Cognitive Radio Policy-Based Adaptive Blind Rendezvous Protocols for Disaster Response

Saim Ghafoor(✉), Cormac J. Sreenan, and Kenneth N. Brown

Department of Computer Science, University College Cork, Cork, Ireland
saim.ghafoor@insight-centre.org, {cjs,k.brown}@cs.ucc.ie

Abstract. In disaster scenarios, with damaged network infrastructure, cognitive radio (CR) can be used to provide temporary network access in the first few hours. Since spectrum occupancy will be unknown, the radios must rely on spectrum sensing and opportunistic access. An initial goal is to establish rendezvous between CR nodes to set up the network. The unknown primary radio (PR) activity and CR node topology makes this a challenging task. Existing blind rendezvous strategies provide guarantees on time to rendezvous, but assume channels with no PR activity and no external interferers. To handle this problem of blind multi-node rendezvous in the presence of primary users, we propose an Extended Modular Clock Algorithm which abandons the guarantee on time to rendezvous, an information exchange mechanism for the multi-node problem, and various cognitive radio operating policies. We show that the adapted protocols can achieve up to 80% improvement in the expected time to rendezvous and reduce the harmful interference caused to the primary radio.

Keywords: Adaptive radio · Blind rendezvous
Cognitive radio network · Cognitive radio operating policy
Disaster response network · Primary radio activity

1 Introduction

In many disaster scenarios, communication networks are vital for ensuring efficient and effective first response; however, the disaster may have caused significant damage to the existing infrastructure. Cognitive Radio (CR) can provide an effective solution for creating an initial disaster response network until a more permanent network is re-established [1]. The CR can sense what links exist to the remaining infrastructure, sense what spectrum is available, and exploit the spectrum opportunistically while avoiding primary radio activity. Given the nature of the disaster, with unknown PR activity and spectrum spatial diversity, each CR node must sense the spectrum independently rather than using spectrum databases, and must rendezvous with each other in available channels. This creates the challenging problem of efficiently achieving rendezvous in an unknown environment with unknown primary radio activity.

P. Marques et al. (Eds.): CROWNCOM 2017, LNICST 228, pp. 88–99, 2018.
https://doi.org/10.1007/978-3-319-76207-4_8

For two CR nodes, we define rendezvous as the completion of a handshake mechanism between the radios on a single channel. This assumes that the two radios are within transmission range of each other, that they coincide on the channel for a sufficient time period, and that the channel has no detectable primary radio activity or excessive interference for the radios over that time period. When there is no predefined schedule for visiting channels, and no common control channel, this is known as the blind rendezvous problem. There are some sophisticated blind rendezvous protocols, including Modular Clock Algorithm (MCA) [2], Modified MCA (MMCA) [2] and Jump-Stay [3]. These provide guarantees on the time to rendezvous, but they assume a set of channels on which there is no primary radio activity or external interference. Unpredictable arrivals of primary radios on these channels invalidates the guarantee, and may result in the CR nodes causing unacceptable interference to the primary radio. We make two contributions to address this problem. First, we propose an Extended Modular Clock Algorithm (EMCA) which abandons the guarantee, but has a shorter cycle time, and is intended to reduce the average time to rendezvous. Secondly, we explore different operating policies for the CR nodes to handle the behaviour of the primary users, with the aim of reducing harmful interference without adversely affecting time to rendezvous. In addition, we propose an information exchange mechanism for the multi-node problem, to further reduce the time to rendezvous.

We conduct an empirical investigation of these protocols in simulation. We generate randomised but realistic PR activity patterns, and consider the rendezvous problem for different numbers of unsynchronised CR nodes. We measure the average time to rendezvous and the amount of harmful interference experienced by the primary radios. We demonstrate that EMCA with appropriate operating policies can achieve up to 80% improvement in the average time to rendezvous compared to the existing blind rendezvous protocols. We demonstrate that policies which temporarily blacklist channels with detected PR activity are able to reduce incidents of harmful interference caused to the primary users.

To summarise our contributions,

1. we propose an Extended Modular Clock Algorithm (EMCA) to provide better expected time to rendezvous for unknown environments with PR activity;
2. we propose CR operating policies which adapt to PR activities to reduce harmful interference on PR systems, specifically Normal, Reactive with and without timeslot truncation, and Proactive, which attempts to learn and avoid PR activity;
3. we propose a neighbor exchange mechanism to expedite the rendezvous process for multiple CR nodes; and
4. we demonstrate the effectiveness of EMCA and the operating policies on simulated primary radio activity patterns.

2 Related Work

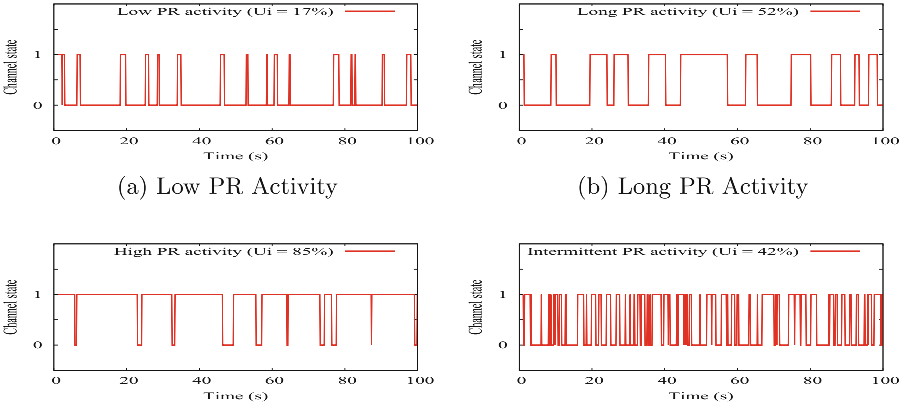
Blind rendezvous strategies (e.g. [2–7]) have gained much attention in CR Adhoc Networks. The modular clock algorithm (MCA) [2] is a blind rendezvous protocol which guarantees rendezvous for radios with identical channel sets. Each radio cycles through its channel set of size m for up to $2P$ slots, where P is the smallest prime $\geq m$, before restarting with a new hopping rate. MMCA [2] caters for different channel sets, with limit $2P^2$. Asynchronous timeslots are shown to be beneficial in reducing TTR in MCA [8]. Jump-Stay (JS) [3] and Extended JS [4], with limits $3P$ and $4P$, is similar to MCA/MMCA, but alternates rounds between hopping and staying on the same channel. All of these protocols assume that the channels are free from primary radio activity. If a primary user appears, the protocols either interfere, or lose their guarantee of rendezvous. To help avoid harmful interference, IEEE 802.22 specifies operating policies for CR deployment and operation [9] for broadband services using TV White Spaces (TVWS). There appear to be no studies analysing rendezvous performance based on PR activity patterns and practical operating policies.

3 System Preliminaries

System Model: We consider an LxL network area, with N nodes. Each node, due to spatial diversity can only access m channels from G randomly, where $G = \{1, 2, 3, \ldots, n\}$. Therefore, common channels among nodes may vary. We assume a connected topology, where all nodes are within range of each other. Each CR is equipped with a single wireless interface. We further assume a time slotted system where timeslot (TS) duration is fixed and known to all users. We assume that nodes are not synchronised with each other. We assume that a CR initially performs sensing for channel accessibility and excludes channels occupied by e.g. emergency services or other prioritised users. Later, it can perform fast sensing for PR detection [9]. For PR detection, we assume an energy detection model and, for identification, a technique such as cyclostationary signatures can be used [10]. We assume that PR traffic is evenly distributed in the space, however our proposed algorithm doesn't depends on the evenly distributed PR traffic.

Primary User Activity Model and Patterns: The performance of cognitive network highly depends on PR activity patterns. PR activity models are widely used to represent a spectrum usage pattern and measurements for performance evaluation [11,12]. We use a popular continuous time alternating ON/OFF Markov Renewal Process to model PR activity [11,13,14]. In this model, the duration of ON/OFF states of a channel i are denoted as T^i_{ON} and T^i_{OFF}. The renewal period $Z_i(t)$ will occur when one ON/OFF period is complete, where, $Z_i(t) = T^i_{ON} + T^i_{OFF}$. We have used the formulation mentioned in [13–15], where the channels ON/OFF periods are both exponentially distributed with p.d.f., $f_X(t) = \lambda_X \times e^{-\lambda_X(t)}$ for ON state and $f_Y(t) = \lambda_Y \times e^{-\lambda_Y(t)}$ for OFF state. The duration of time in which channel i is in the ON state i.e. U^i, is given as:

$$U^i = \frac{E[T^i_{ON}]}{E[T^i_{ON}] + E[T^i_{OFF}]} = \frac{\lambda_Y}{\lambda_X + \lambda_Y} \tag{1}$$



(a) Low PR Activity (b) Long PR Activity

(c) High PR Activity (d) Intermittent PR Activity

Fig. 1. Different PR activity patterns.

Algorithm 1. Function EMCA

Input: t (counter to change r_i), i (node id), T (timeslot), m_i.
Output: c (selected channel by node i).

```
calculate p_i, the prime number greater than or equal to m_i
if T_i < 1 then
    choose initial j_i^t = rand[0, m_i)
    choose r_i from [0, p_i) randomly
end if
if t ≥ p_i then
    choose r_i from [0, p_i) randomly
    t_i = 0
end if
j_i^t = (j_i^t + r_i) mod p_i
if j_i^t < m_i then
    c = c_{i, j_i^t}
else
    c = c_{i, rand([0, m_i))}
end if
    return c;
```

where $E[T_{ON}] = 1/\lambda_{ON}$ and $E[T_{OFF}] = 1/\lambda_{OFF}$ are the means of exponential distributions, and λ_X and λ_Y are the exponential distribution rate parameters. The probability of channel i being in the ON or OFF state at time t can be calculated as below, where $P_{ON}(t) + P_{OFF}(t) = 1$. To illustrate, PR activity patterns are shown in Fig. 1.

$$P_{ON}(t) = \frac{\lambda_Y}{\lambda_X + \lambda_Y} - \frac{\lambda_Y}{\lambda_X + \lambda_Y} e^{-(\lambda_X + \lambda_Y)t} \tag{2}$$

$$P_{OFF}(t) = \frac{\lambda_X}{\lambda_X + \lambda_Y} + \frac{\lambda_Y}{\lambda_X + \lambda_Y} e^{-(\lambda_X + \lambda_Y)t} \tag{3}$$

4 Extended Modular Clock Algorithm (EMCA) with Neighbor Information Exchange Mechanism

EMCA (Algorithm 1) is based on the Modular Arithmetic approach of MCA [2], adapted to account for the effect of PR activity. In EMCA, r is the rate/step

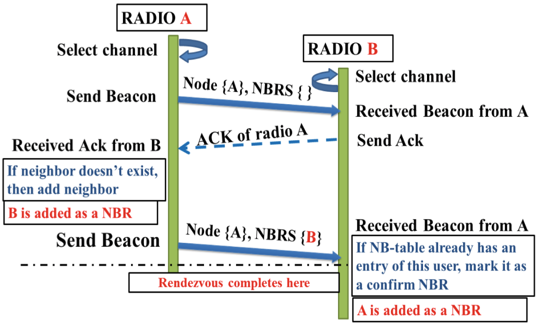


Fig. 2. Neighbor information passing mechanism.

value by which CRs hop the available channel set (ACS), m is the total number of channels, and P is the smallest prime number larger or equal to m. EMCA initializes by choosing an initial index and a rate value randomly. The rate value will remain same for a rendezvous cycle of P timeslots. If rendezvous does not occur within P, then r will be re-selected. At each iteration, the next index value will be calculated using $\mathrm{mod}(P)$. In EMCA, the rendezvous cycle is short (P TS) and unavailable channels are remapped randomly from ACS to avoid biased selection of channels early in the order. In MCA [2] and MMCA [2], the iteration limits are $2P$ and $2P^2$, to ensure rendezvous if different rates are selected, even if the sequence starts are not synchronised. Since we cannot guarantee rendezvous even if two radios are on the same channel in the same slot, because of unknown PR activity, we reduce this limit to allow a search of all channels, but a faster rate re-selection, in the hope of speeding up the time to rendezvous.

For a successful rendezvous, two nodes must complete a handshake process. We propose a beaconing mechanism in which nodes embed into the beacon a list of neighboring nodes they have overheard. As shown in Fig. 2, if two nodes find their own ID in each other's beacons, then we assume rendezvous can be completed. For example, when Node B receives a beacon from A it will send an ACK. A now knows that B has received its beacon, and adds B to its neighbour list. If B receives A's next beacon, it will discover its own ID in the list. It knows that A has received its ACK, and can add A to its neighbour list.

5 Cognitive Radio Operating Policies

A CR must be able to identify and vacate channels occupied by a primary user, and avoid those channels for some specified time. These restrictions are described in [9] as channel availability check (CAC) and channel non-occupancy period (CNP). CAC is the time during which a channel should be checked for the presence of a PR. CNP is the period during which a CR should avoid transmission on a channel which is already detected as occupied. For TV white spaces (TVWS) [9], which have long activity patterns, the CAC time is by default 30 s,

Algorithm 2. EMCA for Normal Operating Policy

Input: m_i (total number of channels), i (node id).

```
t_i = 0
T_i = 0
while not rendezvous with all nodes do
    c = EMCA(i, t_i, T_i, m_i)
    condition = false
    beacon = 0
    if channel c is in BLC list then
        Do nothing
    else if channel c is occupied then
        Add c in BLC_i
    else
        while beacon ≠ 5 & condition ≠ true do
            if channel c is occupied then
                Add c in BLC_i
                condition = true
            else
                Attempt rendezvous on c
            end if
        end while
    end if
    wait for timeslot to end
    t_i = t_i + 1
    T_i = T_i + 1
end while
```

and CNP is a minimum of 10 min. In order to respect the specification, we have proposed different CR operating policies, which work with the rendezvous strategies and are adaptable in response to PR activity. The intention is to reduce harmful interference. Specifically, each channel will be checked for PR activity at the time of selection and before each beacon transmission, and will be blacklisted (BL) if found to be occupied; BL channels will be kept in a blacklisted channel list (BLC) and will not be used for transmission or PR detection until its CNP expires. The proposed policies are described below, together with a simple "Listen before Talk" policy for comparison.

Listen Before Talk (LBT) or No Policy: In LBT, a channel will be checked before every transmission, and will not be used if PR activity is detected. However, such channels are not blacklisted, and rendezvous attempts will continue at the next scheduled beacon transmission.

Normal Operating Policy (Norm): At the start of a timeslot, the selected channel will be checked for PR activity. If PR is detected, the whole timeslot will be abandoned; otherwise, the beacon transmission phase will start with LBT. If before any beacon transmission a PR is detected the radio will not transmit for the rest of the timeslot. If PR activity is detected, the channel will be moved to BLC, and remain there until its CNP expires. At the start of the next timeslot, a new channel will be selected. The policy is shown in Algorithm 2.

Reactive Operating Policy: The Normal policy wastes time by staying silent on the current channel. To avoid this, the Reactive policy immediately continues hopping through the channels using its existing channel selection algorithm. CR operating limitations are as before, where LBT is followed with CAC/CNP checks. There are two variations, depending on whether or not the timeslot is truncated on PR activity detection. Maintaining the timeslot structure keeps any time synchronisation between nodes, while starting a new timeslot means that a node will reach the P limit faster (in real time), and so if needed can change its rate more quickly.

Algorithm 3. EMCA for Reactive Operating Policy

Input: m_i (total number of channels), i (node id).

```
t_i = 0
T_i = 0
while not rendezvous with all nodes do
    c = EMCA(i, t_i, T_i, m_i)
    channel_unoccupied = false
    channel_occupied = false
    beacon_sent = 0
    channel_select = 0
    procedure Channel Selection
        if c is in BLC_i then
            while channel_select ≠ m_i & channel_unoccupied ≠ true do
                c = EMCA(i, t_i, T_i, m_i)
                if c is in BLC_i then
                    Do nothing
                else if c is occupied then
                    Add c in BLC_i
                else
                    channel c is unoccupied
                    channel_unoccupied = true
                    selected_channel = c
                end if
            end while
        else if c is occupied then
            while channel_select ≠ m_i & channel_unoccupied ≠ true do
                c = EMCA(i, t_i, T_i, m_i)
                if c is in BLC_i then
                    Do nothing
                else if c is occupied then
                    Add c in BLC_i
                else
                    channel c is unoccupied
                    channel_unoccupied = true
                    selected_channel = c
                end if
            end while
        else
            channel c is idle
            selected_channel = c
        end if
    end procedure
    procedure Beacon Transmission
        while beacon_sent ≠ 5 do & channel_occupied ≠ true
            if selected_channel is occupied then
                Add c in BLC_i
                channel_occupied = true
            else
                selected_channel is unoccupied
                Attempt rendezvous on selected_channel
            end if
        end while
    end procedure
    wait for timeslot to end
    t_i = t_i + 1
    T_i = T_i + 1
end while
```

Without Timeslot Truncation (RwoT): The node will search for a free channel until one is found or all channels are examined. If no free channel is found, the node will remain quiet until the end of the timeslot. A new rate and index will be selected when node completes a full round (Algorithm 3).

With Timeslot Truncation (RwT): Each time a node selects a new channel, the timeslot number will also increase. Algorithm 3 is also applicable for RwT, but where the TS increment occurs with every channel selection.

Proactive Operating Policy (Pro): The Proactive policy attempts to learn the behaviour of the primary users, going beyond the use of the blacklist. For each channel, it maintains a channel weight C_w^i, which approximates the channel's probability of being unoccupied (or OFF), as shown in Eq. 4. Channel state matching is defined as positive successful match (PSM) (Estimated State (ES) = 0, Observed State (OS) = 0), negative successful match (NSM) (ES = 1, OS = 1), false alarm (FA) (ES = 1, OS = 0) and miss detection (MD) (ES = 0, OS = 1). MD occurs when a node declares an occupied channel as unoccupied and FA occurs when node declares an unoccupied channel as occupied. Using the C_w^i values, each node then maintains a sorted Weighted Channels list (WCL).

$$C_w^i(weight) = \frac{(P_{PSM} + P_{FA})}{(P_{PSM} + P_{NSM} + P_{FA} + P_{MD})} \tag{4}$$

Algorithm 4. EMCA for Proactive Operating Policy

Input: m_i (total number of channels), i (node id).

```
t_i = 0
T_i = 0
while not rendezvous with all nodes do
    c = EMCA(i, t_i, T_i, m_i)
    update channel weight
    channel_unoccupied = false
    channel_occupied = false
    beacon_sent = 0
    channel_select = 0
    procedure CHANNEL SELECTION
        if c is in BLC_i then
            while channel_select ≠ m_i & channel_unoccupied ≠ true do
                select channel from WCL_i
                update channel weight
                if c is in BLC_i then
                    Do nothing
                else if c is occupied then
                    Add c in BLC_i
                else
                    channel c is unoccupied
                    channel_unoccupied = true
                    selected_channel = c
                end if
            end while
        else if c is occupied then
            while channel_select ≠ m_i & channel_unoccupied ≠ true do
                select channel from WCL_i
                update channel weight
                if c is in BLC_i then
                    Do nothing
                else if c is occupied then
                    Add c in BLC_i
                else
                    channel c is unoccupied
                    channel_unoccupied = true
                    selected_channel = c
                end if
            end while
        else
            channel c is idle
            selected_channel = c
        end if
    end procedure
    procedure BEACON TRANSMISSION
        Similar as in other policies
    end procedure
    wait for timeslot to end
    t_i = t_i + 1
    T_i = T_i + 1
end while
```

The policy starts by selecting a channel in each timeslot as normal. However, if channel is occupied then WCL will be used to pick another channel in proportion to the weights in WCL. The intention is to augment an existing channel selection algorithm by temporarily returning to channels most likely to be free, rather than staying silent during a slot when PR activity is detected (Algorithm 4).

6 Performance Evaluation

We evaluate the channel selection algorithm and operating policies in simulation, in order to be able to account for the effect of asynchronous cognitive radio nodes, and uncertain primary user activity. We measure both the time to rendezvous and the amount of harmful interference caused to the primary users, and we compare EMCA to MMCA, JS and to a random channel selection. In each case, we apply the different operating policies uniformly to each rendezvous protocol.

Simulation Setup: Our evaluation uses the well known network simulator NS-2, with extensions to the Cognitive Radio Cognitive Network framework [16], notably for PR activity and channel prediction at the MAC layer, and rendezvous strategies and policies at the network layer. The number of CR nodes used are 2 and 10, where each node can access only 7 out of 10 possible channels, selected randomly. Each node starts within a window of one time slot and at a random time. The CNP time is used as 3xTS. Each TS is divided into five equal parts, where beacon transmissions are scheduled randomly within first half of every part, so that each node will have sufficient time for beaconing/listening.

Table 1. Rate parameter values for channel states used in simulation

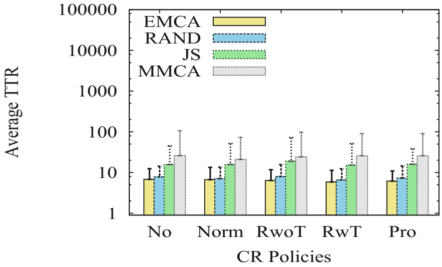
PR activity	Simulation parameters	Channel ids									
		1	2	3	4	5	6	7	8	9	10
High	$\lambda_{\mathbf{X}}$	0.25	0.3	0.25	0.23	0.22	0.25	0.22	0.23	0.32	0.21
	$\lambda_{\mathbf{Y}}$	0.93	1	1.03	1.45	1.10	0.64	1.41	1.59	0.64	1.45
	$\mathbf{U_i}$	0.79	0.77	0.8	0.86	0.84	0.72	0.87	0.87	0.66	0.87
Mix	$\lambda_{\mathbf{X}}$	10000	1.03	0.22	0.22	1.33	10000	1.28	0.23	0.25	1.79
	$\lambda_{\mathbf{Y}}$	0	0.3	0.31	1.2	1.2	0	0.28	0.49	0.93	1.3
	$\mathbf{U_i}$	0	0.23	0.58	0.85	0.47	0	0.18	0.68	0.79	0.42

The Tx range is 250 m for CRs, and network area is $1000 \times 1000\,\text{m}^2$. PR activity patterns are generated using rate parameters λ_X and λ_Y, as shown in Table 1. For Zero PR activity, rate values are used as $\lambda_X = 10000$ and $\lambda_Y = 0$. We consider Zero, High and Mixed PR activity patterns, where in mixed PR activity each channel follows a different traffic pattern, as shown in Fig. 1. For space reasons, we omit Low and Long PR patterns, whose results lie between Zero and High. The metrics used for evaluation are (i) Average TTR (ATTR), which is the time from when the first node starts to the time when last node receive its beacon confirmation and (ii) Harmful Interference (HI), which is the average number of times when interference is caused by a CR towards PR.

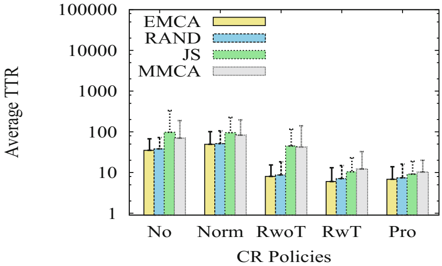
Performance of EMCA with CR Operating Policies: To evaluate the performance of EMCA over different CR operating policies, we vary PR activities with different traffic patterns shown in Fig. 1. We run 100 simulations for each case and take the average of all simulations. Each simulation runs until each node finds every other node in the network. We show the average time to rendezvous for each rendezvous algorithm and operating policy, for each traffic pattern, in Fig. 3 (for 2 nodes) and Fig. 4 (for 10 nodes).

For the zero PR case (Figs. 3a and 4a), EMCA achieves the lowest time to rendezvous, and Random is only marginally slower. As expected, MMCA and JS are significantly slower, because their rendezvous guarantee requires longer cycles before changing the rate. For these zero PR experiments, the operating policies do not apply and so do not affect TTR.

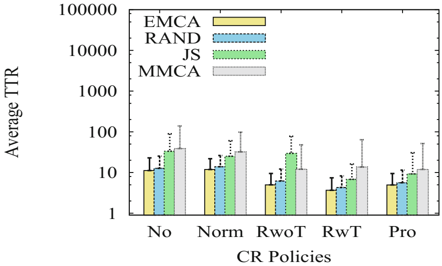
When we introduce PR activity (Figs. 3 and 4(b) and (c)), EMCA and Random still outperform the other two algorithms, which suffer from the longer cycle times even though the rendezvous guarantee no longer applies. EMCA is still the fastest algorithm, with the improvement over JS and MMCA ranging up to one order of magnitude depending on the operating policy. Random is still only marginally worse than EMCA. The impact of the different operating policies is now clearer. The Normal policy is slower than the others, and its TTR increases with higher PR activity, as any detected PR activity causes the nodes to stop transmitting. The reactive and proactive policies show that this time can be used more effectively. RwT shows up to 80% improvement over the Normal policy



(a) Zero PR Activity

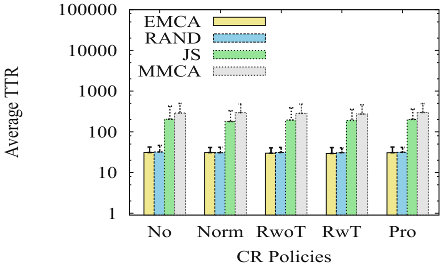


(b) High PR Activity

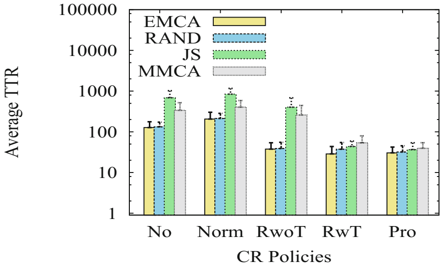


(c) Mix PR Activity

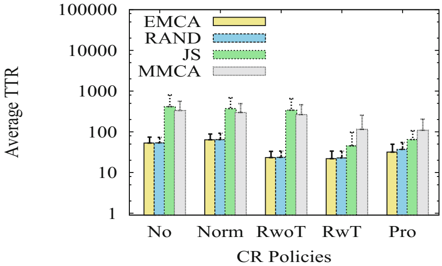
Fig. 3. ATTR for 2 nodes



(a) Zero PR Activity



(b) High PR Activity

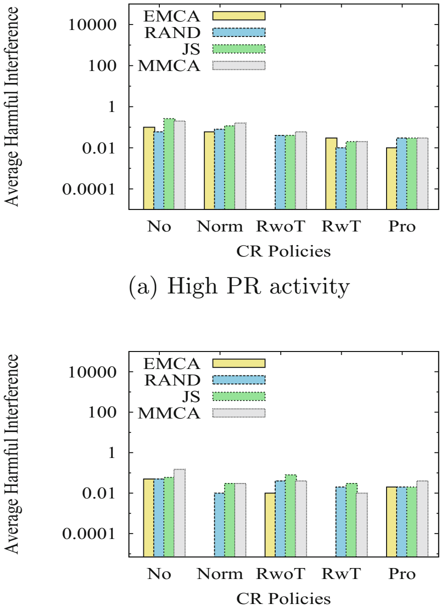


(c) Mix PR Activity

Fig. 4. ATTR for 10 nodes

(Fig. 3b), with Proactive only slightly slower. The Proactive policy also brings the TTR for JS and MMCA down to close to the level of EMCA and Random.

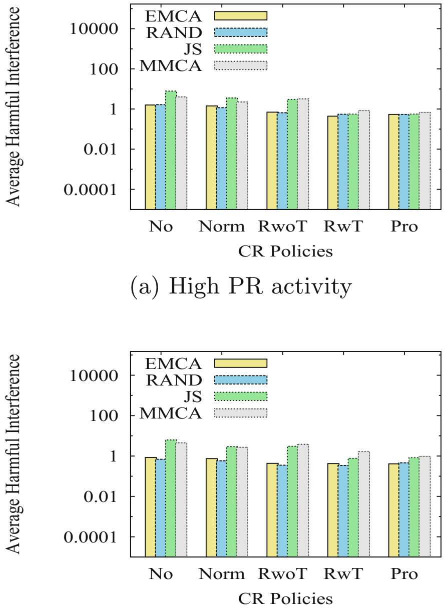
In Figs. 5 and 6, we show the average number of incidents of harmful interference (i.e. when CR transmissions coincide with PR activity) in the same experiments as for Figs. 3 and 4. There is obviously no harmful interference in the zero PR case, and so the graphs are omitted. For the High and Mix PR cases, we again see the benefits of the Reactive and Proactive policies. At some points, the harmful interference is observed as zero even with PR activity for EMCA. In the two node experiments, harmful interference is relatively low, with less than one incident expected per full rendezvous cycle dropping to between 1% and 2% chance of any incident for the reactive and proactive policies. In the



(a) High PR activity

(b) Mix PR Activity

Fig. 5. Harmful interference for 2 nodes



(a) High PR activity

(b) Mix PR Activity

Fig. 6. Harmful interference for 10 nodes

10 node cases, harmful interference is higher, particularly for JS and MMCA because of their higher TTRs, but dropping to below one expected instance per rendezvous cycle for the reactive and proactive policies. Proactive and Reactive, though, are marginally better. Considering both time to rendezvous and harmful interference, the results show that EMCA with the Proactive policy is the preferred configuration.

7 Conclusion

It is widely acknowledged that the flexibility of cognitive radio networks makes them especially suitable for operation in unknown environments, such as disaster response. Blind rendezvous is essential in such situations, but existing techniques make assumptions about primary radio activity and the radio environment. In order to overcome these restrictive assumptions, this paper presented an Extended Modular Clock blind rendezvous protocol, which is an adaptive protocol and can minimize the network setup delay in a disaster situation. Experiments with a variety of primary radio traffic models show up to 80% improvement in the key metric time to rendezvous. Reductions in the effect of harmful interference in comparison with existing rendezvous strategies is also observed empirically. Furthermore, three different operating policies are presented to improve adaptation to primary radio activities. The best policy is

Proactive, which prefers to return to channels with lower previous PR activity. It offers an order of magnitude improvement in time to rendezvous over the basic LBT policy, and improves the performance for all of the studied rendezvous algorithms. This study can help regulatory/standard-bodies and service providers for CR deployment in urban and mission critical areas over different spectrum bands. Future work will focus on developing a more sophisticated learning scheme for Proactive, and on multihop blind rendezvous.

Acknowledgement. This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under the CTVR (SFI 10/CE/I1853), CONNECT (13/RC/2077) and INSIGHT (SFI/12/RC/2289) grants.

References

1. Ghafoor, S., Sutton, P.D., Sreenan, C.J., Brown, K.N.: Cognitive radio for disaster response networks: survey, potential, and challenges. IEEE Wirel. Commun. **21**(5), 70–80 (2014)
2. Theis, N.C., Thomas, R.W., DaSilva, L.A.: Rendezvous for cognitive radios. IEEE Trans. Mob. Comput. **10**(2), 216–227 (2011)
3. Liu, H., Lin, Z., Chu, X., Leung, Y.W.: Jump-stay rendezvous algorithm for CRNs. IEEE Tran. Parallel Distrib. Syst. **23**(10), 1867–1881 (2012)
4. Lin, Z., Liu, H., Chu, X., Leung, Y.W.: Enhanced jump-stay rendezvous algorithm for cognitive radio networks. IEEE Commun. Lett. **17**(9), 1742–1745 (2013)
5. Zhang, Y., Yu, G., Li, Q., Wang, H., Zhu, X., Wang, B.: Channel-hopping-based communication rendezvous in CRNs. IEEE Trans. Netw. **22**(3), 889–902 (2014)
6. Chao, C.M., Fu, H.Y., Zhang, L.R.: A fast rendezvous-guarantee channel hopping protocol for CRNs. IEEE Trans. Veh. Technol. **64**(12), 5804–5816 (2015)
7. Pu, H., Gu, Z., Lin, X., Hua, Q.S., Jin, H.: Dynamic rendezvous algorithms for cognitive radio networks. In: IEEE International Conference on Communications (ICC), pp. 1–6 (2016)
8. Robertson, A., Tran, L., Molnar, J., Fu, E.H.F.: Experimental comparison of blind rendezvous algorithms for tactical networks. In: IEEE WoWMoM, pp. 1–6 (2012)
9. IEEE 802.22.2: Part 22.2: Installation and Deployment of CR Systems (2012)
10. Sutton, P.D., Nolan, K.E., Doyle, L.E.: Cyclostationary signatures in practical cognitive radio applications. J. Sel. Areas Commun. **26**(1), 13–24 (2008)
11. Saleem, Y., Rehmani, M.H.: Primary radio user activity models for cognitive radio networks - a survey. J. Netw. Comput. Appl. **43**, 1–16 (2014)
12. Saleem, Y., Yau, K.A., Mohamad, H., Ramli, N., Rehmani, M.H.: SMART- a SpectruM-Aware clusteR-based rouTing scheme for distributed cognitive radio networks. Elsevier Comput. Netw. **91**, 196–224 (2015)
13. Kim, H., Shin, K.: Efficient discovery of spectrum opportunities with MAC-layer sensing in cognitive radio networks. IEEE Trans. Mob. Comput. **7**(5), 533–545 (2008)
14. Kim, H., Shin, K.: Fast discovery of spectrum opportunities in cognitive radio networks. In: IEEE DySPAN 2008 (2008)
15. Min, A.W., Shin, K.G.: Exploiting multi-channel diversity in spectrum-agile network. In: Proceedings of the INFOCOM (2008)
16. http://faculty.uml.edu/Tricia_Chigan/Research/CRCN_Simulator.htm