Brief Announcement: Synchronous Distributed Algorithms for Node Discovery and Configuration in Multi-channel Cognitive Radio Networks

Srinivasan Krishnamurthy, R. Chandrasekaran, Neeraj Mittal, and S. Venkatesan

School of Engr. and Comp. Science, The University of Texas at Dallas, Richardson TX, USA

Introduction and Related Work. Cognitive Radios (CR) [4] enable flexible and improved radio spectrum utilization by allowing a group of CR nodes to utilize unused channels without interference with the channels' owners [1]. A CR node has a set of available (wireless) channels and this set varies with time, location and activities of the primary (licensed) users. Determining a node's neighborhood and common channels for communication is non-trivial. We propose several efficient distributed algorithms for *node discovery and configuration* (*NDC*) for various CR node models. Krishnamurthy et al. [3] proposed a solution for *NDC* without a control channel. In their model, each node has a single transceiver. If N is the maximum number of nodes and M is the maximum number of channels, their algorithm uses $2NM$ timeslots for *NDC*. Other works assume a common control channel, or every node is equipped with a separate radio interface for each channel or an identical frequency distribution at each node. Please see [2] for a thorough discussion of related work. Here, we extend the algorithms of [3] to other CR node models.

System Model. We assume that during *NDC*, the CR network topology is a static *multi-hop multi-channel* wireless network. Each node i has a unique id $\mathcal{UID}_i$ in the range $[1 \dots N]$, where N is an upper bound on the total number of nodes. We assume $\mathcal{UID}_i = i$. All nodes know N. Let $\mathcal{A}_{univ} = \{c_1, c_2 \dots c_M\}$ represent the universal set of available channels. All nodes know M and $\mathcal{A}_{univ}$. Every channel has a unique id and all nodes know the mapping between the channel id and its frequency *a priori*. Node i is aware of its channel availability set $\mathcal{A}_i$ and it can receive and transmit on any channel of $\mathcal{A}_i$. Communication is loss-free. Nodes i and j are said to be *neighbors* if i and j are within each other's radio range and $\mathcal{A}_i \cap \mathcal{A}_j \neq \emptyset$. Communication between non-neighbors is by multi-hop transmissions. The set of channels that is common to i and nodes that are within k hops from i is referred to as the *k-local channel set*, $\mathcal{L}_i^k$. Node i needs to know $\mathcal{L}_i^2$ for collision-free communication with its neighbors.

Nodes invoke *NDC* every T time units to account for changes in network topology and/or channel availability sets. The starting times of *NDC* are known to all. The clocks of the nodes are synchronized. *NDC* is said to be complete when each node i determines its set of neighbors and its 2-hop local channel set $\mathcal{L}_i^2$. Time is divided into *timeslots* of equal duration. A message transmitted by a node is *delivered* to all its neighbors in the same timeslot. A receiver successfully *receives* a message if and only if there is exactly one message being delivered at that timeslot on the channel it is tuned to. If two or more neighbors of a node i transmit on the same channel in a given timeslot, a *collision* occurs and i does not receive any of those transmitted messages. Nodes cannot distinguish between a collision and background noise.

S. Dolev (Ed.): DISC 2006, LNCS 4167, pp. 572–574, 2006.

Our Contributions. *NDC* consists of two rounds of identical duration. A *round* is the number of timeslots required for each node to communicate with all of its neighbors. Each timeslot is assigned to one or more node-channel pairs depending on the model. When a node-channel pair (i, c) is assigned to a timeslot t, node i transmits on channel c during timeslot t only if $c \in \mathcal{A}_i$; i remains silent otherwise. During round 1, each node i communicates its channel availability set $\mathcal{A}_i$ and receives $\mathcal{A}_j$ from each neighbor j. Thus, node i finds its neighbors $\mathcal{NBR}_i$ and their respective channel availability sets by the end of round 1. Node i then computes $\mathcal{L}_i^1$. In round 2, i sends $\mathcal{L}_i^1$ to its neighbors.

Limited Channel Divergence with Single Transceiver: Suppose that channel availability across nodes does not vary drastically. Such a situation may arise in scenarios where the CR network is deployed in regions that have a scarce population of primary users and/or other sources of interference. We assume that the sets of channels available to any pair of neighboring nodes i and j differ by at most k i.e. $|\mathcal{A}_i - \mathcal{A}_j| \leq k$, and that the nodes know k. Since $\mathcal{A}_i$ and $\mathcal{A}_j$ differ by at most k channels, if i transmits on any subset of $(k+1)$ channels, there is at least one channel on which j can receive the message *provided* j tunes to the correct channel(s) in the appropriate timeslot(s). Let $\mathcal{A}_i^{k+1}$ be the set of $(k+1)$ lower-most channels in $\mathcal{A}_i$. Let $c_x \in \mathcal{A}_i^{k+1}$. In our algorithm, i transmits on c_x for $(k+1)$ timeslots while the other nodes listen on their respective $(k+1)$ lower-most channels, one channel at a time. Node i repeats the transmission behavior for every channel in $\mathcal{A}_i^{k+1}$; the remaining nodes listen on their $(k+1)$ lower-most channels as before. Thus, each node transmits for $(k+1)^2$ timeslots and a round of this algorithm consists of $\left(N \times (k+1)^2\right)$ timeslots. For $k \ll M$, this is a drastic improvement over the NM timeslots required for *NDC* in [3].

Nodes with Multiple Receivers: Multiple transmissions may be supported by increasing the number of receivers at each node. Each node has r receivers ($r > 0$) and a single transmitter. At any given time, a node can be (i) transmitting on one channel, (ii) receiving on up to r channels or (iii) turned off. Simultaneous transmission and reception (even on different channels) at a node is not allowed. Collisions in a timeslot t are avoided by assigning a unique channel for each node that is transmitting in t. Note that a node cannot receive when it is transmitting. As a result, if nodes i and j are assigned to transmit simultaneously on channels c_i and c_j in a timeslot t, then there must be at least two other timeslots t_i and t_j for the nodes to communicate with each other. We use $\oplus_b$ to denote the modulo b addition: $x \oplus_b y = (x + y - 1) \mod b + 1$. Therefore, given the sequence of numbers 1, 2, $\cdots$, b, the subsequence of j consecutive numbers starting from number i (with wrap-around) is given by $i, i \oplus_b 1, \cdots, i \oplus_b (j-1)$.

Let $G = \{1, 2, \cdots, N\}$ be the set of nodes. We assume that r divides both N and M. G is partitioned into $\frac{N}{r}$ groups $G_1, G_2, \cdots, G_{\frac{N}{r}}$ each of size r. Let $G_1 = \{1, 2, \cdots, r\}$, $G_2 = \{r+1, r+2, \cdots, 2r\}$, and so on be the groups. Each round of the algorithm consists of two sub-rounds. The first sub-round handles inter-group communications among nodes and consists of $\frac{N}{r}$ blocks. Each block consists of M timeslots. In block i, nodes in group G_i transmit and all other nodes listen. Specifically, in timeslot j of block i, r nodes in G_i, given by $(i-1)*r+1$, $(i-1)*r+2, \cdots, i*r$, transmit on r channels c_j, $c_{j \oplus_M 1}, \cdots, c_{j \oplus_M (r-1)}$, respectively. The remaining nodes receive on these r channels simultaneously. The second sub-round employs a divide-and-conquer approach to achieve intra-group communication. For the sake of brevity, we will only describe the second sub-round for the special case of $M = N = x$, such that each node has at least x receivers and $x = 2y$ for some integer y. The algorithm for this special case is used as a subroutine to achieve inter-group communication for general M, N and r in $O(\max(N, M) \log r)$ timeslots. Thus, the time complexity of a round of the algorithm for *NDC* is $O\left(\frac{N \times M}{r} + \max(N, M) \log r\right)$ [2].

Algorithm DAC: Consider a group of nodes $A = \{a_1, a_2, \cdots, a_x\}$ and a group of channels $B = \{b_1, b_2 \cdots, b_x\}$. The algorithm ensures that every node in A listens to every other node in A on all x channels in $O(x \log x)$ timeslots. Let DAC(A_1, B_1) and DAC(A_2, B_2) be two instances of the algorithm DAC. We use DAC(A_1, B_1) $\|$ DAC(A_2, B_2) to denote the algorithm obtained by running the two instances concurrently. To run the two instances concurrently, they should not interfere with each other. Thus, $A_1 \cap A_2 = \emptyset$ and $B_1 \cap B_2 = \emptyset$. Further, we also ensure that $|A_1| = |A_2|$. As a result, the running time of the resulting algorithm is same as that of the either instance. Likewise, we use DAC(A_1, B_1) $\circ$ DAC(A_2, B_2) to denote the algorithm obtained by running the two instances serially, one-by-one.

Let $A = A_1 \cup A_2$ be a partition of A, where $A_1 = \{a_1, a_2 \cdots, a_y\}$ and $A_2 = \{a_{y+1}, a_{y+2}, \cdots, a_{2y}\}$ (recall that $x = 2y$). The algorithm consists of three blocks. In the first block consisting of x timeslots, all nodes in A_1 transmit on all channels in B whereas nodes in A_2 only listen. Specifically, in timeslot i of the first block, nodes a_1, a_2, $\cdots$, a_y transmit on channels b_i, $b_{i \oplus_x 1}$, $\cdots$, $b_{i \oplus_x (y-1)}$, respectively. The second block is similar to the first block except that roles of A_1 and A_2 are reversed. Finally, in the third block, we recursively invoke the algorithm. Let $B = B_1 \cup B_2$ be a partition of B into two equal halves. The third block is given by $\big($DAC(A_1,B_1) $\|$ DAC(A_2,B_2)$\big)$ $\circ$ $\big($DAC(A_1,B_2) $\|$ DAC(A_2,B_1)$\big)$. Let $T(x)$ denote the running time of DAC(A, B). Then, $T(x) = 2x + 2T(x/2)$, which yields $T(x) = O(x \log x)$.

Discussion: Details of the algorithms and the proofs of correctness appear in [2]. In addition, we have also proposed solutions for other CR node models – (i) Limited channel divergence with multiple receivers and (ii) Nodes with multiple transceivers [2]. We have proved that $O(NM)$ timeslots is a lower bound for oblivious deterministic algorithms for *NDC* for the single transceiver model. (The proof is to appear in a future article.) Hence, we believe that the solutions presented here are close to optimal for models considered in this paper. Future work will focus on proving lower bounds and investigating adaptive deterministic algorithms that could potentially be faster than the oblivious algorithms proposed in this paper.

References

[1] R. W. Broderson, A. Wolisz, D. Cabric, S. M. Mishra, and D. Willkomm. *Corvus: A Cognitive Radio Approach for Usage of Virtual Unlicensed Spectrum.*

[2] S. Krishnamurthy, R. Chandrasekaran, S. Venkatesan, and N. Mittal. Algorithms for Node Discovery and Configuration in Cognitive Radio Networks. Technical Report UTDCS-23-06, The Univ. of Texas at Dallas, Richardson, TX, May 2006.

[3] S. Krishnamurthy, M. Thoppian, S. Kuppa, R. Chandrasekaran, S. Venkatesan, N. Mittal, and R. Prakash. Time-efficient Layer-2 Auto-configuration for Cognitive Radios. In *PDCS 2005*, Phoenix, AZ, November 2005.

[4] J. Mitola. *Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio.*