# Channelization for Dynamic Multi-Frequency, Multi-Hop Wireless Cellular Networks

JaeSheung Shin*, Raju Kumar*, Parthu Kishen+, Thomas F. La Porta*

* Department of Computer Science and Engineering,
\+ Department of Electrical Engineering,
The Pennsylvania State University,
University Park, PA 16802

## *Abstract*

Multi-hop relaying in cellular networks can greatly increase capacity and performance by exploiting the best available links to a base station. We envision an environment in which relay networks are dynamically formed when performance on the radio access network is degraded and then dissolved when the performance improves or the radio spectrum on which the relay network is operating is reclaimed. Each relay network operates on a different frequency band. Likewise, a relay network may channelize its frequency band to offer non-interfering links among the mobile nodes within a single relay network. We propose a set of algorithms used to form such relay networks on-demand. Each algorithm provides a simple and distributed frequency assignment scheme. We also propose two enhancements to improve network throughput of resulting relay networks. We evaluate these algorithms in terms of the overhead of the relay network formation. The evaluation results show that having nodes outmost from the BS initiate route discovery first is the best approach for reducing the formation overhead. The results also show that there is a large increase in throughput when using multiple frequencies in a relay network. Further, the performance of the network using multiple frequencies based on our simple frequency assignment is very close to that of a network using optimal frequency assignment.

*Keywords*: Frequency Assignment; Multi-hop Wireless Cellular Networks; Network Formation; Relay Network

## 1. Introduction

Mobile nodes in traditional wireless cellular networks communicate through centralized base stations (*BS*) in a pre-defined spectrum. To improve the performance of such cellular networks, several studies on multi-hop wireless cellular networks have been undertaken [1][6][13]. We investigate relaying in a third generation (*3G*) wireless system, CDMA2000 1xEVDO [4][8]. In the 1xEVDO system, the bit rate achieved during each time interval depends on its signal quality to the mobile node. If the BS can schedule nodes with better signal quality more often, then a higher average bit rate for the network can be achieved.

In this paper, we envision a multi-hop wireless cellular network that uses 1xEVDO scheduling [16]. Multiple relay networks are *dynamically formed* when performance on the radio access network is degraded. A disjoint frequency band for each relay network is allocated dynamically by the BS. In this way, multiple non-interfering relay networks may operate in parallel.

The dynamic nature of these relay networks motivates the need for an explicit procedure for mobile nodes to form a relay network. Moreover, it requires that every mobile node be able to communicate over a wide range of frequency bands. We assume that each mobile node is equipped with an agile radio [5] in addition to a cellular interface to meet this requirement. The cellular interface is leveraged so that the BS may broadcast information during relay network formation.

We also allow the band used by a single relay network to be divided into multiple orthogonal frequencies to construct a relay network comprised of non-interfering links. This allows multiple nodes within

---

* This work was supported in part by NSF grant CNS-0508114 and DARPA grant BAA 05-42.

* Corresponding author. Tel.: +1-814-865-9505; Fax.: +1-814-865-3176

*E-mail addresses*: jsshin@cse.psu.edu (JaeSheung Shin), rajukuma@cse.psu.edu (Raju Kumar), parthu@cse.psu.edu (Parthu Kishen), tlp@cse.psu.edu (Thomas F. La Porta)

communication range of each other to transmit simultaneously without relying on a MAC protocol or distributed scheduling algorithm to resolve contention and prevent collisions.

In this paper, we present three relay network formation algorithms. Each algorithm first determines gateway (*GW*) nodes which are best suited for acting as a bridge between the relay network and the BS. The algorithms then discover a path from each node through the relay network to the GW node. Each algorithm also provides a simple and distributed frequency assignment scheme to build relay networks with non-interfering links to improve network throughput.

We also propose two enhancements to increase network throughput of resulting relay networks. In the first enhancement, each node detects interference from remote nodes based on its received Signal-to-Noise-plus-Interference Ratio (*SNIR*), and may dynamically switch channels to reduce interference. Second, we support GW reselection so that a new GW with a higher achievable data rate may be used.

We compare these algorithms in terms of the overhead of the relay network formation. We also measure the throughput of the resulting relay network in three scenarios: (1) relay network using a single frequency; (2) relay network using multiple frequencies assigned by our distributed frequency assignment; and (3) relay network using multiple frequencies with optimal assignment. We also quantify the benefits of the enhancements.

The results lead us to conclude that having nodes outmost from the BS initiate route discovery first is the best approach for reducing the relay network formation overhead. The results also show that using simple and distributed frequency assignment we can achieve high throughput gains over using networks that uses only a single frequency. This simple frequency assignment algorithm achieves 80-85% of the optimal average throughput. Moreover, they show that the enhancements further improve average throughput.

The rest of the paper is organized as follows. In section 2, we briefly discuss related work in this area. In section 3, we present the network model for dynamic multi-frequency, multi-hop wireless cellular networks. Basic operations used for relay network formation and frequency assignment are explained in section 4. In section 5, we describe two enhancements to increase network throughput of resulting relay networks. In section 6, we present our simulation environment and results. In section 7, we discuss the related issues of the paper. We conclude the paper in section 8.

## 2. Related Work

Work on multi-hop wireless networks can be broadly grouped into two areas: those that consider a single frequency on which the network operates, and those which propose to use multiple frequencies.

There has been a great deal of work on single frequency multi-hop wireless networks to improve cellular network performance [1][6][13]. Several papers address relaying in GSM networks [1][6]. We instead investigate a 3G wireless environment in which the sharing of communication resources is done via a combination of regulating power and time division multiplexing.

For example, in the 1xEV-DO system, the BS schedules only a single node for downlink transmission at any instant and transmits at full power. The bit rate achieved during each time interval depends on its signal quality to the mobile node, which can be simplified to a function of distance. If the BS can schedule nodes with better signal quality more often, then a higher average bit rate for the network can be achieved.

To maximize throughput of 3G networks, the UCAN [13] system proposes that the BS transmits all downlink data to mobile nodes with high signal quality (corresponding to the GWs in this paper). These nodes then forward data to other nodes in the network through a high speed relay network operating in a different spectrum than the 3G interface, specifically using an 802.11 network. In this way, the downlink from the BS can always run at its maximum rate and all users achieve higher throughput. It is reported that UCAN can achieve improvements of the average and maximum throughput of up to 37% and 82%, respectively. Our work is most similar to the UCAN system. However, it is different from ours in that the relay network operates on a *single* frequency and a *persistent* 802.11 network exists for use as the relay network in UCAN system.

Recently, there has been a great deal of effort on advanced wireless networks in which nodes can simultaneously communicate with their neighbors using multiple radio interfaces over multiple orthogonal channels [2][11][12].

In [12] it is shown that the network throughput can be significantly improved when mobile nodes are equipped with multiple interfaces and enabled to utilize multiple channels. In order to form multiple orthogonal links efficiently, several channel assignment schemes are proposed in [2][11]. These efforts differ from ours in two key ways. First, they consider a *centralized* channel assignment scheme. Second, they are designed for *ad hoc or mesh networks*, and do not, therefore, rely on

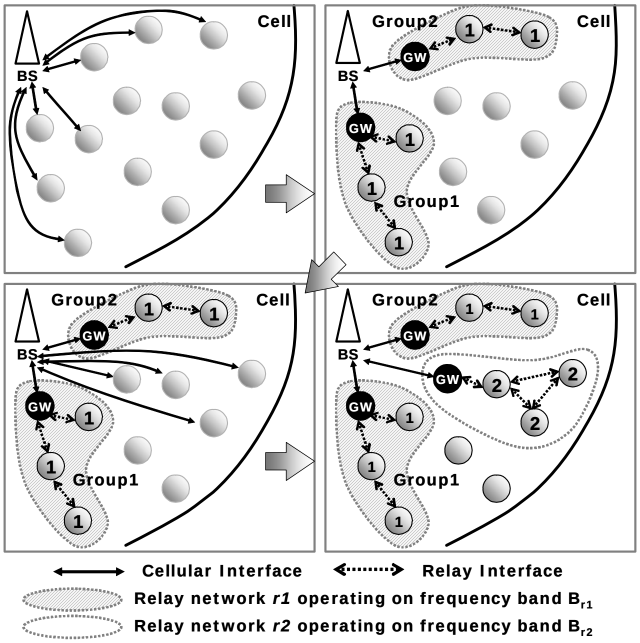


**Fig. 1. Example of relay network formation**

ordering of requests to improve the efficiency of frequency assignments and BS's broadcasts to disseminate information.

## 3. Network Model

In this paper, we focus on a single cell environment in which there is a BS and several mobile nodes. In figure 1, seven mobile nodes communicate with the BS initially. The BS advertises a new frequency band, $B_{r1}$, available on which a relay network is formed. Then, the nodes form a relay network $r_1$ operating on $B_{r1}$. At some time later, the BS advertises frequency band $B_{r2}$ on which a new relay network $r_2$ is formed.

### 3.1. Frequency Band Allocation

A frequency band is allocated by the BS whenever a relay network is formed. Let $S$ be the set of all available radio frequency bands, $S = \{B_1, B_2, \ldots, B_N\}$, and $R$ be the set of relay networks currently operating, $R = \{r_1, r_2, \ldots, r_R\}$. If $B_{ri}$ and $B_{rj}$ are frequency bands on which relay network $r_i$ and $r_j \in R$ operate, respectively, they should satisfy the requirements: (1) $B_{ri} \in S$ and $B_{rj} \in S$, (2) $B_{ri} \neq B_{rj}$

### 3.2. Frequency Assignment

Each frequency band may be divided into multiple orthogonal frequencies. For example, a band $B_{ri}$ consists of the set of orthogonal frequencies, $F_{ri} = \{f_1, \ldots, f_K\}$, where $K$ is the maximum number of orthogonal frequencies in $B_{ri}$. These frequencies may be used to construct a relay network comprised of non-interfering links so that multiple nodes within range of each other may transmit simultaneously without relying on a MAC protocol or distributed scheduling algorithm to resolve contention and prevent collisions.

### 3.3. Dynamic Formation and Dissolution of the Relay Network

The formation and the dissolution of each relay network are performed through the following steps:

(1) If the BS schedules a large number of nodes that have poor signal quality, it may advertise a frequency band on which a relay network may be formed. The frequency advertisement is broadcast over the cellular control channel. This information includes all available orthogonal frequencies in the band and a control channel used for the relay network formation.
(2) Some mobile nodes form a relay network operating on the introduced band.
(3) Once the relay network is created, the BS schedules only the GW nodes for transmission. Nodes in the relay network communicate with the BS via the GWs.
(4) At a later time, the relay network may be dissolved and nodes return to using the cellular interface with the BS.

In this paper, we present the relay network formation algorithms used in step (2) and compare them in terms of the overhead of the relay network formation. We also measure the network performance of the resulting relay network in step (3).

## 4. Network Formation and Frequency Assignment Algorithms

Each relay network is formed in two phases. In Phase I, GW nodes are chosen for each group. A group consists of a set of mobile nodes which are reachable to each other. The transmission radius of a node in the relay network (e.g. 250m in 802.11) is very small compared to the cellular coverage (e.g. 10~20 km). Thus a relay network generally consists of several isolated groups of mobile nodes. In figure 1, the relay network $r_1$ consists of two isolated groups of mobile nodes. Each group needs at least one GW node to act as a bridge between the BS and the group.

Phase II of relay network formation consists of two steps. In the first step, the nodes join the relay network by establishing a path to one of the GWs. In the second step, while returning a route reply (*RREP*) message to the source node of a route request (*RREQ*), the GW and intermediate nodes on the reverse path assign orthogonal frequencies to the links on the path.

Since the relay networks are dynamically formed and dissolved in our network, we use a modified version of AODV [14] as the ad-hoc routing protocol to find the path from the mobile nodes to the GW. In this modified AODV, the *RREQ* contains path information like DSR [9]. Each intermediate node appends its identification to the *RREQ* before forwarding it. Thus, upon receiving the *RREQ*, the GW node can learn the members of specific groups within the relay network.

We leverage two optimization features of reactive ad-hoc routing protocols. First, a node may *passively learn a route* to a destination if it is part of a longer path to the destination. In this case, it will not launch its own *RREQ*. Second, a node that has previously learned a route may *immediately return* this route in response to a request without a further search.

These two features can greatly reduce the number of messages flooded to find routes to the GWs. In order to make the utmost use of the passive route learning, intuitively the furthest node from the BS is the best choice to launch a route request first. This will greatly reduce the load at the GW node. To fully leverage the immediate response to a *RREQ*, scheduling the nodes nearest the BS to launch a *RREQ* first is the best choice.

Motivated by these observations, we propose the three relay network formation algorithms which use node's location information - two centralized algorithms: Furthest First (*FF*), Nearest First (*NF*); and a distributed algorithm: Locally Outmost First (*LOF*). The algorithms dictate the scheduling by which nodes send out their own *RREQ*.

### 4.1. Phase I – GW Discovery

In our environment, the BS broadcasts the frequency band information on which a relay network is formed over the cellular control channels so that all nodes within the cell receive it simultaneously. A pre-agreed upon channel within the relay network frequency band is defined as the signaling channel used to establish and maintain the relay network.

To select GW nodes, every node forming a new relay network periodically broadcasts a neighbor advertisement (*NADV*) message over the control channel as soon as it receives new frequency band information. The *NADV* contains the identification of the sending node, and a metric indicative of the received signal quality from the BS, and the distance of the node from the BS.

Whenever a node receives a *NADV* message from its neighbors, it compares its signal strength to the BS with their own. If the node has better signal quality to the BS compared to all one-hop neighbors, it acts as a GW node as shown in figure 2. If several one-hop neighbors have the same signal quality, one may be selected according to additional metrics such as processing power, battery lifetime, etc.

```
GW_Discovery()

$D_i$ = Distance from the BS of node $i$;
Receive $NADV$ from all neighbors;
$D_k$ = Distance from the BS of neighbor k;
If ( $D_i$ == min ($D_i$, $D_k$) ) for all neighbors k,
    then node $i$ acts as a GW;
```

**Fig. 2. GW discovery algorithm**

### 4.2. Phase II – Joining the Relay Network

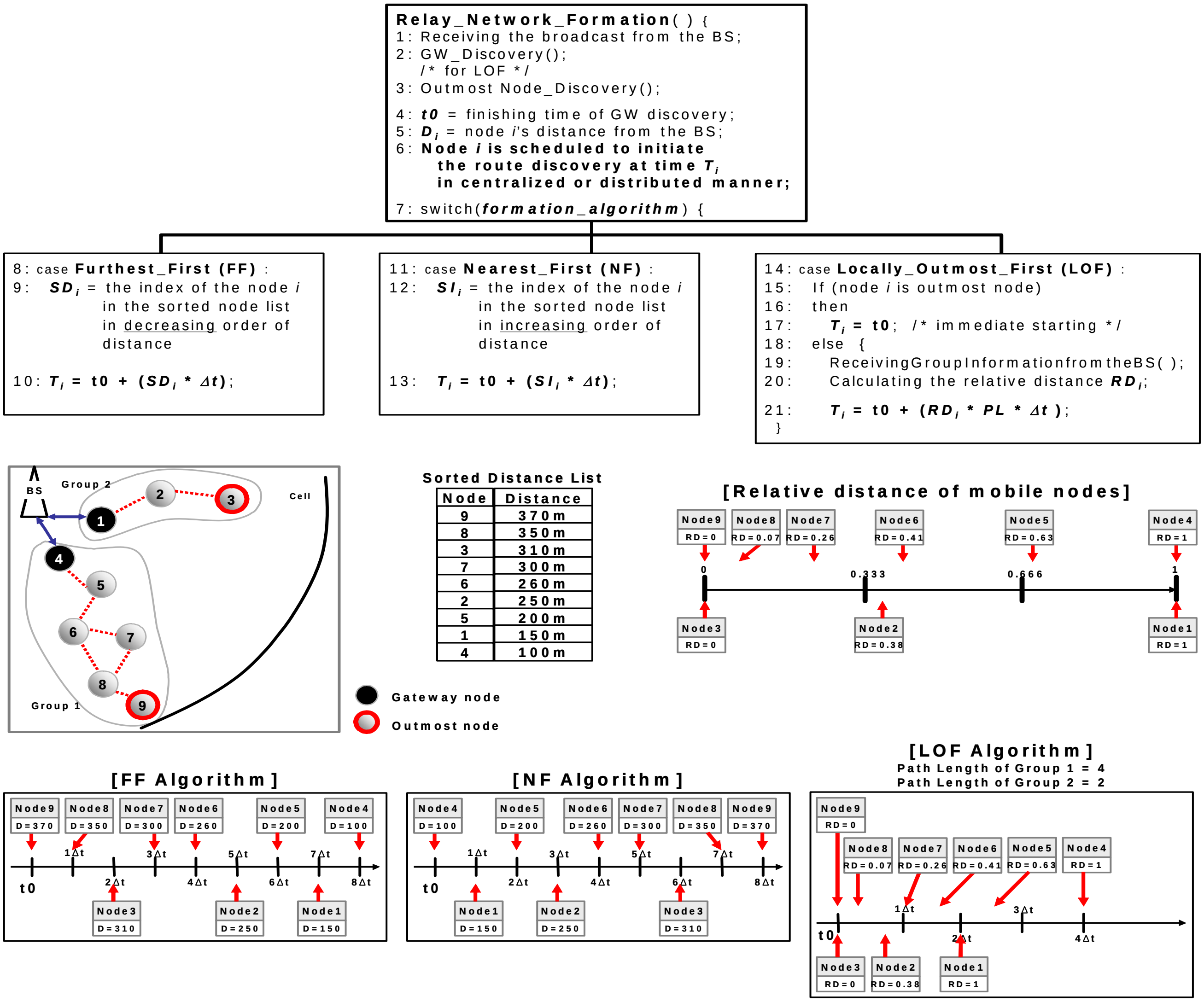
In this section we discuss the procedure by which nodes join a relay network. This consists of two steps: initiating route discovery and assigning frequencies on each link in the relay network.

#### 4.2.1. Step 1: Initiating a Route Discovery

The difference among our various network formation algorithms is the schedule by which nodes initiate route discovery.

***Baseline(BL)*** – As a baseline, we consider the case in which every node joining the relay network initiates a route discovery just after GW discovery phase without any specific scheduling for initiating route discovery. This is very simple, but runs the risk of severe congestion on the relay network during the formation phase because every node sends out *RREQ* almost simultaneously. It may also overload the GW node with many *RREQ*s during a short time period.

***Furthest First(FF)*** – To solve the problems of BL, we exploit the ability of nodes to passively learn routes. This optimization reduces both the congestion caused by the potential *RREQ* storm in the relay network and the load on the GW. From this observation, the best



```
Relay_Network_Formation( ) {
  Receiving the broadcast from the BS;
  GW_Discovery();
  /* for LOF */
  Outmost Node_Discovery();
  t0 = finishing time of GW discovery;
  $D_i$ = node i's distance from the BS;
  Node i is scheduled to initiate
    the route discovery at time $T_i$
    in centralized or distributed manner;
  switch(formation_algorithm) {

  case Furthest_First (FF) :
    $SD_i$ = the index of the node i
          in the sorted node list
          in decreasing order of
          distance
    $T_i = t0 + (SD_i * \Delta t)$;

  case Nearest_First (NF) :
    $SI_i$ = the index of the node i
          in the sorted node list
          in increasing order of
          distance
    $T_i = t0 + (SI_i * \Delta t)$;

  case Locally_Outmost_First (LOF) :
    If (node i is outmost node)
    then
      $T_i = t0$; /* immediate starting */
    else {
      ReceivingGroupInformationfromtheBS( );
      Calculating the relative distance $RD_i$;
      $T_i = t0 + (RD_i * PL * \Delta t)$;
  }
```

Sorted Distance List

| Node | Distance |
|---|---|
| 9 | 370m |
| 8 | 350m |
| 3 | 310m |
| 7 | 300m |
| 6 | 260m |
| 2 | 250m |
| 5 | 200m |
| 1 | 150m |
| 4 | 100m |

**Fig. 3. Formation algorithms and example of scheduled initiation of route discovery**

schedule is to have the nodes furthest from the BS launch *RREQ*s first, so that as many nodes will passively learn a route to the BS as possible.

Note that the formation algorithm schedules the *initiation* of each node's route discovery. It is impossible to anticipate which node will be located at the end of the longest path from the GW nodes until all nodes complete the route discovery. However, the nodes furthest from the BS generally have the greatest possibility to have the longest path from the GW nodes. This algorithm thus forces the furthest node to launch *RREQ* first.

There are two drawbacks with this algorithm. First, it is impractical for a node to learn the exact location of all other nodes. We therefore assume that the BS may act as a central controller and track each node's location in the cell. The BS forces the node furthest from it to launch a *RREQ* first by sending a pre-defined signal to the node over the cellular interface.

Second, if a node is not on the path between the furthest node and the BS, it will not passively learn a route. However, it is impossible to anticipate which nodes may or may not passively learn a route to the BS in advance. To ensure that all nodes learn a route, the BS schedules each node to launch its own *RREQ* in decreasing order of distance from the BS at every certain time interval, Δt. Thus, if a node passively learns a route before its scheduled time, the time slot assigned to the node is wasted. This introduces unnecessary latency into the network formation process.

As shown in figure 3, node 9 (furthest from the BS) is scheduled to send out its *RREQ* at time *t0;* node 8 (the next furthest node) is scheduled at $t0+\Delta t$, and so on. After node 9 launches its *RREQ* at time *t0*, nodes 5, 6, and 8 may passively learn a route to the BS. Thus, if they receive the *RREP* before *t0+6Δt*, *t0+4Δt*, and *t0+Δt*, respectively, they will not send out their own *RREQ*s and the time slots assigned to the nodes are wasted.

***Nearest First(NF)*** **–** When a node already having a path to the GW receives a *RREQ*, it can immediately return a *RREP* to the source node. In so doing, the majority of the *RREQ* traffic is replied to without being forwarded to the GWs. Therefore, having the node nearest to the BS initiate a *RREQ* first will also lower load on the network. As with FF, this scheme requires the BS to act as a central controller.

For example, in figure 3, if node 6 has a route to the BS before receiving the *RREQ* generated by node 9 at time *t0+8Δ*t, it returns a *RREP* to node 9 immediately.

In addition to requiring a centralized controller, this algorithm has the additional drawback that almost all nodes must launch a *RREQ*, at least to reach the node preceding them to the GW.

***Locally Outmost First (LOF)*** **–** FF and NF reduce the number of signaling messages flooded in the network. However, mobile nodes may experience long latency due to the strict, sequential scheduling. They are, moreover, not practical because the centralized scheduler must know the distance to all nodes. We present a *distributed* algorithm, called *Locally Outmost First*, to achieve the benefits of FF and NF, and the potential low latency of the baseline case. Without any centralized controller, this algorithm allows each node to make a schedule for its own route discovery based on its distance and the group information broadcast by the BS.

This algorithm is composed of two steps - outmost node discovery and scheduling. A relay network consists of several isolated groups of mobile nodes as shown in figure 1; each group can form the relay network independently. In this algorithm, all outmost nodes in *each group* initiate a route discovery first just after outmost node discovery. Thus, several paths will be discovered simultaneously. The *outmost node* has the greatest distance from the BS compared to all neighbors within its transmission range. Each group has at least one outmost node. For example, nodes 9 and 3 act as the outmost node in groups 1 and 2, respectively, in figure 3.

### (1) Outmost Node Discovery

This procedure can be overlapped with GW discovery. During the GW discovery, each node compares its distance from the BS with all neighbors' distances. As shown in figure 4, if a node has a greater distance compared to all neighbors, it becomes an outmost node.

```
Outmost_Node_Discovery()
$D_i$ = Distance from the BS of node $i$;
$D_k$ = Distance from the BS of neighbor $k$;
if ( $D_i$ == max ($D_i$, $D_k$) ) for all neighbors $k$,
    then node $i$ acts as a outmost node;
```

**Fig. 4. Outmost node discovery algorithm**

### (2) Scheduling

The GW node initially delivers the group information to the BS when it receives *RREQs* sent from the outmost nodes which passed through a new path. The BS then broadcasts the group information so that mobile nodes in the cell exploit it. The group information contains the ID of the GW, the distance of the GW ($D_G$), the ID of the outmost node that sent the *RREQ*, the distance of the outmost node ($D_O$), and the path length (*PL*).

With the characteristic of flooding, all nodes in the group receive at least one *RREQ* sent from the outmost nodes. If a node receives a *RREQ* from an outmost node and the BS broadcasts group information including the outmost node and a GW, the node belongs to the same group as the outmost node and the GW. Moreover, the node may be on one of the paths between the outmost node and the GW. Thus, based on the broadcast group information and the information in received *RREQ*, each node can calculate its *relative distance* between the GW node and the outmost node. It then makes a schedule for its own route discovery with the relative distance as shown in figure 5. The relative distance has a value in the range [0..1]. A smaller value indicates that the node is closer to the outmost node.

```
Scheduling()
$D_i$ = Distance from the BS of node $i$;
$D_G$ = Distance from the BS of GW;
$D_O$ = Distance from the BS of the outmost node;
$PL$ = Path length = hop count between the outmost node
       and the GW;
Relative distance of node $i$
    $RD_i = 1 - \left( \frac{D_i - D_G}{D_O - D_G} \right)$        eq. (1)
$T_i$ = Scheduled time of node $i$'s route discovery
    = $t0 + (RD_i \times PL \times \Delta t)$
```

**Fig. 5. Scheduling of route discovery algorithm**

As shown in figure 3, since nodes 3 and 9 are outmost nodes, they send out *RREQs* simultaneously at time *t0*. Then, the intermediate nodes can calculate their relative distances based on the group information. For example, node 6 belongs to group 1 in which the GW node and the outmost node are nodes 4 and 9, respectively. Node 4's distance from the BS is 100m and node 9's distance is 370m. Thus, the relative distance of node 6 is 0.4. The path length is 4. Thus, node 6 makes a schedule for its route discovery at time $t0+1.6\Delta t$. If it does not passively learn a route before $t0+1.6\Delta t$, it sends out its own *RREQ* at the scheduled time. Since this scheduled time is inversely proportional to its relative distance, this algorithm makes use of the passive route learning.

### 4.2.2. Step 2: Assigning Frequency

Each node establishes a single path to a GW node. The GW node returns a *RREP* to the source node on behalf of the BS. While returning the *RREP*, the GW and intermediate nodes on the reverse path are responsible for assigning a non-interfering frequency to links on the path. In this section, we present a simple distributed frequency assignment algorithm.

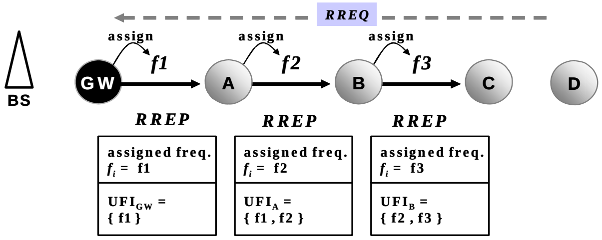
We define the used frequency information (*UFI*) of a node as the set of frequencies used on its all incident links. To make a local frequency assignment when a path is being established, a node requires the *UFI* of all neighbors. This information is received in two ways: first, the *NADV* messages periodically broadcast include the *UFI* of a node. Second, the *UFI* is included in the *RREP* generated by a node. Figure 6 shows the algorithm and an example of the frequency assignment on a path.

Each node $i$ in the relay network $r_i$ maintains a set of available frequencies in the relay network, $Af_i$. Initially $Af_i = F_{ri}$, all frequencies in the relay network. When node $i$ receives $UFI_j$ from the *NADV* generated by node $j$, it recalculates $Af_i = Af_i \cap \overline{UFI}_j$. When node $i$ receives a *RREP* from node *i-1*, it further recalculates $Af_i = Af_i \cap \overline{UFI}_{(i-1)}$. It then assigns a frequency to its next-hop-link by choosing from the resultant $Af_i$ as shown in figure 6.

If the node receiving the *RREP* has no available non-interfering frequencies, it may select a frequency that is already chosen by the previous nodes. In this case, the MAC protocol will resolve contention between the competing links, thus lowering network performance. In order to alleviate the degradation of network performance, the node picks up a frequency from $UFI_{(i-1)}$ except $f_{i-1}$.

```
Frequency_assignment( )

$Af_i$        = Set of available frequencies of node $i$ ;
$UFI_i$       = Set of frequencies used by node $i$ ;
$UFI_j$       = Set of frequencies used by neighboring node $j$ ;
$UFI_{(i-1)}$ = Set of frequencies used by the previous node on the reverse path ;
$f_{i-1}$     = Frequency for upstream link of node $i$ assigned by the previous node ;

 /* Initial Condition */
$Af_i$  = $F_{ri}$;

/* Upon receiving NADV from a neighboring node j */
/* recalculate Af_i                                */
$Af_i = Af_i \cap \overline{UFI_j}$ ;  return;

/* Upon receiving RREP from the previous node on the reverse path  */
/* recalculate Af_i and UFI_i                                      */
$Af_i = Af_i \cap \overline{UFI_{(i-1)}}$ ;
$UFI_i = UFI_i \cup \{ f_{i-1} \}$;

/* Assign a frequency */
if ( $Af_i$ != Ø )
   /* There is an available frequency */
   then { choose a frequency $f_i$ from $Af_i$ and assign it to its next-hop-link ;
              $Af_i = Af_i \cap \overline{\{ f_i \}}$;
              $UFI_i = UFI_i \cup \{ f_i \}$;
          }

   /* There is no available frequency */
   else { pick up a frequency $f_i$ from $UFI_{(i-1)}$ except $f_{i-1}$ and
              assign it to its next-hop-link ;
              $UFI_i = UFI_i \cup \{ f_i \}$;
          }

insert $UFI_i$ and $f_i$ into RREP message and return to the next node ;
```

**(a) Frequency assignment algorithm**



**(b) Example of frequency assignment**

**Fig. 6. Frequency assignment**

### 4.3. Transient Behavior

In order to transfer downlink data to the destination node, BS needs to keep a list of nodes on the relay network. Thus, the GW node delivers the group membership information to the BS when it receives a *RREQ* that passed through a new path. Since some intermediate nodes which have a route to the GW may immediately return a *RREP* to the source node, some *RREQ*s do not reach the GW. Thus, group membership information maintained by the BS may be incomplete even if the source node already joined the relay network. In this case, the BS will continue to use its cellular interface to communicate with the source node in the downlink until it receives data from the node via the relay network on the uplink. At this time, the BS will store a record of the node being in the relay network group served by a GW and transfer the next downlink data through the relay network.

Due to mobility, links in the relay networks may break and new nodes may join a relay network. In these cases, the effected nodes issue *RREQs* as if they were joining a relay network for the first time. Any node receiving the *RREQ* responds with an immediate *RREP* as during relay network formation. Since the nodes already maintain the used frequency information, frequency assignment on the new link is made as during network formation.

### 4.4. Multiple GWs in the Same Isolated Group

In order to reduce the relay network formation delay, our GW discovery algorithm utilizes the information carried by *NADV* messages that traverse only one hop. Therefore, depending on the topology (e.g. concave hull), there may be multiple GWs in the same isolated group.

In this case, the group is divided into multiple subgroups logically. Each subgroup has only one GW and consists of nodes which establish a path to this GW. A node in this isolated group may receive multiple *RREP*s from the GWs; such a node selects a GW that has better signal quality to the BS than itself and is the fewest hops away. The frequencies assigned to links on the paths unselected have a soft state. Thus, they are released unless a data packet is transmitted before the timer expires.

## 5. Enhancements

In addition to the basic algorithms described in section 4, we propose two enhancements to improve the throughput of resulting network: local optimization and GW reselection.

### 5.1. Local Optimization

When multiple frequencies are available for a relay network, all nodes make frequency assignments according to the algorithm described in section 4.

In this algorithm, each node assigns a frequency to the next-hop-link based on the $Af_i$ calculated. This may not guarantee non-interfering frequency assignments for the following reasons. First, when choosing a frequency, each node cannot consider the subsequent frequency assignment of other nodes on a different branch of the relay network. As shown in figure 7, while establishing a path, the *UFI* maintained in nodes D and B will not include frequency *f3*. They thus may each assign *f3* to the link to node E and C, respectively, which will result in contention on these links. In this case, the MAC protocol will arbitrate the transmission of the nodes and they will achieve lower throughput.

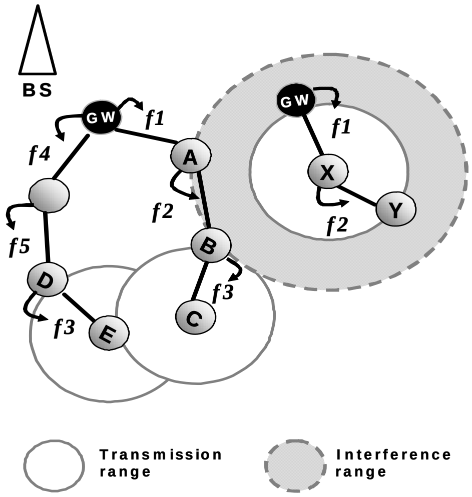


**Fig. 7. Potential interference**

Second, the proposed frequency assignment algorithm ignores the fact that a node's interference range may be greater than its transmission range. As shown in figure 7, node A and node X are not within transmission range and belong to different groups. They thus cannot exchange their *UFI*. Moreover, the RTS-CTS exchange will not occur between these nodes. As a result, these two nodes may assign the same frequency, $f_2$, to the next-hop-link. This leads the nodes to interfere with each other resulting in higher bit error rates and lower throughput at node B.

#### 5.1.1. Local Optimization Algorithm

In general, interference from remote nodes causes a drop in received SNIR at the interfered node so that it experiences a higher bit error rate. This local optimization algorithm consists of two phases in which each node measures received SNIR of all its incident links periodically and reassigns a new frequency to the interfered link if SNIR of the link has fallen off markedly.

#### (1) Phase 1: Interference Detection

Based on the statistical information of the radio interface module, the MAC layer periodically calculates a moving average of the received SNIR. If the MAC layer detects a decrease of the received SNIR below a threshold, it notifies the routing (AODV) layer.

### (2) Phase 2: Frequency Reassignment

Based on the information received from MAC layer, the routing layer recognizes the interfered link and determines the addresses of the node which sends data on this link. It then selects a new frequency from its stored $Af_i$, inserts it into a reassign frequency request (*RAFRQ*) message, and sends it to the node as shown in figure 8. After sending or receiving the acknowledgement of the *RAFRQ* message, the routing layer reassigns the new frequency to the interfered link.

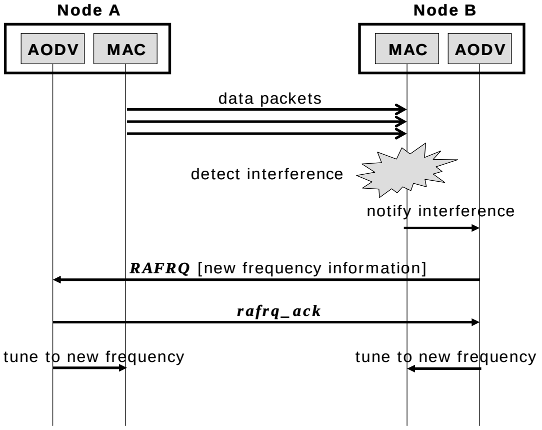
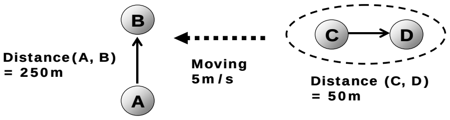


**Fig. 8. Signaling flows for local optimization**

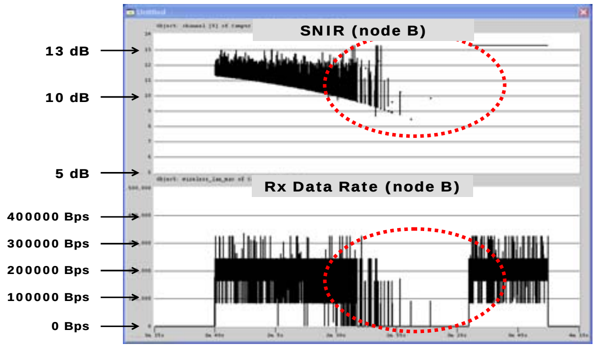
## 5.1.2. Effect of Local Optimization

We measure the effect of local optimization with a simple scenario. In this scenario, there are two pairs of nodes, as shown in figure 9-(a). Nodes A and C are transmitting a fixed amount of data, 6 Mbits, to nodes B and D, respectively, on the same frequency. The distance between nodes A and B is 250m and the distance between nodes C and D is 50m. Thus, node D has a higher received SNIR than node B. The pair of nodes C and D are moving toward the pair of nodes A and B at a speed of 5m/s.

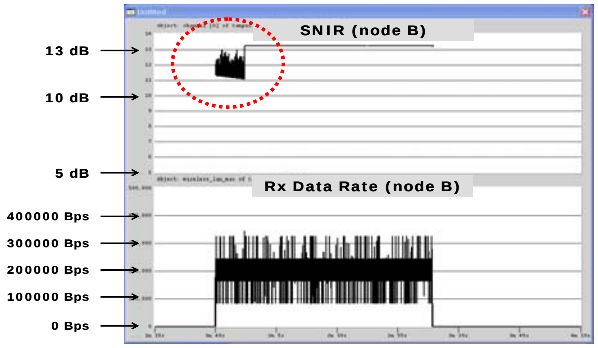
Without local optimization, the received SNIR of node B degrades rapidly as the two pairs of nodes get closer. This causes a higher bit error rate at node B and the received data rate of node B drops down to 0 bps as shown in figure 9-(b). After node C completes the transmission of data, or moves away, node B will receive rest of the data from node A. With local optimization, node B is able to detect the interference from nodes C and D and switch to a new frequency. As a result, two pairs of nodes can transmit data in parallel without any interference so that they can achieve higher throughput as shown in figure 9-(c).



(a) Scenario



(b) Without local optimization



(c) With local optimization

**Fig. 9. Effect of local optimization**

The frequency switching of a pair of nodes can trigger a cascade of frequency switching in the network. Our local optimization algorithm is inspired by the work in [10] which claims that if every node selects its channel within a finite number of frequency switches, frequency assignment reaches a stable state where nodes cease changing frequencies.

## 5.2. GW Reselection

The GW acts as a bridge between the BS and an isolated group of nodes in the relay network. Thus, the data rate of the GW node has a great impact on the average throughput of this group of nodes.

In the 1xEVDO system, based on the received SNIR, mobile nodes transmit a 4-bit data rate control (*DRC*) sequence to the BS to request a specific data rate [4][8]. A GW with a higher DRC can achieve a higher data rate. In this paper, we consider a scenario in which the average throughput of the group of nodes improves because of GW reselection as shown in figure 10.

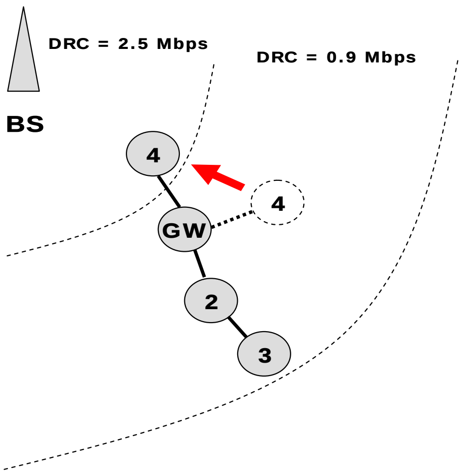


**Fig. 10. Example of GW reselection scenario**

### 5.2.1. GW Reselection and Re-registration

If a non-GW node in the group moves toward the BS it may be able to achieve a higher data rate with the BS than the current GW. In this case, this node should be selected as a new GW replacing the current GW. For example, figure 10 shows the scenario in which node 4 is selected as a new GW with higher data rate. As shown in figure 11, GW reselection and re-registration are performed through the following procedure:

When a node is selected as a GW, it sends a register GW (*REGGW*) message to the BS. This message contains ID of the node and its expected DRC. If the expected DRC is affordable, the BS acknowledges *REGGW* message and then transmits data to the GW with the data rate corresponding to the DRC.

We assume that the *HELLO* message generated by a GW contains the status information of the GW, for example, node ID, distance from the BS, and current DRC. Based on periodically received *HELLO* messages, the neighboring nodes of a GW know the current status of the GW. If a node gets closer to the BS so that it may achieve a higher DRC than current GW, it sends a notify new GW (*NOTNEWGW*) message to current GW. Upon receiving a *NOTNEWGW* message, the current GW returns the acknowledgement of *NOTNEWGW* message containing all its routing information to the new GW. The new GW also sends a *REGGW* message to the BS which contains its expected DRC. Then, the BS acknowledges *REGGW* message, updates its routing information, and then transmits data to the new GW with the corresponding data rate.

At this time, the former GW establishes a link to the new GW over which it forwards all the data it receives from the relay network.

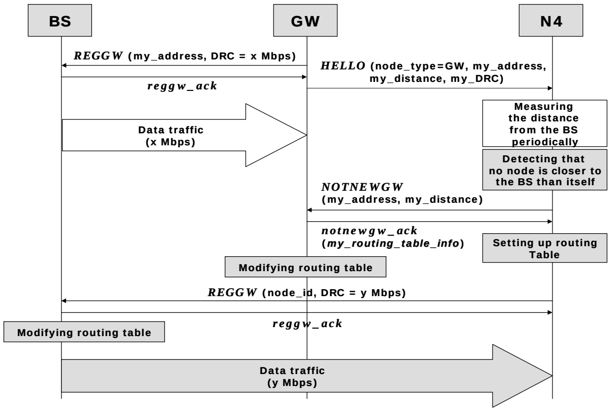
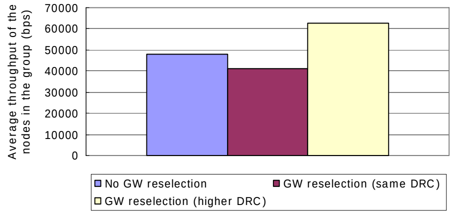
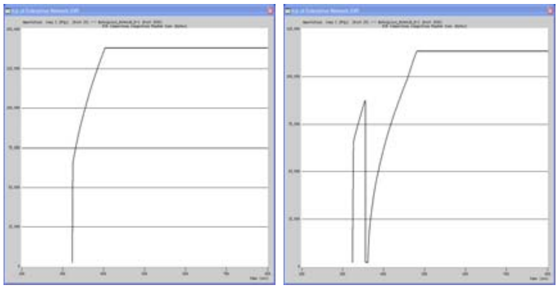


**Fig. 11. Signaling flows for GW reselection**

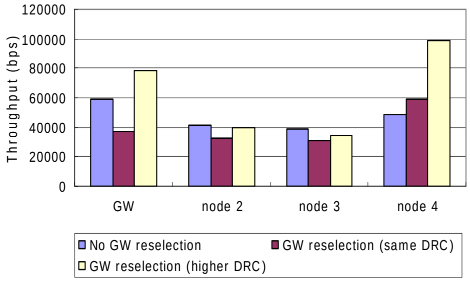


**(a) Comparison of average throughput of the nodes**



No GW reselection  GW reselection

**(b) Congestion window size of former GW**



**(c) Throughput of the nodes**

**Fig. 12. Effect of GW reselection**

### 5.2.2. Effect of GW Reselection

Based on the scenario in figure 10, we measure the average throughput of the nodes. Figure 12-(a) shows that we can achieve higher average throughput of the nodes if a node which has a higher DRC than the current GW is selected as a new GW.

However, GW reselection requires that at least the

BS, former GW, and new GW should update their routing information while they are transmitting or relaying data. Some data packets may be dropped during this period. Moreover, the BS stops data transmission to former GW and resumes it to the new GW. As a result, the TCP congestion window size of all nodes in the group may be reduced. In fact, our trace, shown in figure 12-(b), shows that the congestion window of the nodes drops to 0.

As shown in figure 12-(c), however, we still achieve improved average throughput if the new GW has a higher data rate to the BS than the former GW. We also show that if the new GW does not achieve higher throughput than the former GW, overall throughput will decrease due to the drop in the congestion window.

## 6. Performance Evaluation

We simulated the relay network formation algorithms in NS-2 v.2.1b9a to compare them in terms of the overhead during network formation. OPNET v.11.5 was used to measure the throughput gains achieved by the single and multi-frequency relay networks.

### 6.1. Simulation Environment

The air interface to all nodes when no relay network is in operation is based on a 1xEVDO. This interface is also used between the GW nodes and the BS when a relay network is in operation. We use a simplified approximation of a 1xEVDO system in which the BS schedules the mobile nodes on the forward cellular link with three classes of data rates: 2.5Mbps, 921Kbps, and 153Kbps.

The relay network uses IEEE 802.11a which provides up to 12 orthogonal frequencies. But 4 frequencies out of 12 are dedicated to point-to-point transmission. Thus, we assume that the relay network uses only 8 frequencies in these simulations.

The experiments are based on a 4000x4000m$^2$ 6-sector cell with up to 100 mobile nodes. In this scenario, each node in the relay network downloads 6Mbits data from an FTP server. Table 1 summarizes the simulation parameters.

The value of $\Delta t$ used in the formation algorithms discussed in section 4 is initially broadcast by the BS. In a 1xEvDO system, each mobile node can occupy maximum 16 timeslots (each of which is 1.67ms) during its turn. Thus, in order to synchronize with this system, $\Delta t$ is set to 26.72ms in this simulation.

| Parameter | Value |
|---|---|
| Cell size (BS at center) | 4000m X 4000m |
| N (# of nodes / cell) | 1 ➔ 100 |
| # of sectors / cell | 6 sectors |
| $DRC_i$ (Classes of HDR links) | $DRC_3$ : 2457 Kbps<br>$DRC_2$ : 921 Kbps<br>$DRC_1$ : 153 Kbps |
| Downloaded file size | 6 Mbits / user |
| Application | FTP/TCP |
| Advertised frequency band | Frequency band used for 802.11a |
| # of orthogonal frequencies | 8 |
| Air interface range | 250 m |

**Table 1. Simulation parameters**

### 6.2. Comparison of the Formation Algorithms

We use three metrics to compare the performance of our relay network formation algorithms: *signaling traffic*, *formation latency*, and *GW load* as shown in table 2.

These metrics indicate the *overhead* of the relay network formation. Signaling traffic generated indicates the degree of network congestion during network formation. Since relay networks are typically formed when the network is experiencing poor performance, formation latency is critical. The processing load at GWs is proportional to the traffic intensity of the cellular interface between the BS and GWs during the formation process.

For the comparison, 100 different network topologies are generated in each case of 10 to 100 nodes. Each data point is the average over the runs.

The results of the simulation are shown in figures 13-16. It is clear that all algorithms incur trade-offs. In general, the algorithms with strict scheduling like NF and FF have the highest latency (figure 13) due to their sequential nature, while those with more parallelism have lower latency at the expense of higher signaling traffic (figure 14) and GW load (figure 15).

NF has the highest latency, but the fewest messages to hit the GW nodes. Even though BL has low latency, the signaling traffic and the load at GW node during network formation is about three times of that when using FF and NF. LOF has good performance in terms of signaling traffic and load at the GW node, but high latency at high node density.

Therefore, we define the *weighted overall overhead* of each algorithm as shown in Table 2. It is the summation of relative value of the three metrics which are given different weights according to their importance.

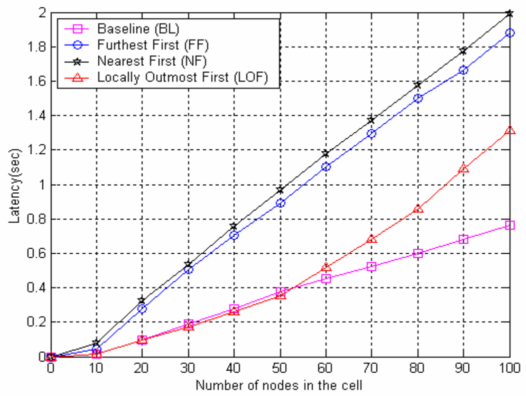


Fig. 13. Formation latency

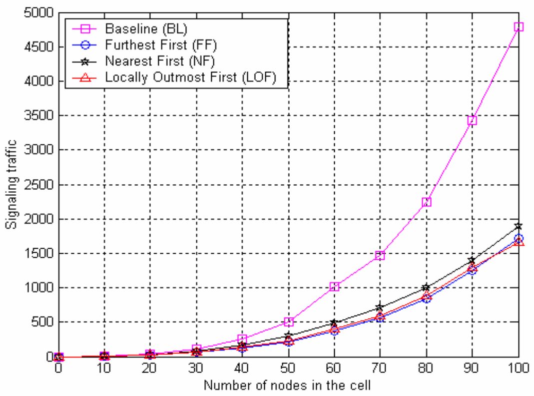


Fig. 14. Signaling traffic

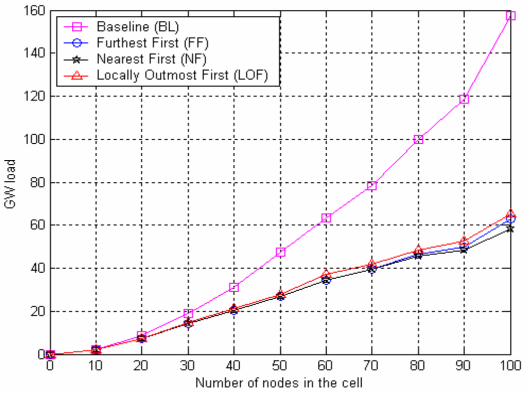


Fig. 15. Load at GWs

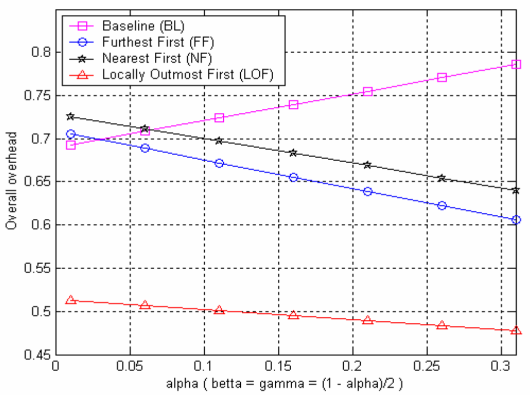


Fig. 16. Overall overhead ( $\beta = \gamma = (1-\alpha)/2$ )

In figure 16, β and γ are set to be equal, meaning that low latency and low load on the GW are of equal importance. We evaluate the overhead over the range when the signaling traffic is considered highly unimportant ($\alpha = 0.01$), through the case in which all three metrics are of equal importance ($\alpha = \beta = \gamma = 0.33$). As shown, LOF is the best performing algorithm in terms of overall formation overhead.

## 6.3. Throughput of the Relay Network

Since LOF is the best overall performing algorithm, we measure the throughput of the relay networks resulting from LOF using OPNET. We consider both single and multi-frequency relay networks formed by LOF. We also consider the multi-frequency relay networks with optimal frequency assignments as discussed below.

We use two metrics to compare the throughput of the relay networks: *overall network throughput* and *average node throughput* as shown in table 2.

The overall network throughput indicates the total throughput of the relay network as measured at the BS until all nodes complete their FTP transaction. Thus, it generally depends on the throughput of the node with the lowest data rate. The average node throughput is calculated based on the average FTP response time of the nodes in the relay network.

### 1) Single Frequency Relay Networks

In this section, we examine the performance gain of the single frequency relay network over the pure 1xEVDO system. Our simplified system supports data rates on the forward link of 2.5Mbps, 921Kbps, and 153Kbps.

Considering a time period in which the DRC of all nodes is constant, if all nodes in the 1xEVDO network are backlogged, each node is scheduled for the same amount of time at the rate supported by their DRC. The network throughput under these conditions is simply the weighted average of the throughput of all nodes in the 1xEVDO network.

Thus when there are $m$ active nodes in the sector, the maximum achievable data rate of node $i$, $R\_hdr_i$, and the maximum average data rate of the nodes, $\overline{R\_hdr}$, are given by

| **Performance of the formation algorithms**<br>**($S$ = type of formation algorithm (e.g. BL, FF, NF, LOF))** | | |
|---|---|---|
| **Name** | Description | Formula |
| **Signaling Traffic** | Total number of routing messages received by all mobile nodes forming the relay network | **M(S) =** $\sum_{i=1}^{N} msg_i$ **, where** $msg_i$ **= # of received messages by node i** |
| **Formation Latency** | The time elapsed between the first *RREQ* and all nodes having a route to the BS | **L(S) =** ***t_final – t_init*** **, where** ***t_final*** **= time when all nodes have a route to the BS**<br>***t_init*** **= time when the first** ***RREQ*** **is sent out** |
| **GW Load** | Total number of routing messages received by all GW nodes | **G(S) =** $\sum_{g=1}^{G} g\_msg_g$ **, where** $g\_msg_g$ **= # of received messages by GW node g** |
| **Weighted Overall Overload** | Summation of relative overhead compared to the other algorithms each of which has different weight according to the importance of the metric | **RM(S)=relative signaling traffic of algorithm S=** $\frac{M(S)}{\max(M(BL), M(FF), M(NF), M(LOF))}$<br>**RL(S)=relative formation latency of algorithm S =** $\frac{L(S)}{\max(L(BL), L(FF), L(NF), L(LOF))}$<br>**RG(S)=relative GW load of algorithm S =** $\frac{G(S)}{\max(G(BL), G(FF), G(NF), G(LOF))}$<br>**WO(S)=weighted overall overhead of alg. S =** $(\alpha \times RM(s)) + (\beta \times RL(s)) + (\gamma \times RG(s))$<br>$\alpha + \beta + \gamma = 1$ |

| **Throughput of the resulting relay network** | | |
|---|---|---|
| **Name** | Description | Formula |
| **Overall Network Throughput** | The total throughput of the relay network as measured at the BS until all nodes complete their FTP transaction | $Rt_i$ **= FTP response time of node** $i$<br>$FS$ **= Download file size of each node**<br>**Longest Response time = max (** $Rt_i$ **),** $1 <= i <= N$<br>**Overall network throughput** $= \frac{FS \times N}{\max(Rt_i)}$ |
| **Average Node Throughput** | Average throughput of each nodes in the relay network | **Average node throughput =** $\frac{FS}{\left(\frac{\sum_{i=1}^{N} Rt_i}{N}\right)}$ |

**Table 2. Performance metrics**

$$R\_hdr_i = DRC_i \times \frac{1}{m}$$ , where $DRC_i$ = data rate of node $i$ eq. (2)

$$\overline{R\_hdr} = \frac{\sum_{i=1}^{m} R\_hdr(i)}{m} \qquad \text{eq. (3)}$$

When using a relay network, the BS schedules the GWs instead of the members of the relay network. In this way, all nodes on the relay network share the data rate sustained on the link between the GW and the BS. We assume the BS schedules GWs proportionately with the number of nodes on the relay network that they support. For example, a GW terminating a relay network with 4 nodes will be scheduled twice as often as a GW node terminating a relay network with 2 nodes. A single node is treated as a GW supporting a relay network of 1 node.

Thus a node can never achieve higher throughput acting as a single node than when a member of a relay network. In fact, if a node joins a relay network in which the GW node has a higher DRC than itself, its throughput will be increased. Thus, the maximum achievable data rate of node *i*, $R\_relay_i$, occurs when a node is a member of a relay network. Likewise, the maximum average relay network throughput, $\overline{R\_relay}$ is achieved when all nodes join a relay network. These values are given by:

$$R\_relay_i = DRC_{GW} \times \frac{1}{m}$$ , $DRC_{GW}$ = data rate of $GW$ eq. (4)

$$\overline{R\_relay} = \frac{\sum_{i=1}^{m} R\_relay_i}{m} \qquad \text{eq. (5)}$$

We compare the simulated throughput achieved with a single frequency relay network with the theoretical maximum achievable throughput of a 1xEVDO system in which no relay network is used. The throughput was obtained using OPNET according to the parameters in table 1. Figure 17 shows the results obtained from three random topologies of a cell with 30 nodes. It can be clearly seen from Figure 17 that a 1xEVDO system operating with relay networks has better node throughput than a pure 1xEVDO system. Note that these results are extremely conservative: the 1xEVDO system is ideal in that it includes no protocol overheads or impact of errors.

We also note that the performance gain achieved by the relay network varies considerably from topology to topology. In figure 17, the performance gain of the relay network over the pure 1xEVDO system is lowest in topology 2 and highest in topology 3.

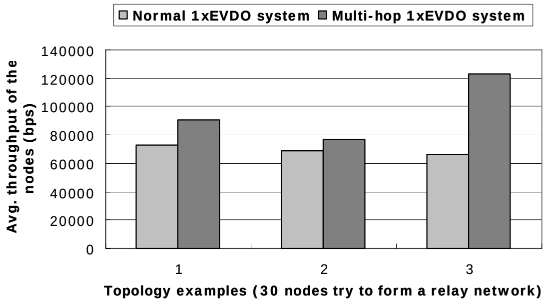


**Fig. 17. Examples of throughput gain**

This can be explained as follows: A relay network improves the performance of a cellular network by exploiting the fact that GW nodes have throughput equal to or higher than the nodes they serve. From table 3 it can be seen that, in topology 2, there are seven nodes with DRC = 1(153Kbps). When the relay network is formed in topology 2, all the nodes that have DRC = 1 originally keep the same data rate because the GW connecting the BS with these nodes also has DRC = 1. Thus these nodes achieve no performance improvement. The slight overall improvement in performance is because the 4 nodes that have DRC = 2(921Kbps) connect to a GW node with DRC = 3(2.5Mbps), thus increasing their throughput. In the case of topology 3, all the nodes that have DRC = 1 are connected through GW nodes that have DRC = 2. Thus a considerable improvement in performance is achieved in this topology.

| | | # of nodes with DRC=3 (2457 Kbps) | # of nodes with DRC=2 (921 Kbps) | # of nodes with DRC=1 (153 Kbps) |
|---|---|---|---|---|
| Topo-logy 1 | Normal | 8 7 | 16 2 | 6 |
| | Relaying | 15 | 11 | 4 |
| Topo-logy 2 | Normal | 6 4 | 17 | 7 |
| | Relaying | 10 | 13 | 7 |
| Topo-logy 3 | Normal | 5 14 | 16 9 | 9 |
| | Relaying | 19 | 11 | 0 |

**Table 3. Examples of nodes' data rate**

### 2) Multi-Frequency Relay Networks

In this section we quantify the performance gains achieved by using a multi-frequency relay network. We consider the relay network formed using the LOF algorithm.

We measure the throughput of each resulting relay network in the following three scenarios.

- **Scenario 1**: The relay network uses a single frequency on all links.
- **Scenario 2**: The relay network may use up to all 8 available frequencies. All nodes in the relay network make frequency assignments according to the frequency assignment scheme in section 4.
- **Scenario 3**: As with scenario 2, the relay network may use up to 8 frequencies. Unlike scenario 2, the frequencies are assigned by a centralized assignment scheme based on the interference constraint in [2]. In order to maximize network throughput, this algorithm assigns a frequency to the link based on an interference constraint that considers a node's interference range as well as transmission range. This scheme makes optimal frequency assignments resulting in no contention and minimum interference.

Besides these basic scenarios, we also quantify the benefits of the local optimization in the case that all nodes can detect interference and reassign a new frequency if needed according to the local optimization algorithm, in addition to scenario 2.

Figures 18 and 19 show that if we exploit multiple frequencies, we can achieve higher overall network throughput and average node throughput. Figure 20 shows the percentage of the throughput achieved by our frequency assignment scheme of section 4 when the optimal throughput which the relay network can achieve is set to 100%. It shows that our frequency assignment scheme achieves at least 84% of the average throughput of the optimal channel assignment algorithm. It also clearly shows that using local optimization results in higher average throughput.

We note that as the number of nodes increases in the network, the improvement in performance of the optimal frequency assignment increases (figure 19). This is because it is more likely that the distributed frequency assignment algorithm will result in two nodes in different relay networks that are within interference range being assigned the same link frequency (see discussion of figure 7).

Figure 21 illustrates this effect. This figure shows the nodes' received SNIR in each scenario with the same topology. It can be clearly seen that there is considerable improvement in SNIR when multiple frequencies are used, and that in some cases, the SNIR when using optimal frequency assignment is higher than when using distributed frequency assignment. However, Figure 21 shows that the local optimization algorithm improves the SNIR to a value significantly close to that obtained with optimal channel assignment.

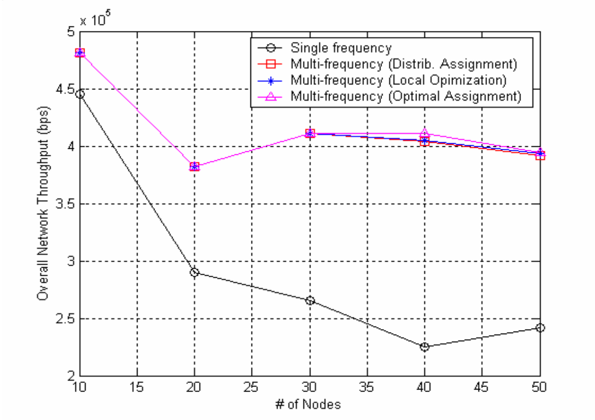


Fig. 18. Overall network throughput

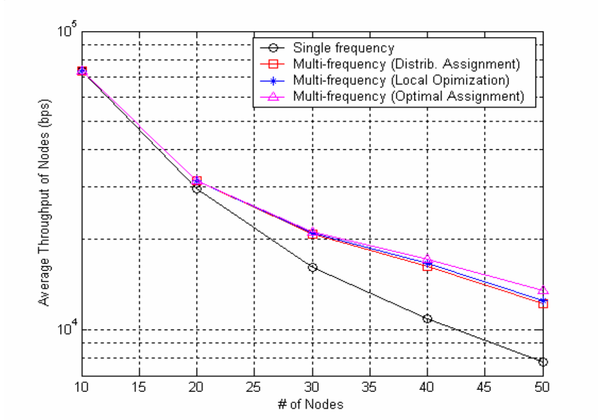


Fig. 19. Average throughput of the nodes in all scenarios

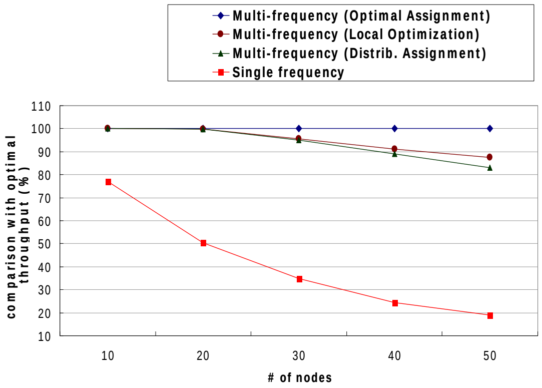


Fig. 20. Throughput Comparison

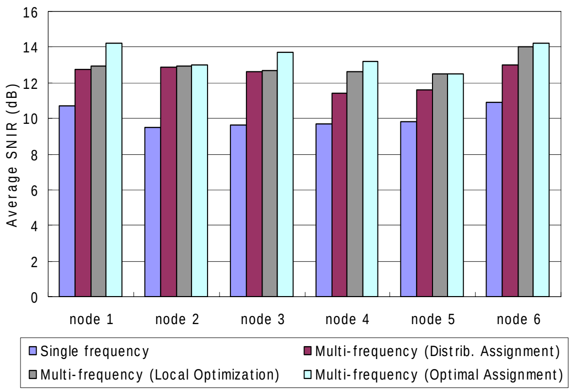


Fig. 21. Example of node's SNIR (Example of 50 node topology)

Since Figures 19 and 20 plot the average throughput of all nodes in the relay network, and not only for nodes that experience an SNIR improvement due to the local optimization, the improvement in throughout is somewhat obscured.

## 7. Discussion

Research [3][15] has shown that in multi-hop wireless cellular networks, the transmissions from a mobile terminal to the BS are broken into multiple wireless hops, and hence require less total transmission power than single hop cellular networks.

In this paper, we assume the use of 802.11 interfaces for relay networks. In [7], it is shown that the 802.11 ad-hoc mode reduces the transmission power significantly compared to the cellular mode and achieves higher network throughput.

However, multi-hop relaying may lead to higher energy consumption for gateway nodes. In this paper, the GW is chosen depending only its signal quality from the BS. We can consider an improved GW selection metric that includes the residual battery power of the node in addition to the signal quality from the BS. Based on this metric, the GW can be reselected periodically for balancing energy consumption.

## 8. Conclusion

In this paper we analyzed the formation of relay networks for dynamic multi-frequency, multi-hop wireless cellular networks. We propose two centralized algorithms and one distributed algorithm for network formation. While establishing paths to the GW nodes, mobile nodes can make non-interfering frequency assignments to the relay links based on limited hop information. As a result, the number of interfering links can be reduced and hence we can achieve improved network throughput. For further improvement of network throughput, we also propose two enhancements to support local optimization and GW reselection.

Our results show that schemes scheduling nodes furthest from the BS to initiate route discovery first make good use of passive route discovery and hence reduce the relay network formation overhead. Moreover, they can build efficient relay networks which attain high throughput. The distributed LOF algorithm achieves a good trade-off between relay network formation overhead and latency.

We measured the throughput of the relay network resulting from the LOF algorithm. The results show that by exploiting multiple frequencies, we can achieve higher overall network throughput and average node throughput. Our simple distributed frequency assignment scheme achieves 80-85% of the optimal average node throughput. Moreover, the enhancements further improve average throughput.

## References

[1] G.M. Aggelou and R. Tafazolli. On the relaying capability of next generation GSM cellular networks. IEEE Wireless Communications, 8(1):40–47, 2001.

[2] Mansoor Alicherry et al. Joint Channel Assignment and Routing for Throughput Optimization in Multi-radio Wireless Mesh Networks. In Proc. of ACM MOBICOM, 2005.

[3] A. Ananda Kusuma, L. Andrew. Minimum Power Routing for Multihop Cellular Networks, In Proc. of IEEE GLOBECOM, 2002.

[4] P. Bender et al. CDMA/HDR: a bandwidth-efficient high-peed wireless data service for nomadic users, IEEE Communications Magazine, pp.70-78, vol. 38, 2000.

[5] J. Glossner, et al. A software-defined communications baseband design. IEEE Comm. Magazine, 41(1):120–128, 2003.

[6] W. Hongyi, Q. Chunming, S. De, and O. Tonguz. Integrated cellular and ad hoc relaying systems: ICAR. IEEE Journal on Selected Areas of Communications, 19(10):2105–2115, 2001.

[7] H. Hsieh and R. Sivakumar. Performance Comparison of Cellular and Multi-hop Wireless Networks: A Quantitative Study, In Proc. of ACM SIGMETRICS, 2001.

[8] A. Jalali et al. Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system. In Proc. of IEEE VTC, 2000.

[9] David B. Johnson and David A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," In Mobile Computing, pages 153-181, Kluwer Academic Publishers, 1996.

[10] B. Ko, V. Misra, J. Padhye, D. Rubenstein, “Distributed Channel Assignment in Multi-radio 802.11 Mesh Networks”, Tech Report, Columbia Univ., 2006

[11] Murali Kodialam et al. Routing and Channel Assignment for Throughput Maximization in Multi-radio, Multi-channel Wireless Mesh Networks. In Proc. of ACM MOBICOM, 2005.

[12] Pradeep Kyasanur et al. Capacity of Multi-Channel Wireless Networks: Impact of Number of Channels and Interfaces. Tech. Report in Department of Computer Science at UIUC, 2005.

[13] H. Luo et al. UCAN: a unified cellular and ad-hoc network architecture. In Proc. of ACM MOBICOM, pages 353–367, 2003.

[14] Charles E. Perkins et al. Ad-Hoc On Demand Distance Vector Routing. In Proc. of IEEE WMCSA, 1999.

[15] T. Rouse, I. Band, S. McLaughlin. Capacity and Power Investigation of Opportunity riven Multiple Access (ODMA) Networks in TDD-CDMA Based Systems, In Proc. Of IEEE ICC, 2002.

[16] JaeSheung Shin, Parthu Kishen, Thomas F. La Porta. Dynamic Multi-Frequency, Multi-Hop Wireless Cellular Network. In Proc. of IFIP the fifth annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET), 2006.