a centralized optimal schedule for a time-slotted system. The constraints on the nodes are that every node can either be transmitting to or receiving from at most one other node during a particular time slot.

From [3], if the link flow vector f does not satisfy the following inequality at every node i, then the flow vector is not schedulable.

$$\sum_{l \in \aleph_i} \frac{f_l}{c_l} \leq 1 \quad \forall i \tag{1}$$

where c_l is the capacity of that particular link.

Equation(1) gives the necessary condition for schedulability at every node. Note that this condition can be locally verified by each node individually. While this condition is necessary, it is not sufficient for achieving an optimal schedule for a link flow vector f because it is derived by averaging over time slots. In [3], the authors develop a similar per-node sufficient condition for schedulability for variable τ. Following [3] we develop below a sufficient condition for fixed τ, since we are assuming that each timeslot has a network-wide fixed value. First some necessary definitions are introduced.

Definition 1: A multigraph is defined as a graph where there may be multiple edges between the same pair of nodes. An alternate representation of a multigraph is to have an integral weight w_l on each link l in the network graph such that w_l represents the number of edges in the multigraph between the same pair of nodes.

A scheduling multigraph is defined on the network graph and used to determine an optimal schedule. If τ is the length of a time slot, and c_l is the capacity of link l, then in one time slot τc_l bits can be sent on that link. Thus in order to support a flow of f_l bits/sec the link must be scheduled for $f_l/(\lceil \tau c_l \rceil)$ slots. The weight of the scheduling multigraph w_l is the number of slots needed to support the flow of f_l bits/sec and is defined as,

$$w_l = \lceil \frac{f_l}{\tau c_l} \rceil \tag{2}$$

Definition 2: If Δ is defined as the degree of the scheduling multigraph, then

$$\Delta = \max_{\forall i} \sum_{l \in \aleph_i} w_l \tag{3}$$

Definition 3: The chromatic index L of the scheduling multigraph is defined as the minimum number of colors required to color the edges of the scheduling multigraph, such that no two edges incident on the same node have the same color. From [3] the chromatic index must satisfy $L \geq \Delta$, and $L \leq 3\Delta/2$.

The notion of an optimal schedule now follows from [3]. It is clear that the length of the feasible schedule must be at least L, where the time slots are $1, 2, ..., L$, and each time slot is of length τ seconds. So in each time slot, all the edges corresponding to a particular color are activated. We introduce the following theorem from [3].

Theorem 1: For a flow vector f, and the timeslot τ which is fixed, f is schedulable *iff.* $L\tau \leq 1$.

We now introduce the sufficient condition for schedulability at every node expressed in terms of the link flow vector f if τ is fixed.

Theorem 2: The flow vector f is schedulable if,

$$\sum_{l \in \aleph_i} \lceil \frac{f_l}{c_l \tau} \rceil \leq \frac{2}{3\tau} \quad \forall i \tag{4}$$

Proof: Let us begin by assuming that

$$\sum_{l \in \aleph_i} \lceil \frac{f_l}{c_l \tau} \rceil \leq \frac{2}{3\tau}$$

and then show that the flow vector is schedulable. From (2)

$$\sum_{l \in \aleph_i} w_l \leq \frac{2}{3\tau}$$

From (3)

$$\begin{aligned} \Delta &\leq \frac{2}{3\tau} \\ L &\leq \frac{3\Delta}{2} \\ &\leq \frac{1}{\tau} \\ L\tau &\leq 1 \end{aligned}$$

So by Theorem 1 the flow vector f is schedulable if this condition is met at every node. ■

IV. Objective Function

Our objective function will be total power spent by all the nodes in the network to forward a given set of flows at a given rate through the network. Our control variables will be routing fractions (variables) at each node. The power level at each node will be chosen such that it will be possible to forward a flow at a given rate to a neighbor at a given signal to interference ratio. The total power is equal to the sum of all the link powers. We assume a time slotted communication model where in every slot a node is either sending or receiving. Examples of such a slotted MAC layer communication model are TDMA and CDMA. We derive expressions for link power for CDMA and TDMA.

A. CDMA

In CDMA, nodes use joint signature sequences, which allows different nodes to be transmitting into the channel simultaneously. The receivers receive the transmission and decode the code sequence using the code of a particular sender. In this case, neither the sender nor the receiver is affected by interference from nearby nodes. However the sender can be sending transmissions for only one receiver at a time and similarly the receiver can only be receiving transmissions from only one sender at a time. Specifically, in one time slot, a node can be either transmitting to or receiving from only one other node.

Let us next consider the relationship between link power, link flow, and the channel characteristics on the link. The link flow on link (i, k), f_{ik} is related to the channel characteristics by the following equation,

$$\begin{aligned} f_{ik} &= \log(1 + SIR_{ik}) \\ &= \log(1 + \frac{p_{ik} d_{ik}^{-\alpha}}{\frac{N_0}{2} + \sigma_{ik}^2}) \\ p_{ik} &= (2^{f_{ik}} - 1)(\frac{N_0}{2} + \sigma_{ik}^2) d_{ik}^{\alpha} \end{aligned} \tag{5}$$

where,

1) SIR_{ik} is the signal-to-interference ratio at the receiver k.
2) p_{ik} is the power expended by node i to send a flow of f_{ik} to node k.
3) d_{ik} is the transmission distance from node i to node k. We assume that there is a mechanism for calculating the transmission distance from i to k.
4) α is a constant, normally taken as 2 or 4. We take α as 2.
5) N_0 is the white noise experienced by the receiving node k. We assume that this can be estimated by k, and is propagated from k to all its neighbors including i.
6) σ_{ik}^2 is the noise variance experienced by node k on the link (i,k). We assume that this too can be estimated by the receiving node k.

The derivative of the first order of p_{ik} is given below. Both $p'_{ik}(f_{ik})$ and $p''_{ik}(f_{ik})$ are continuous in f_{ik}.

$$p'_{ik}(f_{ik}) = 2^{f_{ik}} (\ln 2)(\frac{N_0}{2} + \sigma_{ik}^2) d_{ik}^{\alpha} \tag{6}$$

B. TDMA

In TDMA, nodes again have time-slotted access to the channel as in CDMA. However since nodes do not use code sequences, interference from nearby nodes will affect communication at a node. In the most restrictive model of such interference, a link can be used only when no other link whose communication can possibly interfere with that link L is used. More realistically, however from a physical layer perspective a transmission from a nearby link need not completely interfere with communication on link L. Specifically, the transmitted power of neighboring nodes affect the signal to interference ratio (SIR) at the receiver. We generate our communication model for TDMA with this condition in mind.

We assume that the links that affect a link are only the links that belong to the neighbor set, $\aleph_k$ of the receiver k. In this case we include a term I_{jk} that models the interference from these other links, at receiver k. Thus, the link flow on link (i,k), f_{ik} is related to the channel characteristics by the following equation,

$$\begin{aligned} f_{ik} &= \log(1 + SIR_{ik}) \\ &= \log(1 + \frac{p_{ik} d_{ik}^{-\alpha}}{I_{jk} + \frac{N_0}{2} + \sigma_{ik} 2}) \\ p_{ik} &= (2^{f_{ik}} - 1)(I_{jk} + \frac{N_0}{2} + \sigma_{ik}^2) d_{ik}^{\alpha} \end{aligned} \tag{7}$$

All the terms are similar to CDMA except I_{jk}, which models the interference from neighboring nodes in the definition of SIR. $\aleph_k$ is the set of all incoming and outgoing links on k.

$$I_{jk} = \sum_{j \in \aleph_k, j \neq i} p_{jm} d_{jk}^{-\alpha} \tag{8}$$

where m is the node with which j is communicating.

We note that in our definition of TDMA, even though the SIR at the receiver is affected by communication on nearby links, the only scheduling constraint is that a node can be either transmitting to, or receiving from, only one other node.

The first order derivative $p'_{ik}(f_{ik})$ is continuous and is given by,

$$p'_{ik}(f_{ik}) = 2^{f_{ik}} (\ln 2)(I_{jk} + \frac{N_0}{2} + \sigma_{ik}^2) d_{ik}^{\alpha} \tag{9}$$

C. Objective Function

Now that we have defined link power, we define the global objective function P_T as the total power expended over the whole network:

$$\begin{aligned} P_T &= \sum_{(i,k)} p_{ik}(f_{ik}) + \sum_i \frac{1}{\frac{2}{3\tau} - \sum_{l \in \aleph_i} \lceil \frac{f_l}{c_l \tau} \rceil} \\ &= \sum_{(i,k)} p_{ik}(f_{ik}) + \\ & \sum_i \frac{1}{\frac{2}{3\tau} - \sum_{k \in N_i} \lceil \frac{f_{ik}}{c_{ik}\tau} \rceil - \sum_{j \in M_i} \lceil \frac{f_{ji}}{c_{ji}\tau} \rceil} \end{aligned} \tag{10}$$

The first term in the summation is the sum of all the link powers. The second term is a penalty function derived from Theorem 2 that restricts the space of optimization of the objective function. This term ensures that the objective function is valid only if $\sum_{l \in \aleph_i} \lceil f_l/(c_l \tau) \rceil < 2/(3\tau)$ at every node, that is flows on all the links incident on the node satisfy Theorem 2.

V. Conditions on Routing Fractions

In the following section we consider the constraints under which (10) is to be minimized. Our focus will be on the flow rate f_{il}, and their dependence on routing. We develop the mathematical conditions on routing fractions for minimizing total power expended in the network. The conditions follow to a great extent from [1] which minimizes total delay in a wired network. The objective function in [1] is delay. In our case the objective function P_T given by equation (10) is power. We develop the framework for minimization of power, with our objective function and certain variations to the conditions in [1]. The main variation is that in wired networks link flows can approach the capacity of the link, that is $f_l \leq c_l$. In our case because of schedulability considerations, the links flows must satisfy equation (4). The objective function in our case is given by equation (10) and is the sum of two convex functions, while it is a single function in [1]. So the objective function and the feasible space of optimization is different from [1].

A. Definitions

- Let $L_s = (i, k)$ denote the set of links, $\forall i, k \in N$, where N is the set of nodes in the network.
- $r_i(j)$ is the traffic first entering the network at node i and destined to node j.
- $t_i(j)$ is the total traffic at node i for node j. $t_i(j)$ also includes traffic to node j from other nodes passing through node i.
- Let the routing fraction, $\phi_{ik}(j)$, denote the fraction of $t_i(j)$ passing over the link (i, k). $\phi_{ik}(j) = 0$ if $i = j$, that is the traffic has reached the destination, or if $(i, k) \notin L_s$.

B. Flow Conservation

The total flow at node i for destination j is the sum of the input at node i for j and the flows from neighboring nodes to j passing through i.

$$t_i(j) = r_i(j) + \sum_{l \in M_i} t_l(j)\phi_{li}(j) \quad \forall i, j \tag{11}$$

Here, $\sum_{l \in M_i} t_l(j)\phi_{li}(j)$ indicates the flows from the upstream neighbors of i, passing through i to j.

The flow on a link (i, k) is defined as,

$$f_{ik} = \sum_j t_i(j)\phi_{ik}(j) \tag{12}$$

Theorem (1) in [1] states that if the routing variable set ϕ satisfies the following conditions, and the network has an input set r, then equation (11) has a unique solution for t. Each component $t_i(j)$ is nonnegative and continuously differentiable as a function of r and ϕ.

1) $\phi_{ik}(j) = 0$ if $(i, k) \notin L_s$ or $i = j$.
2) $\sum_k \phi_{ik}(j) = 1$.
3) For each i, j there is a routing path from i to j, that is there is a set of nodes such that $k, l, ..., m$ exist and $\phi_{ik}(j) > 0, \phi_{kl}(j) > 0, ..., \phi_{mj}(j) > 0$.

C. Necessary and Sufficient Conditions

The necessary and sufficient conditions for minimizing P_T follow from equations (5) to (9) in [1] which are the necessary and sufficient conditions for minimizing delay. In our case, the marginal derivatives of P_T with respect to r and ϕ are different from [1] because (10) is the sum of two convex functions. The marginal derivatives of P_T with respect to r and ϕ are:

$$\frac{\delta P_T}{\delta r_i(j)} = \sum_k \phi_{ik}(j)[p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)}] + \frac{(\frac{1}{\tau c_{ik}})}{[\frac{2}{3\tau} - \sum_{k \in N_i} \lceil \frac{f_{ik}}{c_{ik}\tau} \rceil - \sum_{j \in M_i} \lceil \frac{f_{ji}}{c_{ji}\tau} \rceil]^2} \tag{13}$$

$$\frac{\delta P_T}{\delta \phi_{ik}(j)} = t_i(j)[p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)} + \frac{(\frac{1}{\tau c_{ik}})}{[\frac{2}{3\tau} - \sum_{k \in N_i}\lceil \frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j \in M_i}\lceil \frac{f_{ji}}{c_{ji}\tau}\rceil]^2}] \quad (14)$$

Necessary Condition: The necessary condition for minimizing P_T involves finding routing fractions $\phi_{ik}(j)$ subject to the constraints on $\phi_{ik}(j)$,

1) $\phi_{ik}(j) \geq 0 \;\; \forall i, k, j$
2) $\sum_k \phi_{ik}(j) = 1 \;\; \forall i, k, j$

such that,

$$\begin{aligned} \frac{\delta P_T}{\delta \phi_{ik}(j)} &= \lambda_{ij} \quad \phi_{ik}(j) > 0 \\ &\geq \lambda_{ij} \quad \phi_{ik}(j) = 0 \end{aligned} \quad (15)$$

This indicates that at the optimal condition, the marginal derivatives of P_T with respect to the routing fractions, $\delta P_T/\delta\phi_{ik}(j)$ are equal for positive routing fractions. If the routing fraction is 0 on a link then this marginal derivative has to at least as large.

Equation (15) is not sufficient to minimize P_T. This is due to the presence of the term $t_i(j)$ in equation (14). If at a particular node $t_i(j)$ is 0 then equation (15) is automatically satisfied. As a result of this, the marginal derivatives along the outgoing links of the node are not balanced. However, even if $t_i(j) = 0$ at a particular node, the marginal derivatives at that node have to be balanced as they affect the routing at upstream nodes. So $t_i(j)$ has to be removed from the necessary condition.

Sufficient Condition: If $p_{ik}(f_{ik})$ is convex $\cup$ and continuously differentiable in $(\sum_{k \in N_i}\lceil \frac{f_{ik}}{c_{ik}\tau}\rceil + \sum_{j \in M_i}\lceil \frac{f_{ji}}{c_{ji}\tau}\rceil) < \frac{2}{3\tau})$ then the sufficient condition for minimizing P_T is given below.

$$\frac{(\frac{1}{\tau c_{ik}})}{[\frac{2}{3\tau} - \sum_{k \in N_i}\lceil \frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j \in M_i}\lceil \frac{f_{ji}}{c_{ji}\tau}\rceil]^2}] = A_{ik} \quad (16)$$

$$p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)} + A_{ik} \geq \frac{\delta P_T}{\delta r_i(j)} \quad (17)$$

$$\begin{aligned} & p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)} + A_{ik} \quad (18) \\ & - \min_{m:(i,m) \in L}[p'_{im}(f_{im}) + \frac{\delta P_T}{\delta r_m(j)} + A_{im}] \geq 0 \end{aligned}$$

for all $i \neq j, (i,k) \in L_s$ with equality for $\phi_{ik}(j)$ greater than 0.

VI. Algorithm

In this section we develop the algorithm for achieving the optimal routing fractions. The algorithm for minimizing P_T consists of iteratively balancing the flows at every node, so that the marginal derivatives along the different outgoing links from a node are equal. We now develop the various stages of the algorithm.

A. Neighbor Selection

Every node selects a set of nodes, which are one hop away from it as neighbors. We present two different heuristics that try to mitigate the interference experienced by the nodes from their neighbors. Neighbor selection is performed once only before the distributed routing fraction calculation is performed.

1) *Transmission Distance*: Consider a node k which is a neighbor of node i. Then the link power p_{ik} on the link (i,k) from i to k is defined by (5) and (7). d_{ik} in equations (5) and (7) indicate the transmission distance from i to k. We have a threshold distance d_{thresh}. The neighbor set of a node i, N_i is the set of all the nodes such that, $\forall k$, $d_{ik} \leq d_{thresh}$. The parameter d_{thresh} can be empirically determined.
2) *Signal Strength*: The second parameter that is used for neighbor selection is the signal strength of the nodes. In equations (5) and (7) the term SIR_{ik} denotes the signal to interference ratio from i to k. Similar to above we have a term SIR_{thresh} which can be empirically determined. So the neighbor set of a node i, N_i is the set of all the nodes such that, $\forall k$, $SIR_{ik} \leq SIR_{thresh}$.

B. Routing Variable Iteration

The algorithm defines a set of blocked nodes $B_i(j)$ for which $\phi_{ik}(j) = 0$ whose flows cannot be increased from 0 in order to avoid loop formation. The definition of $B_i(j)$ is the same as equation (15) in [1], with the link derivative in our case being $p'_{ik}(f_{ik})$. Because of lack of space we omit details here.

The algorithm to modify ϕ is as follows. Let ϕ^1 be the modified routing variable set.

If $k \in B_i(j)$ $\phi^1_{ik}(j) = 0$.

If $k \notin B_i(j)$,

$$\begin{aligned} a_{ik}(j) &= p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)} + A_{ik} - \\ & \quad \min_{m \notin B_i(j)}[p'_{im}(f_{im}) + \frac{\delta P_T}{\delta r_m(j)} + A_{im}] \\ \Delta_{ik}(j) &= \min[\phi_{ik}(j), \frac{\eta a_{ik}(j)}{t_i(j)}] \end{aligned} \quad (19)$$

here η is the stepsize parameter that leads to convergence. Let $k_{min}(i,j)$ be the value of m that achieves this minimization. Then

$$\begin{aligned} \phi^1_{ik}(j) &= \phi_{ik}(j) - \Delta_{ik}(j) \quad if \quad k \neq k_{min}(i,j) \\ &= \phi_{ik}(j) + \sum_{k \neq k_{min}(i,j)} \Delta_{ik}(j) \\ & \quad if \quad k = k_{min}(i,j) \end{aligned} \quad (20)$$

The algorithm above reduces the fraction of traffic on the link with the largest marginal cost and increases traffic on the other links at i. The amount of traffic increase on the link is proportional to the stepsize parameter and inversely proportional to the total traffic to that destination. The convergence of the algorithm depends on the stepsize parameter η. The algorithm keeps iterating till it achieves minimum P_T.

The routing variable iteration in equation (19) requires $\delta P_T/\delta r_k(j)$ at i from each of its downstream neighbors. Every node i calculates its $\delta P_T/\delta r_i(j)$ according to (13) and propagates it to each of its upstream neighbors, that is $\forall j, j \in M_i$. It is clearly seen that this procedure works if only the routes are loop free.

Feasibility of ϕ^1: Let f^1_{ik} be the flow on link (i,k) generated by ϕ^1_{ik}. If $p_{ik}(f^1_{ik}) \notin (\sum_{k \in N_i} \lceil \frac{f^1_{ik}}{c_{ik}\tau} \rceil + \sum_{j \in M_i} \lceil \frac{f^1_{ji}}{c_{ji}\tau} \rceil) < \frac{2}{3\tau})$, then $A^1_{ik} \rightarrow \infty$. Then, $\phi^1_{ik}(j) = \phi_{ik}(j) \quad \forall j$. This guarantees that if ϕ generates a set of flows f that is feasible, then f^1 generated by ϕ^1 is also feasible.

C. *Initial Route Selection*

The algorithm must begin with an initial route that is loop free. We use the DSR[8] to select the initial route between every source and every destination. The routing fraction ϕ at each node is set to 1. At the next iteration, the routing fractions to other members of the neighbor set are changed according to (20).

D. *Estimating Interference*

In case of TDMA the signal to interference ratio includes I_{jk} which is the interference experienced by the node from neighboring nodes. The routing algorithm above does not actually schedule the flows but only incorporates the schedulability constraint so that the final flow vector is schedulable. In the absence of the actual schedule the exact interference from the neighboring nodes during a particular time slot is not known. We are assuming that I_{jk} is fixed over L slots.

VII. Simulations

In this section we present simulation results that demonstrate the convergence of the algorithm in different network settings.

A. *Network Setting*

We simulate a network of 50 wireless ad-hoc nodes. We assume that the nodes are stationary and are located on a $500 * 500$ square grid. We choose the neighbors according to the first heuristic, that is, transmission distance, with d_{thresh} set to 95. Thus, every node chooses all nodes that are within a distance of 95 from itself as neighbors. We simulate the presence of noise in the environment. $N_0/2$ at each of the nodes is set to .01. We model the presence of σ^2_{ik} in the environment by an exponential distribution with a mean of 1. We simulate the number of iterations it takes for the algorithm to converge in different network settings. The convergence of the algorithm depends on the stepsize parameter η. The theoretical value of η that guarantees convergence is very small. Let $M = max_{p_{(ik)}} p''_{ik}(f_{ik})$, that is M is equal to the second derivative of link power of the link having maximum link power. The η that guarantees convergence is equal to $1/[MN^6]$ [1], where N is the total number of nodes in the network. We find that in most cases the algorithm converges for a much larger value of η. We typically choose the maximum value of η that leads to convergence, but also show the dependence of convergence on η.

B. Single Source

In this case we simulate the algorithm with a single source and a single destination. That is, exactly one of the 50 nodes is a source, and one of the other 49 is the destination. The algorithm needs to be initialized with a loop-free path. The loop-free path between the source and the destination is chosen using DSR[8]. In the first iteration nodes choose only one neighbor to forward to, and the corresponding routing fraction ϕ at each node is set to 1. During subsequent iterations, nodes split the flows to all their neighbors. This may lead to loop formation. The algorithm has loop detection techniques in which all the nodes that can form a potential loop are put in a blocked set. In our simulations, we choose only routes that are loop free.

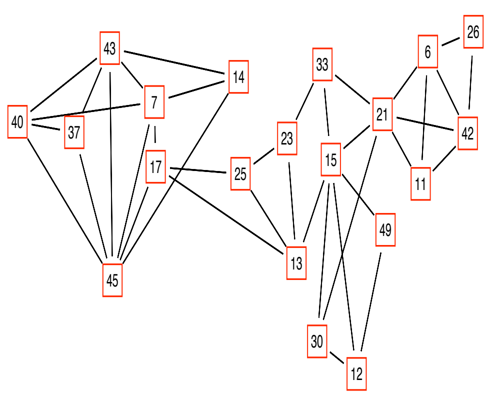

Fig. 1. Topology of single source single destination, source = 40, destination = 26.

The topology of the single source, single destination case is shown in Figure 1. Here node 40 is the source and node 26 is the destination. We simulate the algorithm for two different network types, CDMA and TDMA.

CDMA: In case of CDMA, link power is given by equation (5). Capacity on each link is constant. The initial set of flows are chosen such that they satisfy the capacity-related scheduling constraint, equation (4).

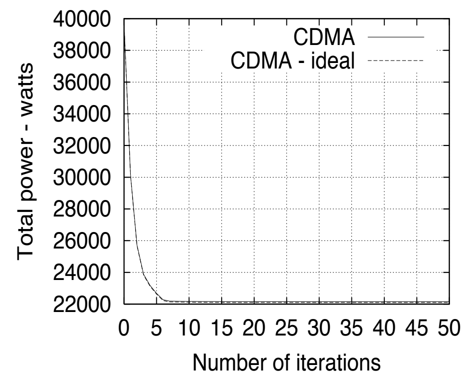

Fig. 2. Total power for a single source single destination CDMA network. $\eta = 10^{-5}$

Figure 2 shows the convergence of the algorithm for a CDMA network with 1 source and 1 destination. The value of η in this case is 10^{-5}, much larger than the theoretical value of η that guarantees convergence. We also plot an ideal case, called CDMA-ideal in which there are no capacity constraints on the flows. The y-axis plots the total amount of power in watts expended in the network at every iteration. From the figure it is seen that both CDMA and CDMA-ideal fare almost equally. The most important observation here is that the total power in the network reduces very fast in the number of iterations and the algorithm reaches within 0.1% of the optimal power in just 9 iterations. This is of significance since even if it may not be possible to run the algorithm for hundreds of iterations to achieve the theoretical minimum power, for all practical purposes it is sufficient to run the algorithm for just a few iterations.

TDMA: Figure 3 demonstrates the same result for TDMA. The network setting and the value of η are the same as CDMA. In TDMA interference from neighboring nodes must be estimated according to the procedure of section VI.D. Both TDMA and TDMA-ideal (which ignores link capacities) behave almost equally. As in the case of CDMA, con-

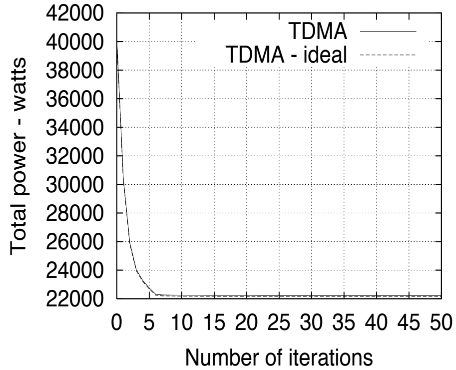

Fig. 3. Total power for a single source single destination TDMA network. $\eta = 10^{-5}$

vergence is very fast, again the algorithm reaches within 0.1% of the optimal power in just 9 iterations.

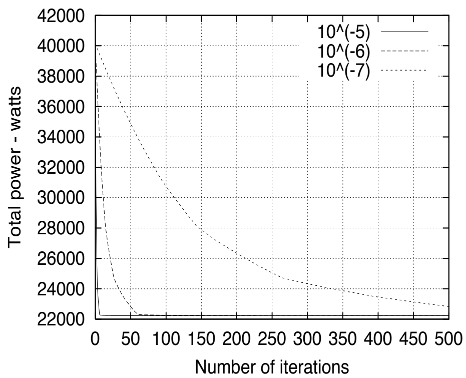

Fig. 4. Comparison of different ηs for a single source single destination TDMA network.

Dependence on η: Figure 4 shows the convergence of the above TDMA network for different values of the stepsize parameter η. The convergence of the algorithm depends to a great extent on the value of η. A large value of η may lead to oscillations, while a very small value leads to slower convergence. From the figure we see that $\eta = 10^{-5}$ leads to very fast convergence with the algorithm reaching within 0.1% of the optimal power in 9 iterations only. While $\eta = 10^{-7}$ also converges, the rate of convergence is much slower. The reason is that a larger value of η allows more flow to be transferred between a node's outgoing links at each iteration.

We verify the convergence of the single source single destination case for a number of <source, destination> pairs. Convergence for these <source, destination> pairs chosen from the network topology in Figure 6 is given in the table in Figure 5..

<s,d>	within % of optimal power in 9 iterations	
	input rate = 1	input rate = 2
40 - 8	1.6	2.3
40 - 24	0.1	0.8
5 - 8	3.5	3.5
5 - 24	0.7	1.1
5 - 26	1.2	3.9
27 - 8	0.9	0.9
27 - 24	0.1	0.3
27 - 26	0.1	1.3

Fig. 5. Rate of convergence of different single source, single destination CDMA networks. within $\%$ of optimal power in 9 iterations for different input rates. $\eta = 10^{-5}$

From Figure 5 we see that in each case the minimum power for all practical purposes is achieved in less than ten iterations for a single source, single destination network.

C. Multiple Sources

In this subsection we present simulation results of convergence of the algorithm for 3 sources and 3 destinations, with each source sending data to each of the destinations. $N_0/2$ at each of the nodes is set to .01 and every node experiences σ_{ik}^2 according to to an exponential distribution with a mean of 1. As in the single source case, we only choose routes that are loop free. We simulate both CDMA and TDMA. The topology is shown in Figure 6.

CDMA: Figure 7 demonstrates the convergence of the algorithm for both CDMA and CDMA-ideal. For CDMA the capacity on each link is set to a constant. Both cases converge in a similar manner. Unlike the single-source, single-destination case, with multiple sources the algorithm takes much longer to converge. In our simulations it takes more

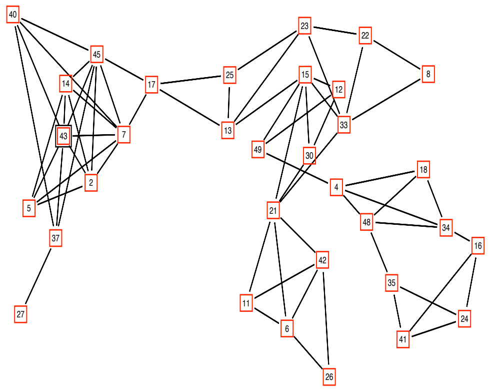

Fig. 6. Topology of multiple sources and destinations, sources = 40, 5 and 27 destinations = 8, 24 and 26.

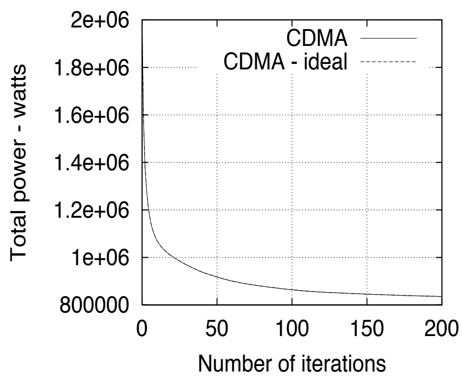

Fig. 7. Total power for the 3 sources 3 destinations CDMA network. $\eta = 10^{-7}$

than 5000 iterations of the algorithm to converge to an absolute minima. However the algorithm reaches within 19% of the optimal power in 25 iterations, within 13% in 50 and within 4% in 200 iterations. Even though convergence is not as fast as for the single-source, single-destination case it is possible to achieve a total power that is quite close to minimum power in a relatively small number of iterations.

Once again we verify the result for other sources and destinations. The table in Figure 8 shows the convergence of the algorithm for 5, 6 and 7 sources and destinations chosen from the same topology as in Figure 6. The convergence pattern is similar to what we describe above with total power close to the minima achieved in a relatively small number of iterations.

no. of s,d	within % of optimal power in *iterations*		
	25	*50*	*200*
5	11.5	5.9	1.5
6	11.5	5.9	1.5
7	16.3	9.3	2.6

Fig. 8. Rate of convergence of different multiple sources, multiple destinations CDMA networks. Within % of optimal power in 25, 50 and 200 iterations. $\eta = 10^{-7}$

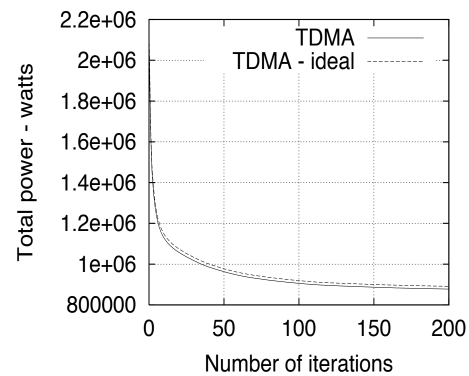

Fig. 9. Total power for a 3 sources 3 destinations TDMA network. $\eta = 10^{-7}$

TDMA: Figure 9 plots convergence of the algorithm in a TDMA network for 3 sources and 3 destinations. The network setting parameters are same as CDMA. The only difference is that in TDMA interference from neighboring nodes is taken into account. Convergence is also same as CDMA, with absolute minima taking more than 5000 iterations but the algorithm reaching within 4% of the optimal power in 200 iterations.

Figure 10 compares the convergence of CDMA and TDMA in this network setting. From Figure 10 we see that the convergence pattern in both settings is the same, but total power for TDMA is slightly higher. This is because interference from neighboring nodes is taken into account in case of TDMA. This inherently increases the amount of noise present in the network, and so it consumes more power to route the same amount of data.

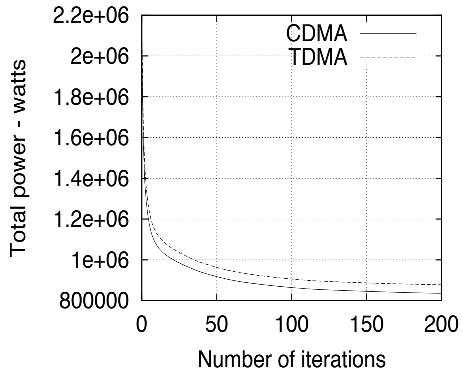

Fig. 10. Comparison of TDMA and CDMA for a 3 sources 3 destinations network. $\eta = 10^{-7}$

At 200 iterations, TDMA converges to 6.6% more power than CDMA.

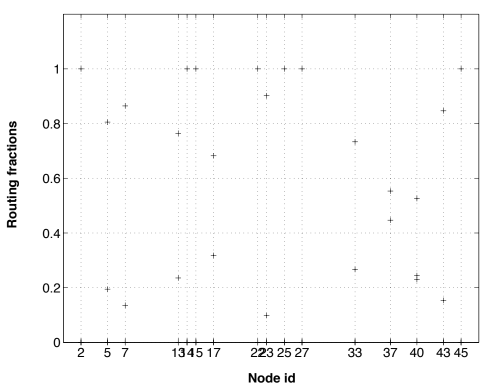

Fig. 11. Routing fractions for destination 8 in the 3 sources 3 destinations CDMA network. $\eta = 10^{-7}$

Routing fractions: Figures 11 and 12 show the routing fractions for TDMA and CDMA for $destination$ 8 in the topology of Figure 6. Note that the routing fractions at a particular node always sum up to 1.

Dependence on η: Figure 13 shows the convergence of the 3 sources, 3 destinations TDMA network for different values of the stepsize parameter η. It is seen that $\eta = 10^{-6}$ converges fastest and $\eta = 10^{-7}$ converges at a similar rate to it. The rate of convergence for $\eta = 10^{-8}$ is much slower than the other two.

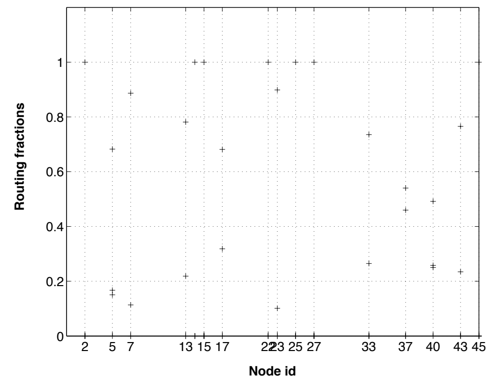

Fig. 12. Routing fractions for destination 8 in the 3 sources 3 destinations TDMA network. $\eta = 10^{-7}$

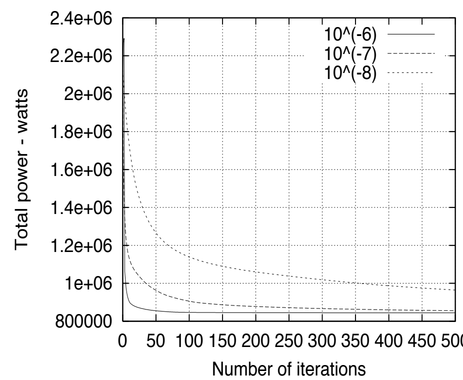

Fig. 13. Comparison of different ηs for a 3 sources 3 destinations TDMA network.

VIII. Conclusion and Future Work

In this paper we have addressed the joint routing and scheduling problem in a wireless ad-hoc network. From the scheduling aspect, we derive schedulability conditions on the flows and incorporate them in the formulation of the optimal routing problem. We develop the optimization conditions on routing fractions and develop a distributed routing algorithm for achieving theoritical minimum power in the network. Our objective function also incorporates interference. Interference is of two types, white noise or ambient noise present in the vicinity of the nodes and external noise that may be introduced into the environment. We simulate the algorithm for CDMA and TDMA networks. Our simulation results show the convergence properties of the algorithm and show that nearly optimal power can be reached in a relatively small number of iterations.

References

[1] Robert G Gallager, "A Minimum Delay Routing Algorithm Using Distributed Computation ", in *IEEE Transactions on Communications*, 1977.

[2] Kamal Jain, Jitendra Padhye, Venkat Padmanabhan, Lili Qiu, "Impact of Interference on Multi-hop Wireless Network Performance", in Proceedings of *Mobicom* 2003.

[3] Murali Kodialam, Thyaga Nandagopal, "Characterizing Achievable Rates in Multi-hop Wireless Networks: The Joint Routing and Scheduling Problem", in proc. *Mobicom* 2003.

[4] Yurong Chen, Emin Gun Sirer, Stephen B. Wicker, "On Selection of Optimal Transmission Power for Ad hoc Networks".

[5] Jae-Hwan Chang, Leandros Tassilius "Energy Conserving Routing in Wireless Ad-hoc Networks".

[6] Tamer ElBatt, Anthony Epremides "Joint Scheduling and Power Control for Wireless Ad-hoc Networks".

[7] Shannon C.E. "A Theorem on Coloring the Lines of a Network".

[8] David B. Jhonson, David A. Maltz "Dynamic Source Routing in Ad Hoc Wireless Networks".

[9] Yuan Xue, Baocun Li "A Location-aided Power-aware Routing Protocol in Mobile Ad Hoc Networks".

IX. Appendix

A. Marginal Derivative of P_T

We derive equations (13) and (14) here.

$$P_T = \sum_{(i,k)} p_{ik}(f_{ik}) + \sum_i \frac{1}{\frac{2}{3\tau} - \sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil}$$

Let

$$A = \sum_{(i,k)} p_{ik}(f_{ik})$$

$$B = \sum_i \frac{1}{\frac{2}{3\tau} - \sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil}$$

$$\frac{\delta A}{\delta r_i(j)} = \sum_k \phi_{ik}(j)[p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)}]$$

Let us look at $\delta B/\delta r_i(j)$.

$$\frac{\delta B}{\delta r_i(j)} = (\frac{1}{[\frac{2}{3\tau} - \sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil]^2}) (\frac{\delta}{\delta r_i(j)}\sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil + \frac{\delta}{\delta r_i(j)}\sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil)$$

$$\begin{aligned}\frac{\delta}{\delta r_i(j)}\sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil &= \frac{\delta}{\delta r_i(j)}\sum_{k\in N_i}\lceil\frac{\sum_j t_i(j)\phi_{ik}(j)}{c_{ik}\tau}\rceil \\ &= \sum_{k\in N_i}\lceil\frac{\phi_{ik}(j)}{c_{ik}\tau}\rceil \\ &= \sum_{k\in N_i}(\frac{\phi_{ik}(j)}{c_{ik}\tau}) \\ \frac{\delta}{\delta r_i(j)}\sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil &= \frac{\delta}{\delta r_i(j)}\sum_{j\in M_i}\lceil\frac{\sum_l t_j(l)\phi_{ji}(l)}{c_{ji}\tau}\rceil \\ &= 0\end{aligned}$$

$$\frac{\delta B}{\delta r_i(j)} = \frac{\sum_{k\in N_i}(\frac{\phi_{ik}(j)}{c_{ik}\tau})}{[\frac{2}{3\tau} - \sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil]^2}$$

$$\frac{\delta P_T}{\delta r_i(j)} = \frac{\delta A}{\delta r_i(j)} + \frac{\delta B}{\delta r_i(j)}$$

This results in equation (13). We now derive equation (14).

$$\frac{\delta A}{\delta \phi_{ik}(j)} = t_i(j)[p'_{ik}(f_{ik}) + \frac{\delta P_T}{\delta r_k(j)}]$$

$$\frac{\delta B}{\delta \phi_{ik}(j)} = (\frac{1}{[\frac{2}{3\tau} - \sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil - \sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil]^2}) (\frac{\delta}{\delta \phi_{ik}(j)}\sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil + \frac{\delta}{\delta \phi_{ik}(j)}\sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil)$$

$$\begin{aligned}
\frac{\delta}{\delta\phi_{ik}(j)}\sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil &= \frac{\delta}{\delta\phi_{ik}(j)}\sum_{k\in N_i}\lceil\frac{\sum_j t_i(j)\phi_{ik}(j)}{c_{ik}\tau}\rceil \\
&= \frac{t_i(j)}{c_{ik}\tau} \\
\frac{\delta}{\delta\phi_{ik}(j)}\sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil &= \frac{\delta}{\delta\phi_{ik}(j)}\sum_{j\in M_i}\lceil\frac{\sum_l t_j(l)\phi_{ji}(l)}{c_{ji}\tau}\rceil \\
&= 0
\end{aligned}$$

$$\frac{\delta B}{\delta\phi_{ik}(j)} = \frac{(\frac{t_i(j)}{c_{ik}\tau})}{[\frac{2}{3\tau}-\sum_{k\in N_i}\lceil\frac{f_{ik}}{c_{ik}\tau}\rceil-\sum_{j\in M_i}\lceil\frac{f_{ji}}{c_{ji}\tau}\rceil]^2}$$

$$\frac{\delta P_T}{\delta\phi_{ik}(j)} = \frac{\delta A}{\delta\phi_{ik}(j)}+\frac{\delta B}{\delta\phi_{ik}(j)}$$

This results in equation (14).

B. Necessary Condition

We now derive the necessary condition for minimizing P_T, that is equation (15) subject to the constraints.

1) $\phi_{ik}(j)\geq 0 \quad \forall i,k,j$
2) $\sum_k \phi_{ik}(j)=1 \quad \forall i,k,j$

We use Lagrange multipliers to do the minimization. The lagrangian is,

$$L(\phi_{ik}(j),\lambda,\mu) = -P_T+\sum_k\lambda_k(1-\phi_{ik}(j))+\sum_k\mu_k\phi_{ik}(j)$$

The minimization conditions are,

$$-\frac{\delta P_T}{\delta\phi_{ik}(j)}-\lambda_k\frac{\delta\phi_{ik}(j)}{\delta\phi_{ik}(j)}+\mu_k\frac{\delta\phi_{ik}(j)}{\delta\phi_{ik}(j)} = 0 \tag{21}$$

$$\begin{aligned}
\frac{\delta P_T}{\delta\phi_{ik}(j)} &= -\lambda_k+\mu_k \\
\text{Let } -\lambda_k &= \lambda_{ij} \\
\frac{\delta P_T}{\delta\phi_{ik}(j)} &= \lambda_{ij}+\mu_k \\
\mu_k\phi_{ik}(j) &\geq 0 \\
\mu_k &\geq 0 \quad \phi_{ik}(j)=0 \\
\mu_k &= 0 \quad \phi_{ik}(j)>0 \\
\frac{\delta P_T}{\delta\phi_{ik}(j)} &= \lambda_{ij} \quad \phi_{ik}(j)>0 \\
&\geq \lambda_{ij} \quad \phi_{ik}(j)=0
\end{aligned} \tag{22}$$

This is equation (15).