

Communication Mechanisms for Cognitive Radio Networks

Alexandru Popescu, Yong Yao, Markus Fiedler and Adrian Popescu

*Dept. of Communications and Computer Systems
School of Computing, Blekinge Institute of Technology
371 79 Karlskrona, Sweden
app@bth.se, yya@bth.se, mfi@bth.se, apo@bth.se*

Abstract—Comprehensive communication in cognitive radio networks is an important research topic within the scope of empowering cognitive radio functionality in beyond-4G mobile networks. Providing communication for secondary users without interference with primary users is an ambitious task, which requires innovative management architecture designs and routing solutions. Operational challenges such as opportunistic spectrum access, solving problems related to spectrum and network heterogeneities and requests for the provisioning of Quality-of-Service to different applications must be resolved. As part of a novel management architecture, the paper advances a new approach to end-to-end communication in cognitive radio networks based on combining ad-hoc algorithms with spectrum mobility algorithms.

Keywords—routing; constraint optimization; communication mechanisms; spectrum access;

I. INTRODUCTION

The cognitive radio (CR) topic has generated intense research all over the world. Recently funded projects in the European Union (EU) Seventh Framework Programme (FP7) focus on topics like spectrum and energy efficiency, primary user (PUs) impact by secondary users (SUs) opportunistically accessing the spectrum, and spectrum access and management integration with core networks [1].

Rethinking classical wireless network architectures, cross-layer designs allow for the management requirements of multihop cognitive radio networks (CRNs) to be considered by cooperative spectrum-aware communication protocols [2], [3]. Typically, the cognitive framework to achieve this can be implemented in different ways like, centralized, distributed or mixed. These architectures have specific advantages and drawbacks, meaning the trade-offs between these modes of operations need to be considered. For instance, the most obvious trade-offs are single-point of failure in the centralized mode as opposed to increased overhead requirements to maintain the topology in the distributed mode. In particular, the cognitive process collects relevant information, does the actual learning and decides on appropriate actions in response to the observed network behavior [4]. The cognitive process can widely vary depending on purpose, amount of data collected and processed, implementation, performance and other particular conditions. However, there is often a cost in the form of network strain for communicating network state information to keep a centralized cognitive

process updated. Although, since a decentralized approach to CR functionality, means that adaptations to operational parameters of cognitive radio devices (CRDs) are mainly based on local observations done by independent CR users, this is necessary to achieve comprehensive optimization according to parameters like power, cost and throughput. The lack of centralized support offers a limited knowledge of the network topology, which makes reliable route and packet delivery (commonly based on adaptations of traditional distributed ad-hoc routing mechanisms) in a multi-hop CR environment a difficult undertaking. Expanding on this concept, a new architectural approach is advanced in [3] for the management of CRNs, where additional functionality is provided by a centralized entity called Support Node (SN). The aim is to increase the spectrum utilization within particular geographical areas by integrating CR functionality with low power base stations in 4G networks. Designed and developed at the application layer, the suggested CRN management architecture incorporates sensing and prediction, addressing and routing, middleware and decision making. The focus in this paper is on the addressing and routing part of the CRN management architecture. The goal is to devise an efficient routing solution for end-to-end (e2e) communication between network hosts.

The remainder of the paper is organized as follows. Section II describes the basic functionality of the CRN management architecture. Section III provides a short overview of the communication model. In section IV we discuss the implementation of the network model used for communication between network hosts. The simulation results are reported in Section V and section VI concludes the paper.

II. BASIC SCENARIO

A cognitive network is considered to be composed of multiple smaller networks, so-called CRNs, belonging to different geographic domains where every CRN is served by a SN and contains a number of secondary users (SUs) referred to as CRDs. SN is responsible for populating the available spectrum opportunities (SOPs) within its geographical coverage area and to keep track of the current network hosts and associated operational conditions. A knowledge database is maintained at every SN to represent all relevant CRN information [3].

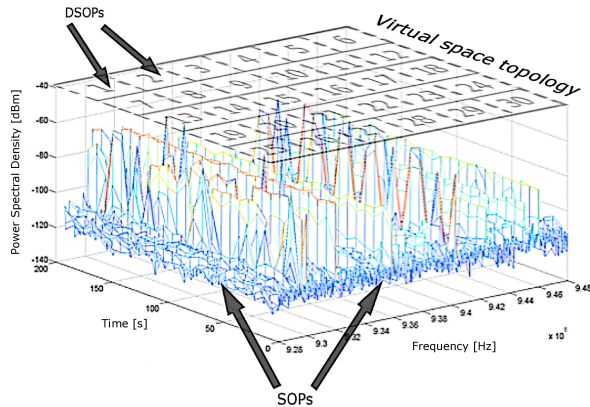


Figure 1. SN imposed structure with regard to available DSOPs

Data representation is based on the geometric structure of a data-cube composed from several 2-dimensional $[0, 1] \times [0, 1]$ cartesian coordinate spaces (CCS). Each CCS represents a different CR-dimension, where all dimensions are functions of time. Ultimately, data representation allows for the operational parameters of each network member device to be described in every CR-dimension, e.g., frequency (depicted as device spectrum opportunities (DSOPs), given that every SU needs a portion of an available spectrum opportunity (SOP) to operate in, regardless of the used modulation [5]), power (transmit output power) and geographic point mapped to a set of virtual space coordinates (x,y) referred to as virtual identifier (VID) Φ . This is selected with regard to the actual geographical location of the hosts in the physical CRN coverage area. Every mapped device is represented in the data-cube with a geometric portion of the space dimension CCS called zone, that surrounds VID Φ . Signaling is exchanged with the deserving SN via a common control channel (CCC) and allows hosts to receive information about mapped adjacent network members (neighbors) [3]. Two network members are considered neighbors if the coordinate span of their geometric zones overlap along one dimension and abut along the other dimension. Simply put the SN imposes a particular geometric structure for its CRN coverage area to create a structured CRN topology (figure 1).

Network members with geographical coordinates located close together access resources like SOPs in the same area of the CRN. Thus, their VIDs in the space dimension representation are also mapped in the proximity of each other. Subsequently, such devices may become neighbors in the virtual CRN topology [3]. A newly arriving CRD that wishes to join a CRN must therefore first contact the responsible SN to request a virtual operational zone (if available) within the imposed topology structure. Naturally, this requires the presence of a free DSOP within an available SOP in the particular SN coverage area [5]. Typically, SOPs are partitioned in three categories with reference to the activity and holding times of the licensed users: Static, Dynamic

and Opportunistic [2], [6]. However, since the holding time of an e2e path must be long enough to allow for it to be traversed [5], only static/dynamic SOPs are considered in the suggested CRN communication framework. Meaning, the disruptive nature of opportunistic SOPs is currently disregarded. If no static/dynamic SOPs are available or predicted to be available in the CRN coverage area, the SN rejects the joining of new CRDs.

Four fundamental operations need to be controlled in the management of a CRN: spectrum sensing, spectrum decision, spectrum sharing and spectrum mobility. Another operation is space mobility. Subsequently, ad-hoc algorithms can be combined with spectrum mobility algorithms to provide an e2e solution for communication. Collaboration between CRDs and the SN is therefore vital and provides us with the necessary operations to communicate within CRNs. To detect possible CRN conflicts such as exploited SOPs being taken over by primary users (PUs), all network members are assumed to perform spectrum sensing [3]. CRDs perform spectrum sensing for their respective channel assignments whereas SN performs global spectrum sensing of the entire CRN coverage area. All collected information is gathered at the SN, which enables the detection of inconsistencies and centralized decision making.

III. NETWORK MODEL

The suggested CRN communication architecture is designed to cover an area with the range of a mobile phone network microcell i.e., with a radius of around 1000 meters [3]. This can obviously be adapted with regard to particular needs and the operating range of individual SNs. CRDs in the CRN may ask to communicate with other CRDs in the own (intra) CRN or other distant (inter) CRNs. The focus in our paper is on intra CRN communication only.

A communication is initiated by a network member conveying a communication request to the local SN. Depending on the type of requested communication, i.e., unoptimized or optimized, SN replies either with a message containing the VID of the destination or with a complete optimized e2e path. The e2e path is computed according to user preferences and the response message from SN has all necessary information to reach the destination [3]. That is to say, intermediate CRDs (all hops) together with channel assignments along the path that are optimized according to, e.g., cost, throughput, delay, security level. An optimized routing path reduces the number of intermediate hops from source to destination by dividing the CRN coverage area into clusters (figure 2). Meaning, CRDs available throughout the geographical area of the particular CRN are grouped together into virtual clusters created at the SN depending on their locations and particular resources accessed, e.g., SOPs. The cluster radius is not fixed and may vary from one CRN to another i.e., the cluster sizes are dimensioned with regard to the specific environmental constraints of the

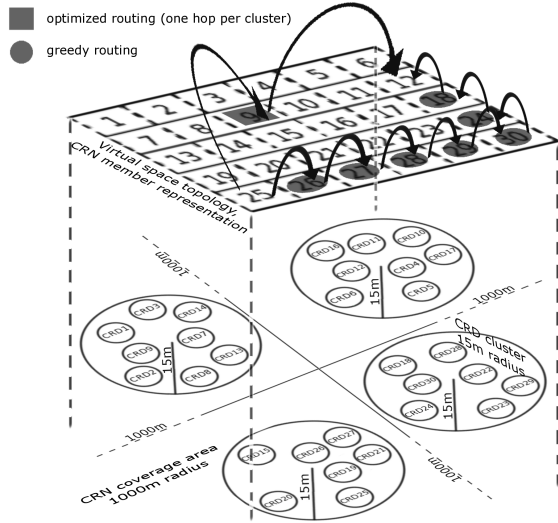


Figure 2. CRN clusters overview, optimized and greedy routing

CRN geographical area. Hence, this requires the cluster sizes of CRNs to be properly dimensioned based on factors like: local, geographical or regulatory considerations, required uplink/downlink data rates by devices in the particular geographical area, rated power output of devices in the CRN, fading characteristics of the channels caused by the obstacles in the coverage area (typically modeled through Rayleigh fading). Ultimately, the dimensioning of the virtual cluster sizes must ensure limited signal degradation during inter and intra cluster communication. A device is considered to belong to a cluster if its geographic coordinates are within the radius of that particular cluster. Given the geographical proximity of CRDs located within the same or adjacent clusters, direct communication (without intermediate relays) between them is considered feasible [3]. Such devices are represented in proximity of each other (neighbors) in the virtual CRN space dimension.

IV. IMPLEMENTATION OF E2E COMMUNICATION

Considering the topology structure imposed by the SN described above, every network member is responsible for a portion of the virtual space representation. Thus, since all network members are initially mapped to specific VIDs and informed about the closest adjacent neighbors by the SN, e2e routing in the CRN can be done in two ways:

- Distributed greedy routing along a line using only information about neighboring nodes and their geometric zones (depicted as greedy routing in figure 2). Routing from source to destination is achieved through a rectilinear distance geometry where a source only requires the VID of the destination from the SN. This entails limited or no QoS guarantees.
- Optimized routing, at the cost of larger overhead and

computational latency (depicted as optimized routing in figure 2). A SN computed and optimized routing path allows for e2e goals of the source to be considered. SN imposes different constraints on the routing path, denoting ad-hoc packet routing with aggregated throughput obtained by employing suitable Multi-Objective, Multi-Constrained optimization algorithms [5].

In the case of distributed greedy routing, the destination is reached by moving from one neighbor to another without any regards to optimization constraints. The algorithm routes along the x and y dimensions towards the destination, by crosschecking the VIDs of neighbors with the VID of the destination. The neighbor with the VID closest to the destination is selected as next hop. Hence, reaching the destination becomes a matter of continuously evaluating the delta value between current devices and the destination, progressing so on the shortest path along the x and y dimensions until the destination is reached (Algorithm 1).

Algorithm 1 Greedy virtual space routing

```

1: procedure GREEDYROUTING( $sd, \Phi$ )
   Route COMMUNICATION message through VSD from  $sd$  towards destination with
   VID  $\Phi$ , via  $cd$ 
2:   if  $\Phi \in sd$  then                                     ▷ Is  $\Phi$  among  $sds$  neighbors  $n$ 
3:      $cd \leftarrow n$                                        ▷ Neighbor  $n$  of  $sd$  is destination
4:   else                                                 ▷  $\Phi$  is not among  $sds$  neighbors
5:     Find device with VID  $\Phi$ 
6:      $cd \leftarrow sd$                                      ▷ Initiate  $cd$ 
7:     while  $\Phi \neq cd$  do                                  ▷ Until VID  $\Phi$  is found
8:       while  $\Phi_x \neq nx$  do                                ▷ Find  $\Phi_x$   $x$  value
9:          $ndv \leftarrow (\Phi_x - nx)$  ▷ Find neighbor of  $cd$  with shortest  $x$ -axis
10:         $ndv$  to VID  $\Phi$ 
11:         $cd \leftarrow n$  ▷ Neighbor  $n$  with shortest  $ndv$  is set as new  $cd$ 
12:      end while
13:      while  $\Phi_y \neq ny$  do                                ▷ Find  $\Phi_y$   $y$  value
14:         $ndv \leftarrow (\Phi_y - ny)$  ▷ Find neighbor of  $cd$  with shortest  $y$ -axis
15:         $ndv$  to VID  $\Phi$ 
16:         $cd \leftarrow n$  ▷ Neighbor  $n$  with shortest  $ndv$  is set as new  $cd$ 
17:      end while
18:    end if
19:    end if
20:    VID  $\Phi$  found i.e., destination is  $cd$ 
21:    return  $cd$ 
22: end procedure

```

where: virtual space dimension (VSD); virtual identifier (VID); current device (cd); source device (sd); neighbor delta value (ndv); Φ_x, Φ_y are x and y axis virtual identifiers for destination point Φ ; nx, ny are x and y axis virtual identifiers for neighbor n of cd

In the case of optimized routing the destination is reached by considering the optimization preferences of the source and accordingly leaping from one cluster to another until the destination is reached. For optimization reasons virtual clusters are created by the SN, which groups together network members depending on location and accessed resources in the CRN coverage area. In other words, the CRN coverage area is subdivided into clusters where each cluster comprises a number of member devices. Given the initial required dimensioning of cluster sizes with regard to the environmental constraints of the CRN geographical area, direct communication between devices in neighboring clusters is considered, excluding so the need of intermediate relays. Hence, an e2e

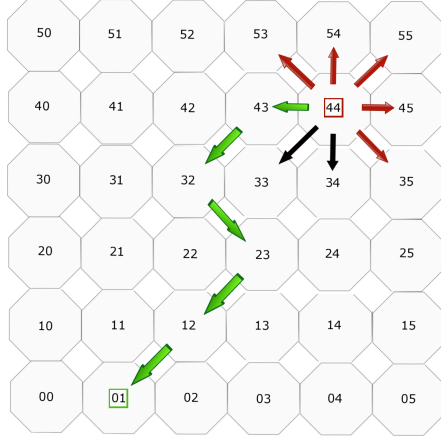


Figure 3. CRN overview depicting CRD clusters, optimized routing

path can be computed where one hop per cluster is sufficient and each hop is determined according to optimization preferences, like throughput, cost, and delay [5]. The optimization problem of selecting an e2e routing path subject to multiple constraints can be of type Multi-Constrained Path (MCP) or Multi-Constrained Optimal Path (MCOP), where popular routing algorithms are Self-Adaptive Multiple Constraints Routing Algorithms (SAMCRA) and Particle Swarm Optimization [7], [8]. Albeit, the optimal solutions for finding a route subject to constraints on two or more additive and multiplicative metrics have been mathematically proven to be NP-complete [9]. This means that the time required to exactly solve the MC(O)P problem can not, in the worst case, be upper-bounded by a polynomial function. Thus, heuristics or approximation algorithms must be used, which have a better chance of running in polynomial time. Some of the main approaches used to solve problems of type MCP are Bandwidth Restricted Path (BRP), Restricted Shortest Path (RSP) and Metrics Combination (MC) [7]. BRP uses the pruning of the links in the graph not satisfying the particular constraints, also known as topology filtering. RSP is a simplification of the original MCP problem, for the particular case of two additive metrics. The policy in this case is that all paths that satisfy the constraint associated with one metric are computed and the best path with reference to the second metric is selected. Finally, MC is about combining a set of QoS metrics into a single metric, and using then well-known policies like Bellman-Ford or Dijkstra for path computation. The optimized e2e path computation, depicted in figure 3 and described in Algorithm 2, employs the constraints of shortest path and least cost, although other constraints can be used as well [5].

An example of optimized routing according to Algorithm 2 is presented in figure 3. A total number of 36 virtual clusters have been considered in our simulation model of the CRN coverage area where communications among their

Algorithm 2 Optimized routing

```

1: procedure OPTIMIZEDCLUSTERROUTING( $sd, \Phi$ )
   Route COMMUNICATION message through VSD from  $sd$  in  $sc$  towards destination
   with VID  $\Phi$ , via device  $d$  in  $pc$ 
2:    $pc \leftarrow sc$  ▷ Initiate  $pc$ 
3:   if  $\Phi \in sc$  then ▷ Is  $\Phi$  among  $cd$  of  $sc$ 
4:      $d \leftarrow lcd$  ▷ Destination is  $lcd$  in  $sc$ 
5:   else ▷  $\Phi$  is not in  $sc$ 
6:     Find device with VID  $\Phi$ 
7:     while  $\Phi \neq pc$  do ▷ Until VID  $\Phi$  cluster is found
8:        $pcdv \leftarrow \sqrt{(\Phi x^2 - px^2) + (\Phi y^2 - py^2)}$  ▷  $pc$  distance to
destination
9:     while  $nc \neq 0$  do ▷ Find  $nc$  closer than  $pc$ 
10:       $ncdv \leftarrow \sqrt{(\Phi x^2 - ncx^2) + (\Phi y^2 - ncy^2)}$  ▷  $nc$ 
distance to destination
11:    if  $ncdv < pcdv$  then
12:      If delta value of current  $nc$  is smaller
13:      while  $cd \neq 0$  do ▷ Find  $lcd$  in  $nc$ 
14:         $lcd \leftarrow cd$  ▷ Save found  $lcd$ 
15:         $pc \leftarrow nc$  ▷  $nc$  with  $lcd$  is new  $pc$ 
16:      end while
17:    end if
18:  end while
19:  end if
20:  end while
21:   $d \leftarrow lcd$  ▷  $lcd$  is selected as next hop device  $d$ 
22:  end while
23:  VID  $\Phi$  found i.e., destination is device  $d$  in  $pc$ 
24:  end if
25:  return  $d$ 
26: end procedure

```

where: virtual space dimension (VSD); virtual identifier (VID); source device (sd); source cluster (sc); device (d); present cluster (pc); cluster devices (cd); present cluster delta value ($pcdv$); neighbor cluster delta value ($ncdv$); neighbor cluster (nc); least cost device (lcd); $\Phi x, \Phi y$ are x and y axis virtual identifiers for destination point Φ ; px, py and ncx, ncy are x and y axis virtual identifiers for pc and nc respectively

hosts are carried out. Prior to executing the first hop, the delta value to the destination is computed for the source cluster and all its neighboring clusters. Given a shortest path constraint, only the clusters with a delta value smaller than the source cluster itself (depicted with black and green arrows in figure 3) are considered to be a viable next hop. Whereas, the clusters with a larger delta value than the source are depicted with red arrows in figure 3 and considered unviable next hops. To select one of the three valid clusters (green arrow) as next hop, the second (least cost) constraint is considered. Meaning, the device offering the least cost regardless of cluster affiliation is selected as next hop, i.e., the optimal shortest path is achieved with regard to both constraints. The process is then repeated (green arrows) for every hop along the path until the destination is reached.

V. SIMULATION RESULTS

To simulate the characteristics of the particular network structure, a topology construction algorithm has been implemented under the C++ object-oriented programming language [3]. The virtual space representation maintained at the SN is set to operate at the application layer and it is initially constructed as a 2-dimensional $[0, 1] \times [0, 1]$ CCS, where every network member is responsible for a small portion (zone) of the CCS. Thus, the network topology is created by populating the virtual space with a varying number of simulated devices. Two network members are

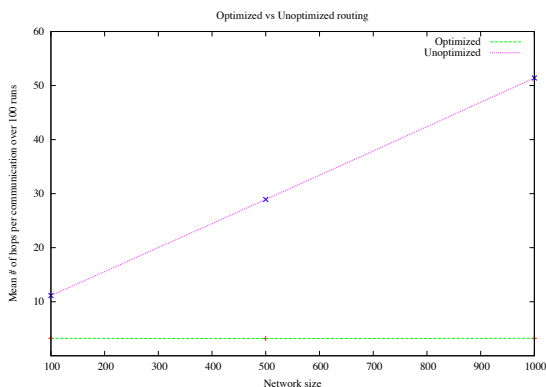


Figure 4. Mean number of hops per communication over 100 runs

considered to be neighbors if the coordinate span of their geometric zones overlap along one dimension and abut along the other dimension. As the focus in our paper is on developing routing algorithms for the advanced network structure, virtual devices are therefore sufficient to test the efficiency of the described routing algorithms. To evaluate the routing performance of the suggested algorithms, a number of experiments have been conducted. Each experiment is initiated by populating the virtual space topology with 100, 500 and 1000 virtual devices. Further, 10 communications between different network members are simulated per experiment according to each respective routing algorithm, i.e., greedy and optimized. Every experiment is repeated 100 times while the mean number of hops (path length) per communication over all simulated runs is computed according to equation 1. Given that 10 communications are simulated for every run the total number of performed communications over 100 runs is $100 \times 10 = 1000$

$$H_n = \frac{1}{1000} \sum_{r=1}^{1000} h_r \quad (1)$$

H_n is the mean path length per communication over all simulated runs and it is computed for every simulated network size n , i.e., 100, 500 and 1000. h_r indicates the total number of resulting hops for 10 communications between randomly selected member devices during simulation run r .

Figure 4 demonstrates the linearity of the mean path lengths per communication, for optimized respective unoptimized communications over the 100 simulated runs and for different network sizes. The actual statistics for each network size are reported in table I with the corresponding confidence intervals (CI). This clearly indicates the advantage of employing a virtual cluster approach to optimize the e2e routing paths between network members. While, the path lengths grow with the network size for the

unoptimized greedy routing solution. Typically, the routing tables used for the particular network structure grow faster than $\log(n)$, meaning faster than the network size. The theoretical increase in this case is $\Theta(dn^{1/d})$, where d is the number of dimensions and n is the network size. Although, in reality this is a low estimate due to the creation of routing loops, which may add additional hops to the actual path length. Routing loops are often noticeable in large networks and occur due to inconsistent routing tables formed by node or link failures. This is a particular common problem in wireless ad-hoc networks with moving hosts, which employ a distributed hop-by-hop routing model (greedy routing) without a comprehensive network overview [10]. Simply put, route discovery operations from source to destination may fail and lead to data packets being sent in an endless loop. Subsequently, this forces a backtracking to find another valid path to the destination. In table I we can see clear indications of routing loops for the unoptimized case and growing network size by considering the 95% CI.

The positive CI growing trend for increasing network sizes and unoptimized routing means that a larger variance in the data set (mean path length per communication) is experienced, which can only be accounted to routing loops. This is more easily observed by viewing the actual mean number of hops for every set of 10 communications conducted under each simulation run. Hence, for unoptimized routing the Bezier curve [11] chases the variance of the data, i.e., approximation of the data trend (figure 5), clearly indicating that the larger the network size grows the more the variance increases. In other words, finding a particular destination in a network with a distributed hop-by-hop routing model becomes increasingly difficult the more crowded the network gets. Thus, this may lead to routing loops and unnecessary large routing paths, which results in visible oscillations for growing network populations (figure 5).

Network size	Unoptimized	CI in %	Optimized	CI in %
100	11.14	± 6.1	3.23	± 3.2
500	28.93	± 6.7	3.19	± 3.3
1000	51.40	± 8.5	3.24	± 3.2

Table I
MEAN PATH LENGTHS PER COMMUNICATION

On the other hand, in table I it is also clear that a CRN with an imposed network structure described above (which uses the suggested centralized path optimization according to virtual clusters), is no longer suffering from growing routing tables with the network size. This can be observed by the almost constant mean path lengths for optimized communication, regardless of network size (figure 6). Albeit, this comes at the cost of computational latency (given the time it takes for a SN to compute an e2e path according to user constraints) and larger overheads (provided that all hops and channel assignments along the e2e path from source

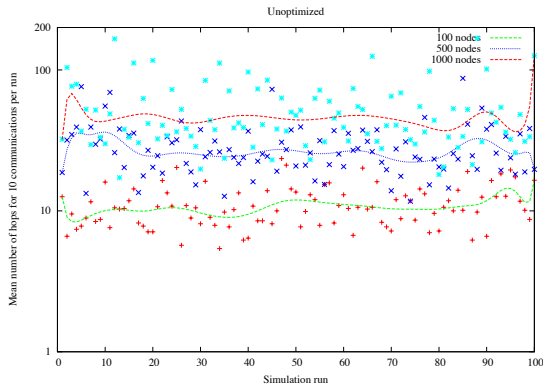


Figure 5. Mean path length per run, unoptimized routing

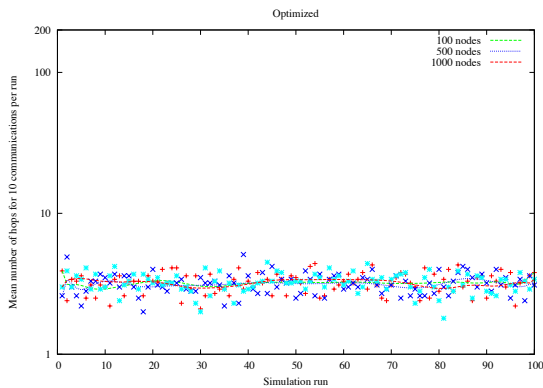


Figure 6. Mean path length per run, optimized routing

to destination must be included in every send packet) [3]. However, through virtual cluster path optimization the CI in table I remain almost constant, regardless of the network size. The main reason for this is that the variance of the path lengths does not increase with network size, which can be observed in figure 6. Complete routing paths containing every hop along the way are in this case computed by the SN with a complete network overview, i.e., routing loops are eliminated. Meaning the advanced solution is suitable for scenarios where resources must be efficiently used, like wireless ad-hoc networks and multihop CRNs.

VI. CONCLUSIONS

By combining ad-hoc and spectrum mobility algorithms a new approach to end-to-end communication in cognitive radio networks has been suggested. With regard to optimizing communication between network members, an efficient routing solution is necessary. Hence, new data

representation and addressing elements are adopted to impose a particular network structure for the operational area of a cognitive radio network. The concept is based on a new architectural approach for the management of cognitive radio networks, designed and developed at the application layer. Additional functionality is provided by a centralized entity called support node. The reported simulation results indicate the feasibility of the suggested routing solutions. Future work entails further development and implementation of the management architecture with additional parameters like, e.g., node mobility, channel availability models, system model for support node access by network members. The simulation studies will also be validated by analytical studies and expanded for additional network scenarios.

REFERENCES

- [1] EU, “Seventh framework programme (fp7)”, 2012, http://cordis.europa.eu/fp7/home_en.html.
- [2] I.A. Akyildiz, L. Won-Yeol, and K. Chowdhury, “Crahn’s: Cognitive radio ad hoc networks”, *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.
- [3] A.O. Popescu, *Cognitive Radio Networks: Elements and Architectures, PhD thesis in preparation*, PhD thesis, School of Computing, Blekinge Institute of Technology, Karlskrona, Sweden., 2013.
- [4] R.W. Thomas, L.A. DaSilva, and A.B. MacKenzie, “Cognitive networks”, in *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks, DySPAN 2005*, November 2005, pp. 352–360.
- [5] A.O. Popescu, D. Erman, M. Fiedler, and A.P. Popescu, “On routing in cognitive radio networks”, in *9th International Conference on Communications, COMM 2012, Bucharest, Romania*, June 21–23 2012.
- [6] H. Khalife, N. Malouch, and S. Fdida, “Multihop cognitive radio networks: to route or not to route”, *IEEE Network*, vol. 23, no. 4, pp. 20–25, July–August 2009.
- [7] X. Masip-Bruin, M. Yannuzzi, J. Domingo-Pascual, A. Fonte, M. Curado, E. Monteiro, F. Kuipers, P. Van Mieghem, S. Avallone, G. Ventre, P. Aranda-Gutiérrez, M. Hollick, R. Steinmetz, L. Iannone, and K. Salamatian, “Research challenges in qos routing”, *Computer Communications*, vol. 29, no. 5, pp. 563–581, Mar. 2006.
- [8] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, 2006.
- [9] W. Zheng and J. Crowcroft, “Quality-of-service routing for supporting multimedia applications”, *Selected Areas in Communications, IEEE Journal on*, vol. 14, no. 7, pp. 1228–1234, sep 1996.
- [10] M.K. Marina and S.R. Das, “Ad hoc on-demand multipath distance vector routing”, *SIGMOBILE Mobile Computing and Communications Review*, vol. 6, no. 3, pp. 92–93, June 2002.
- [11] A. Watt and M. Watt, *Advanced Animation and Rendering Techniques: Theory and Practice*, Addison-Wesley Professional, 1992.