United Nodes: A Cluster based Routing Protocol for Mobile Cognitive Radio Networks

A. Cagatay Talay and D. Turgay Altilar
Dept. of Computer Engineering
Istanbul Technical University
34469, Istanbul, Turkey

Abstract—**In Mobile Cognitive Radio Ad Hoc Networks routing is one of the most important issues to be addressed and desires deep investigation. In this paper, we propose a distributed and efficient cluster based interference aware routing protocol. It incorporates the spectrum availability cost and interference metrics into the routing algorithm to find better routes. A route preservation method is also incorporated in the proposed algorithm to repair the route when it is defective due to primary user activity. Results of ns2 simulations illustrate that, the proposed algorithm can well fit into the mobile cognitive radio ad hoc networks and improve the network performance. The results indicate that The UNITED provides better adaptability to the environment and increases throughput and reduces data delivery latency.**

I. INTRODUCTION

Radio Spectrum is amongst the most heavily used and expensive natural resource around the world. Although almost all the spectrum suitable for wireless communications has been allocated, recent studies and observations indicate that many portions of the radio spectrum are not used for a significant amount of time or in certain geographical areas while unlicensed spectrum bands are always crowded [1]. As a promising solution to scarce spectrum resource, Cognitive Radio (CR) [2] was proposed to enable unlicensed (secondary) users to sense and intelligently access the unoccupied spectrum portions that are not used by the licensed (primary) users at that specific time and location. The main components of an example Cognitive Radio Network (CRN) can be classified into two groups: the licensed (primary) network and the CR (secondary or unlicensed) network. The licensed network is referred to as an existing network, where the primary users have licenses issued by the government licensing authorities to operate in certain spectrum bands. Due to their priority in spectrum access, the operations of primary users must not be affected by unlicensed users. The CR network does not have and require a license to operate in a desired band. The CR users have the opportunity to use both licensed and unlicensed spectrum bands. Since all interactions occur inside the CR network, their spectrum sharing policy can be independent of that of the primary network. In Multi-Hop Mobile Cognitive Radio Networks, the CR nodes sense spectrum and identify available frequency bands, named as Spectrum OPportunities (SOP) or white holes [3], then select one candidate from SOP via predetermined specific policy, which will not cause harmful interference to the licensed nodes. Based on the sensed information, CR users access the licensed band opportunistically when no primary users are using that band and vacate the band immediately upon primary user activity detection. Using these unoccupied channels provides a more effective way to increase the overall network capacity. These new opportunities come with the expense of some important challenges such as routing problem in such a dynamic environment. Facing these challenges and research issues, several approaches have been introduced. One of the proposed approaches is clustering which is used to strategically partition the network into smaller segments. Using such an approach in mobile ad hoc networks has important benefits including optimizing bandwidth usage, balanced distribution of resources and resolving scalability issues in combination with routing schemes.

Clustering schemes can be classified into two, clusters with cluster heads (CHs) and ones without. Clusters without CHs avoid overloading a subset of nodes in the network, making the operation of all nodes equal. However, CHs may serve many purposes within a cluster, such as the allocation of resources to member nodes and coordinating transmission events for nodes in the cluster in order to avoid retransmissions by reducing packet collisions [4]. Clusters controlled by CHs can be organized as either 1-hop clusters or multi-hop clusters which are also known as k-hop clusters. In 1-hop clustering schemes cluster members (CMs) are within transmission range of the CH, that is, within 1-hop of the CH. In multi-hop clustering schemes, the maximum distance between the CH and a CM is k hops, such that CM may reside outside the communication range, where intermediate CM relay messages between CHs and those members. A number of clustering algorithms have been proposed for wireless ad hoc networks [4]–[7], as well as for CR networks [8], [9]. To the best of our knowledge, all of these algorithms considered the spectrum sensing problem except one [10]. Chen et al. [10] proposed a framework based on the use of clustering for cognitive radio networks.

In this paper, we propose the UNITED, United Nodes: A Cluster based Routing Protocol for Mobile Cognitive Radio Networks, for maximizing the network throughput and minimizing the end-to-end delay. The UNITED operates autonomously in a distributed manner at every node. First, nodes organize themselves into several clusters by the clustering algorithm that is based on location, communication efficiency, network connectivity and spectrum availability. Clusters adapts themselves to the dynamic spectrum availability, and to the high mobility of the nodes. After cluster formation, routing is done according to the spectrum usage and interference metrics.

The remaining part of the paper is organized as follows. Details of the UNITED is given in Section II. The simulation and performance analysis of the UNITED is presented in Section III. Finally, conclusions and future research directions are provided in Section IV.

II. UNITED NODES: A CLUSTER BASED ROUTING PROTOCOL FOR MOBILE CR NETWORKS

A. The Clustering Algorithm

In this study, a mobile cognitive radio ad hoc network environment with primary and secondary nodes, where all nodes communicate with each other in their own networks, is considered. There is no communication (i.e., no cooperation) between primary and secondary networks. The network is modeled as a graph $G = (N, L)$ where N is a finite set of nodes,and L is a finite set of unidirectional links. The set of nodes N will be partitioned into M (i.e. we have M CHs) clusters $\{C_1, C_1, \ldots, C_M\}$. Let $E = \{e_i = (x_i, y_i), i = 1, \ldots, N\}$ be the set of node coordinates. The Euclidean distance between node p and node q is defined as

$$dist(e_p, e_q) = \parallel e_p - e_q \parallel = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2}. \quad (1)$$

The minimum distance from node p of one cluster to another cluster C_j is

$$dist^*(e_p, C_j) = min\{dist(e_p, e_q) : e_q \in C_j\}. \quad (2)$$

The maximum directed distance from cluster C_i to C_j for e_p, denoted as $D(C_i, C_j)$ is

$$D(C_i, C_j) = max\{dist^*(e_p, C_j) : e_p \in C_i\}. \quad (3)$$

Consequently, the maximum distance between clusters C_i and C_j is the bigger of the two directed distance,

$$D^*(C_i, C_j) = max\{D(C_i, C_j), D(C_j, C_i)\}. \quad (4)$$

If the maximum distance is d, then every node in the cluster C_i must be within a distance d from some node in C_j and vice versa.

We now describe a distributed clustering algorithm where nodes make autonomous decisions. It is a scalable algorithm, and it can cope with small to extremely large networks. The algorithm must operate in conditions where node mobility and fluctuation in the available spectrum is very high. The clusters should be capable of adapting to cope with such dynamic conditions due to mobility and more importantly to the primary user activity. Formed clusters must cope with the abrupt channel evacuation. Since these challenges are the natural characteristics of mobile ad hoc networks and the features of the used cognitive radios. All nodes in the network are clustered $(C_i \cup \ldots C_M)$, and each node is allowed to join only one cluster $(C_i \cap C_j = \emptyset)$. If a node is a starter node of a cluster, then it is a member of that cluster naturally. If a node is a isolated node (i.e. do not have any neighbor nodes), it forms a single member cluster. If a node is not a isolated node, then it must be a neighbor of a cluster member node and will join to that cluster.

Clustering algorithm is based on the use of a combined weight metric, that takes into account several parameters like distance, transmission power, mobility, the battery power of the nodes, and the sensed information about the available spectrum. Given the chance of changing the weight factors helps us to determine the best metric for various networks. Also we limit the number of nodes (δ) that a CH can accept as a member node for being able to balance the load in the network, and to ensure the efficiency of the network is kept above an expected level. The CH election procedure (Algorithm 1) is invoked at the time of system activation, when there is a drastic change in the network and also when a CH is under the influence of a primary user. This reduces system updates, hence computation and communication costs.

Algorithm 1: Cluster Head Selection (G)

```
repeat
    ∀ node $n$, compute :
        the node degree $\Delta_n = |d_n - \delta|$;
        the mobility measure $M_n(t)$;
        spectrum availability $Sp_n(t)$;
        weighted value $W_n = \alpha\Delta_n + \beta M_n(t) + \gamma Sp_n(t)$
    choose the best node as the Cluster Head remove neighbor nodes of the chosen CH from set $G$
until $G = \emptyset$ ;
```

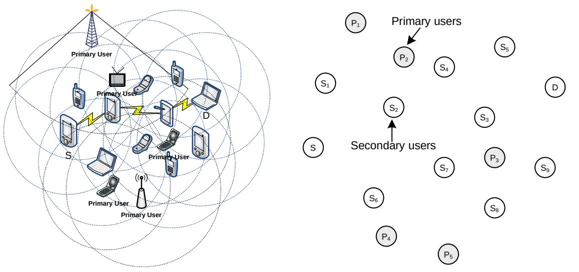
A node that is not belong to any previously constructed cluster is said to be in the unclustered state. In unclustered state, a node cannot inform its neighbors of its presence, nor can it receive information about the neighboring nodes. A node entering a new environment will at first start off in the unclustered state, and also a node will enter the unclustered state if its link to another node fails or aroused after detection of primary user activity. Cluster formation process is described below and given in Algorithm 2.

Algorithm 2: Cluster formation

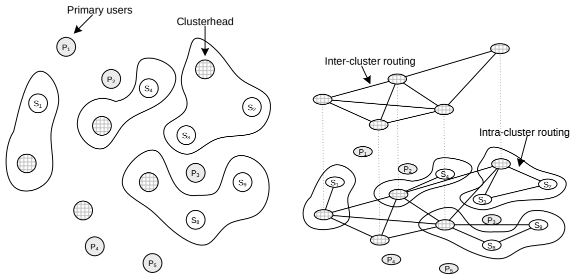
```
1  if $S \subset N$ and $S \neq \emptyset$ then
2      order($S$);
3      $i \leftarrow max(\forall j \in S)$;
4      Join_Request($i$);
5      if Join_Response then
6          if connection denied then
7              go to line 13 ;
8          else
9              return clustered;
10     else
11         $S \leftarrow S - \{i\}$;
12         if $S = \emptyset$ then
13             go to line 21;
14         else
15             go to line 4;
16 else
17     repeat
18         send hello packet with reply request;
19     until $R \neq \emptyset$ ;
20     $S \leftarrow R$;
21     go to line 3;
```

Each node n in the network has a neighbor node table that holds local information about the neighboring nodes like *ID*, *Speed*, *Location*, *Direction*, *Cluster Size* and *Cluster Membership*. Every information is time-stamped to allow expiration after a predetermined threshold, Δt_i. The neighbor table is defined by set N. The CMs of a CH is associated with the set M, and the CHs are associated with the set H.

If a node is in unclustered state, it has to know its 1-hop neighbors before attempting to join a cluster. Therefore, every node has to have a list of its 1-hop neighbors denoted by set S, to determine the most suitable CH neighbor to cluster with. A node, that became unclustered state due to primary user



(a) CR network illustrated (b) 2D mapping of the CR network



(c) Clusterheads selected and clusters formed (d) Connectivity in example cognitive radio network for routing

Fig. 1. A mobile CR ad hoc network example

activity, may already have neighbor information in its table N. Therefore, when a node enters the unclustered state, it first checks its neighbor table. If there is information about 1-hop neighbor nodes that are not expired, it selects these nodes and constitutes set S, note that ($S \subset N$). However, if the neighbor table is empty, then the node enters the request phase. In the request phase, the node tries to discover its neighbors by broadcasting periodic *HELLO* packets. (If the spectrum sensing algorithm also uses *HELLO* messages, there is no need to use additional messages in the proposed scheme.) Note that, in our protocol design a node rarely enters this phase. Generally a node that becomes unclustered due to primary user activity will have sufficient information in its neighbor table to identify its 1-hop neighbors when it has opportunity to access the spectrum again.

Algorithm 3: Order (p, q)

```
output : p: node $p$ is predecessor;
         q: node $q$ is predecessor;
begin
    if $\Delta_p < \delta$ then
        return p;
    else if $\Delta_q < \delta$ then
        return q;
    else if $W_p < W_q$ then
        return p;
    return q;
end
```

The *HELLO* packet contains the node ID and also spectrum and mobility information, which is used by the receiving nodes to filter packets originating from nodes that are moving away in the opposite direction and has little spectrum access opportunity due to heavy primary user activity in that area. Neighboring nodes upon receiving the *HELLO* packet, first check if they they have reached their maximum connection limit δ and respond with unicast response (*RESP*) packets, containing their own information. This information includes their spectrum and mobility parameters, that is, spectrum opportunities, location, speed and direction of travel. Upon receiving the first (*RESP*) packet, receiving node initiates a timer and collects all responses from its neighbors into a list R until the timer expires after a predetermined period. This gives all neighboring nodes chance to find a clear channel and respond to the request. The request process is repeated while $R = \emptyset$. Otherwise the responses are placed into list S, where $S \neq R$ After the unclustered node (e.g. node a) identifies its 1-hop neighbor CHs, such that $S_a \neq \emptyset$, the neighbors in S_a are sorted using order algorithm, which is given in Algorithm 3 for node a clustering. The sorted list S_a will contain the neighbors in order, with the most suitable node to be CH for the node a at the top. The Order algorithm uses weighted metrics of nodes to determine order which based on sensed spectrum availability, relative mobility (location, speed, and direction). Initially the order algorithm checks to see if either neighbor has reached their maximum connection limit. If node p has and node q has not, then q is set as predecessor, and vice versa. At the next level, node a uses the weighted value which is calculated in the cluster head selection process. After sorting list S, the node sends a join request *JOIN_REQ* packet to the node that is at the top of S, requesting that a new link is formed (Algorithm 2). The neighbor receiving the *JOIN_REQ* packet replies with join response *JOIN_RESP* packet, assigning the role of CM to the node that sent the *JOIN_REQ* packet, or denying the connection. If the neighboring node is already a CH, the initiating node joins that cluster. However, if the neighboring node is a member of another cluster, or is in the unclustered state, a new cluster is formed with the neighbor becoming as the CH. The neighboring node receiving the *JOIN_REQ* packet will deny the connection attempt if it has reached its cluster size limit δ. If a connection request is denied, or a *JOIN_RESP* packet is not received after a timeout period, that neighbor is removed from list S, and the node attempts to join the neighbor that has the next highest order in the list S. This process is repeated each time a join request fails until $S = \emptyset$, after which the node enters the request phase.

B. *The Routing Algorithm*

Cognitive radio users are not likely to access spectrum randomly, or have a path to a specific node definitely random but rather get connected in a predictable fashion based on repeating behavioral patterns such that if a node has gain opportunity to access a specific unused spectrum band and has connection to a specific node several times before, it is likely that it will gain connection to that node again. We would like to make use of these observations and information to improve routing performance by defining a cost metric and doing routing according to this metric.

Considering the primary users operate on the primary channels and CR links access the temporarily unused portions of the primary channels on an opportunistic basis. We assume the usage pattern of the primary users which affects an arbitrary CR link i follows an independent two-stage ON/OFF random process. An ON period $T_{on,i}$ represents the time that the primary users are active and interference to the primary users happens if CR link i transmits on the primary channel during that time. An OFF period $T_{off,i}$ represents the time that the primary users are inactive and CR link i has access to the spectrum. To simplify our analysis, we assume both $T_{on,i}$ and $T_{off,i}$ are exponentially-distributed with means equal to $1/\mu_i$ and $1/\lambda_i$ second respectively. The ON/OFF random processes of the primary users activity pattern affecting different CR links are assumed independent. From the perspective of inside the CR system, we should choose routes with the best end-to-end performances and from the perspective of system coexistence, routes should be selected with the minimum interference to the primary systems. When considering the CR system's end-to-end throughput of a route, interference from other CR links along the route should also be taken into account. We call this interference as intra-system interference. Again, we would like to make use of these observations and information to improve routing performance by defining a interference aware metric. Our metric considers the effects of variation in link loss ratio, differences in link transmission rate as well as inter-system and intra-system interference.

1) Spectrum Availability Cost: Spectrum availability is an inherent characteristic in mobile CR Ad Hoc Networks where nodes usually get disconnected due to characteristics of ad hoc networks and CR technology. In UNITED, we track a link's connectivity behavior and assign a persistent cost metric that gets updated periodically to reflect its overall state. Accordingly, if a link is disconnected for a long time, the cost is increased to a high value and for a well connected link the cost will be kept to a small value. In this way, a route can be found between a source and a destination even if there is no continuous end-to-end connectivity. We accomplish this by assigning larger costs to links with larger spectrum unavailability durations. Moreover, in situations where multiple such links have similar average spectrum unavailability durations the link with a history of less frequent disconnection-to-connection transition is assigned a lower cost. The rationale behind this is, for a given spectrum unavailability duration, a link that transitions' less frequently is a better link as it reflects a node has more opportunity to forward a packet to the other nodes. With these guiding principles, the cost of a directional link $L_{i,j}$ is defined as:

$$C_{i,j} = \frac{1 + \left(T_{cost\_window} - \sum_{k=1}^{N_{i,j}^{transition}} T_{i,j}^{k} \right)}{1 + N_{i,j}^{transition}}. \tag{5}$$

$C_{i,j}$, cost of link $L_{i,j}$, is dynamically computed by node i based on its spectrum usage history over a discrete sliding window of length T_{cost_window}. Within a measurement window, the number of times the link status transitions from having opportunity to transmit without any interference to any primary user to causing interference is represented by the parameter, and the duration of the k^{th} connectivity instance is represented by $T_{i,j}^{k}$. The term $\sum_{k=1}^{N_{i,j}^{transition}} T_{i,j}^{k}$ represents the total cumulative connectivity duration within the last measurement window. For a non-disturbed secondary link, this term equals the duration of the measurement window itself, and $N_{i,j}^{transition}$ equals to zero hence the cost reduces to unity. Since this is the minimum possible link cost, a fixed link will always be preferred over interfering links by any link state routing algorithm. Also, since the numerator of the expression for $C_{i,j}$ is dominated by the cumulative link primary user activity time, links with longer interfering times will have higher cost and thus will be avoided by the least cost algorithms. However, among multiple links with similar cumulative disconnectivity durations, the ones with lower transition counts $N_{i,j}^{transition}$ will have lower costs. This ensures that among all links that have similar cumulative disconnection periods, the least cost routing algorithms will not prefer links that cause interference with primary users more frequently. Note that the upper bound of the link cost will be decided by the parameter T_{cost_window} which is set dynamically by node i as a multiple of the measured periodicity of link $L_{i,j}$.

2) Interference Cost: The impact of interference on the network performance is a parameter difficult to estimate. In order to have an accurate view of the current channel (link) state, it is necessary to factor in not only indicators of the channel quality such as nominal throughput or packet loss, but it is also critical to estimate the transmission delay resulting from concurrent data transmissions. The broadcast nature of the wireless medium forces the nodes at interference range of a given source and destination to wait for the medium to be cleared before to have access to it. Consequently, a routing metric properly tailored for CR networks that accounts for these different factors can improve the overall network performance by avoiding lossy links and congested zones.

If we assume that there is no interference in the network, a previously proposed routing metric for ad hoc networks, expected transmission time (ETT) [11] metric gives an idea about the quality of the link quite well as links with less expected transmission time give better throughput. But when there are more interfering flows in the network, unfortunately this is not the case. We need to factor in the varying interference experienced by a link into the routing metric to find paths with better quality. In order to realize this, we need to model interference properly and factor it in the routing metric appropriately.

We use the physical interference model [12] to capture the interference experienced by links in the network. In this model, a communication between nodes m and n is successful if the *SINR* (Signal to Interference and Noise Ratio) at the receiver n is above a certain threshold which depends on the desired transmission characteristics. *SINR* provides useful information on how strong the desired signal is compared to the interferer plus noise in the network. Denoting the signal strength of a packet from node m at node n by $P_n(m)$, a packet on the link $L_{(m,n)}$ from node m to node n is correctly received if

$$\frac{P_n(m)}{N + \sum_{k \in V'} P_n(k)} \geq \beta \tag{6}$$

where N is the background noise, V' is the set of nodes simultaneously transmitting and β is a constant. Considering all partially interfering nodes, *SINR(m)* can be defined as

$$SINR(m) = \frac{P_m(n)}{N + \sum_{k \in \vartheta(m) \setminus \{m,n\}} \Gamma_k \; P_m(k)} \tag{7}$$

where the received interfering signal from node k is weighted using node k's transmission rate Γ_k, which is the normalized rate averaged over a period of time. It gives the fraction of time node k occupies the spectrum. The set of nodes that node m can hear or sense is associated with the set $\vartheta(m)$. We define interference ratio $I_i(m)$ for a node m in a link $L_{m,n}$ as the ratio of interference to the maximum interference (P_{int}^{max}) that a node can still communicate properly, and denoted as

$$I_i(m) = \frac{\sum\limits_{k \in \vartheta(m) \setminus \{m,n\}} \Gamma_k \; P_m(k)}{P_{int}^{max}} \tag{8}$$

where $(0 \leq I_i(m) \leq 1)$. When considering a bidirectional link $L_{m,n}$, I_i is

$$I_i = max(I_i(m), I_i(n)). \tag{9}$$

We define the interference metric of a link l as

$$int_l = ETT_l * \varphi(I_l). \tag{10}$$

where $\varphi(.)$ is the scaling function. ETT_l is weighted with I_l to capture the interference experienced by the link from all of its neighbors including primary users. Naturally, lower values of the int of a link indicates a better link.

3) Combined Routing Metric: We can combine the desirable properties of the two metrics described in (5) and (10) by taking their weighted average:

$$cost = \alpha * C_{i,j} + (1 - \alpha) * int_l \tag{11}$$

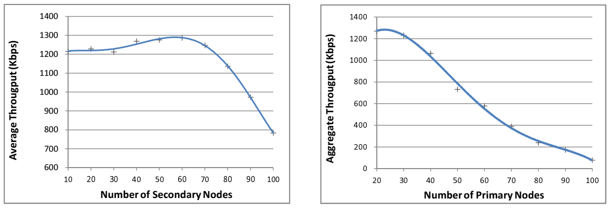
where α is a tunable parameter subject to $0 \leq \alpha \leq 1$. The weighted average can be viewed as an attempt to balance between the spectrum availability and interference cost metrics.

4) Forwarding Strategy: Each node in a cluster can populate its routing table for intra-cluster routing based on the topology using shortest path algorithms considering our metric mentioned above. Each route computed is associated with a lifetime. The route will be removed when it is expired. If a node cannot find a route for the destination of a packet, the data packets will be forwarded to its default route $R_{default}$ and then eventually to the cluster head. This is common when the destination node is in a different cluster. In such case, the cluster head will forward the packet to the cluster head of the destination cluster and in turn the packet is forwarded to the destination node. The details of data forwarding algorithm is shown in Algorithm 4.

Algorithm 4: Forwarding Strategy

```
//For any node $n_i$ and packet $p$;
if dest(p)=$n_i$ then
    receive $p$;
else
    next_hop ← search_rtable($p$);
    if next_hop$\neq \emptyset$ then
        forward($p$, next_hop);
    else if is_ClusterHead($n_i$) then
        drop $p$;
    else
        forward($p$, $R_{default}(n_i)$);
```

5) Route Preservation(Local Repair): Due to the nodes mobility or primary user activity it is necessary to have a maintenance in the routing protocol, where each node has to corroborate the area it belongs to, and update information for the mobility factor and spectrum availability. Every time there is a new CH, all nodes must receive a notification message to know where to transmit the packets. Route maintenance can be done by: (a) jumping the broken node if the next-next hop in the path is reachable; (b) choosing another reachable node(s), which is far from the primary user, to be the next hop which is reachable by the previous node and the next node in the path.



(a) 20 primary nodes in the network (b) 50 secondary nodes in the network

Fig. 2. Average throughput of the UNITED

III. SIMULATION AND PERFORMANCE ANALYSIS

Through simulations constructed in ns2, the performance and functional correctness of the UNITED is evaluated. Unless otherwise noted, simulations are run with the following parameters. Two-ray ground propagation model is used at the radio layer. The bit rate for each channel is 2Mbps. Variable number of mobile nodes up to 100 moving in a rectangular area 1800 m x 1800 m in dimension is modeled. Each node picks a random spot in the rectangle and moves there with a speed uniformly distributed between $0 - 10$ m/s. Upon reaching this point, the node picks a new destination and repeats the process. We model the primary users' activities by using the exponential ON-OFF process as mentioned before. The coverage range of the primary user on its operation channel is 250m. These parameters are set since similar to the default values used in previous study of various protocols. Thus a comparison among the protocols can be done. The following default communication pattern is used. Each source node generates and transmits constant bit rate (CBR) traffic and each message is 1KB in length. The transmission interval for each node is set to 100ms. We also injected Voice-over-IP (VoIP) traffic into the network to make a more realistic scenario. A total of six VoIP CR users are randomly distributed over $8 - 128$Kbit/s with random arrival rates (including packetization intervals according to the codec G.711, G.726 and G.729 recommendations). 50 experiments are performed in random multihop network topologies, for each different parameter settings.

The characteristics of the UNITED are explored under a number of different scenarios. The robustness of UNITED is investigated for various numbers of both primary and secondary nodes, stressing the impact of adaptive transmission range on the throughput performance. The simulations are run for networks of sizes 10 to 100 secondary nodes and 20 to 100 primary nodes. It is shown that since the node density has a great importance on the performance of the UNITED for retaining the path and for the success of the local repair, the UNITED performs high throughput for dense networks. However, after a certain threshold throughput starts to decrease due to the congestion. This situation is illustrated in Fig. 2(a). As expected, average throughput is inversely dependent on the number of primary nodes as seen in Fig. 2(b).

Throughput and end-to-end delay comparisons have been evaluated for the UNITED to show that the UNITED can well

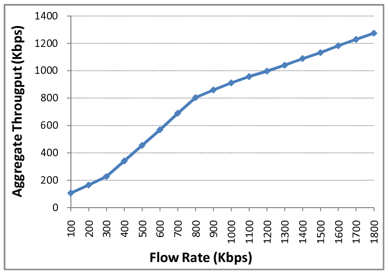


Fig. 3. UNITED performance: throughput observed when two hosts, S and D communicates

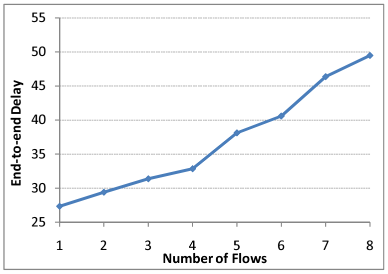


Fig. 4. Average end-to-end delay performance vs. number of flows compared to different protocols

fit the multi-flow multi-channel environment and effectively exploit the potential large communication capacity in CR networks. In the simulations, the rate of flows is varied from 100 Kbps to 1800 Kbps. The nodes are randomly placed in the area, and 8 flows having the same traffic generation rate. As the traffic load becomes higher, the performance improvement of UNITED becomes more significant due to path retaining and local repair. In a dynamic environment, which means the network topology varies frequently, the UNITED adapts to the environment to retain the secondary nodes communication path well or performs a local repair when the distance to the destination is not reachable without harmful interference to the primary user. Also since we use interference as a routing metric, the established route in the UNITED is better in a frequently varying environment. The result is illustrated in Fig. 3.

The end-to-end delay performance of the UNITED is also evaluated. We adjust the number of intersecting flows from 1 to 8 to evaluate the performances upon intersecting flows. The simulation result is shown in Fig. 4. When the number of flows increases, the UNITED seeks a balance between assigning new frequency bands to allow simultaneous transmission and accommodating some nodes on one band to avoid switching delay, also the re-route establishment time is low in the UNITED upon a primary user activity detection. Consequently, the UNITED achieves an overall optimal delay as the number of intersecting flows grows.

To assess the effectiveness of the UNITED, we have also used the normalized routing overhead as a performance metric. Normalized routing overhead can be defined as the total number of control ($N_{control}$) and data packets sent (N_{data}) normalized by the total number of packets successfully delivered in the CR network, also considering the number of flows. The normalized routing overhead of the UNITED is illustrated in Fig. 5.

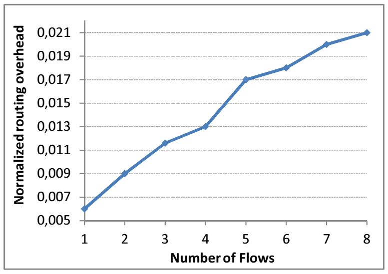


Fig. 5. Normalized routing overhead vs. number of flows

IV. CONCLUSION

In this paper, Cognitive Radio Ad Hoc Networks are investigated and the UNITED, United Nodes: A Cluster based Routing Protocol for Mobile Cognitive Radio Networks, for maximizing the network throughput and minimizing the end-to-end delay is proposed. The UNITED is an autonomous distributed cluster based routing algorithm for mobile cognitive radio ad hoc networks that simultaneously considers the requirements of primary and secondary users. It uses spectrum availability cost and interference metrics to find better routes. Through an implementation in the ns2 simulator, it has been shown that the UNITED achieves significant improvement on the throughput and the end-to-end delay. The adaptability and efficiency of the scheme is proved in simulations.

REFERENCES

[1] J. M. Peha, “Approaches to spectrum sharing,” *IEEE Commun. Mag.*, vol. 43, no. 2, pp. 10–12, Feburary 2005.

[2] J. Mitola, “Cognitive radio: An integrated agent architecture for software defined radio,” PhD Dissertation, Department of Teleinformatics, Royal Institute of Technology (KTH), Stockholm, Sweden, May 2000.

[3] I. F. Akyildiz, W. Y. Lee, M. C. Vuran, and S. Mohanty, “A survey on spectrum management in cognitive radio networks,” *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 40–48, April 2008.

[4] J. Y. Yu and P. H. J. Chong, “A survey of clustering schemes for mobile ad hoc networks,” *IEEE Commun. Surveys Tuts.*, vol. 7, no. 1, pp. 32–48, First Quarter 2005.

[5] M. Chatterjee, S. K. Das, and D. Turgut, “WCA: A weighted clustering algorithm for mobile ad hoc networks,” *Cluster Computing*, vol. 5, no. 2, pp. 193–204, 2002.

[6] V. Kawadia and P. R. Kumar, “Power control and clustering in ad hoc networks,” in *IEEE INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, p. 459469.

[7] O. Younis and S. Fahmy, “HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,” *IEEE Trans. Mobile Comput.*, pp. 366–379, 2004.

[8] C. Sun, W. Zhang, and K. B. Letaief, “Cluster-based cooperative spectrum sensing in cognitive radio systems,” in *Proc. of the IEEE International Conference on Communications (ICC ’07)*. Glasgow, Ireland: IEEE, 24-28 June 2007, pp. 2511 – 2515.

[9] C. Qi, J. Wang, and S. Li, “Weighted-clustering cooperative spectrum sensing in cognitive radio context,” in *Proc. of the WRI International Conference on Communications and Mobile Computing (CMC ’09)*. Yunnan: IEEE, 6-8 Jan. 2009, pp. 102 – 106.

[10] T. Chen, H. Zhang, X. Zhou, G. M. Maggio, and I. Chlamtac, *CogMesh: A Cluster Based Cognitive Radio Mesh Network*. The Netherlands: Springer, November 2007, ch. 34.

[11] R. Draves, J. Padhye, and B. Zill, “Routing in multi-radio, multi-hop wireless mesh networks,” in *Proc. of the of the 10th annual international conference on Mobile computing and networking (MobiCom ’04)*. Philadelphia, PA, USA: ACM, September 26 - October 1 2004, pp. 114–128.

[12] P. Gupta and P. R. Kumar, “The capacity of wireless networks,” *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, March 2000.