

The Impact of Primary User Activity on the Performance of Sequence-Based Rendezvous

Jelena Mišić, Md. Mizanur Rahman, and Vojislav B. Mišić

Ryerson University

Toronto, ON, Canada M5B 2K3

Email: {jmisic, mdmizanur.rahman, vmisic}@ryerson.ca

Abstract—Rendezvous in cognitive networks is often conducted using a predefined or adaptive channel hopping sequence, due to the fact that the time to rendezvous in such cases has a finite upper bound. However, random activity of primary users on any of the channels used can destroy a rendezvous. We show that, in the presence of random primary user activity, the time to rendezvous of sequence-based rendezvous techniques becomes a random variable without a finite upper bound.

Index Terms—opportunistic spectrum access; rendezvous algorithms; sequence-based rendezvous

I. INTRODUCTION

Opportunistic or cognitive spectrum access often relies on channel hopping as the basic mechanism to ensure communication in the presence of unpredictable activity patterns of licensed users [1]. To establish communication with an already operational node in the vicinity, a cognitive node must first undertake the so-called rendezvous procedure in which the nodes try to meet at the same RF channel and exchange synchronization data [11], [7]. A number of approaches to the rendezvous problem have been proposed, with or (preferably) without the aid of a dedicated infrastructure such as a central (base) station or a common control channel [2], [6]. Among the most popular are the protocols in which the channel hopping sequence is constructed so as to obtain a guaranteed finite upper bound for TTR, usually for two nodes but sometimes even for multiple nodes [3], [7], [8], [9], [13], [14], [18], [19].

While the existence of a finite upper bound makes the protocols from the latter group appealing, there are still problems that need to be resolved. First, most sequence-based approaches assume that the rendezvous is accomplished when two nodes ‘hop on a common available channel in the same time slot’ but don’t provide the actual handshaking protocol [13]. Second, some approaches require that nodes’ clocks and, by extension, their sequences, are synchronized [20], even though this requires some form of central authority, in which case it is not blind rendezvous any more. Most importantly, many approaches assume that the channels to hop onto are always available, i.e., free of primary user activity, which is simply unrealistic.

In this paper, we analyze the behavior of sequence-based blind rendezvous protocols under random activity of primary users. We show that the probability of a rendezvous being destroyed by the interference from a primary user transmission is not zero. As the result, the upper bound for TTR in the

two node scenario becomes a random variable without a finite upper bound. While the analysis is performed for two specific sequence-based algorithms, the same conclusion holds for other similar algorithms.

The rest of the paper is organized as follows. In Section II we outline the system model and outline the requirements for a sequence-based rendezvous protocol. The next three Sections use probabilistic analysis to model the activity of primary users and its impact on the rendezvous protocol, including the probability that a rendezvous will be destroyed by the interference from a primary user. Section VI presents the performance evaluation, while Section VII concludes the work.

II. SYNCHRONIZATION IN SEQUENCE-BASED RENDEZVOUS

The basic sequence-based cognitive rendezvous scenario considers two nodes that hop through a common set of available RF channels according to a predefined or pseudo-random sequence, in order to meet at the same channel and exchange synchronization data. Let the duration of the sequence be denoted with Δ , expressed in rendezvous slots or r-slots, which is the predefined time interval δ during which the node remains on a given channel. Of course, the time period δ of the initiator node listen/rendezvous sub-slot must be long enough to allow for successful establishment of the link between the two nodes, even though we do not consider the actual handshaking protocol, more details on which can be found in [18].

Eventually both nodes will hop to the same channel at the same time (or almost the same time) and a rendezvous will occur. The synchronization of channel hopping sequences is, thus, the first level of synchronization; this is the problem that all sequence-based algorithms focus on.

We assume that both nodes are aware of the set of N channels to be used. In the most general case, the nodes hop through (possibly different) subsets $m_i \in N$ and $m_f \in N$ of the channel set; without loss of generality, we may assume that $m_i = m_f = m$. The nodes need not begin to hop at the same time: the first node to begin is called the initiator, the other one the follower. (As in [12], [18], [5], [3], we assume that the nodes are aware of their roles.) The sequence of the follower node will lag the one of the initiator nodes by a time between 0 and $\Delta - 1$ r-slots.

Some of the protocols assume that each node will perform media sensing upon hopping to a new channel, the purpose of

which is to check for the presence of primary user activity on that channel. If such activity is detected, the node will remove that channel from its sequence, recalculate the sequence if necessary, and restart it. Note that the initiator and the follower will update their sequences at different times, as these will typically differ. However, some sequences are still guaranteed to lead to a rendezvous as long as there is at least one idle channel [7], [18].

III. MODELING PRIMARY USER ACTIVITY

As noted above, primary user activity is to be expected on any given channel. For simplicity, we assume that primary users on different channels exhibit independent random activity. Let us assume that T_a and T_i , the durations of active and idle times on a channel, follow random probability distribution with the probability density functions (pdf) $t_a(x)$ and $t_i(x)$, respectively. As active and idle times alternate, the total cycle time on the channel will have the pdf of $t(x) = t_i(x) * t_a(x)$. The average cycle time will be $\overline{T}_{cyc} = \overline{T}_a + \overline{T}_i$.

Major analytical tool which we will use to analyze this problem is renewal theory. A random process which counts the number of cycles of a primary user on a single channel is a renewal process [10]. Beginning of an idle period is therefore a renewal point at which a new probabilistic replica of the original renewal process starts. This is useful because we can calculate all relevant time periods relative to the beginning of the channel idle period.

Let us first consider an arbitrary event on a given channel when the channel is idle. Let the time of that event relative to the renewal point be denoted as ω . The period from onset of the idle period to ω is referred to as the deficit (or elapsed) time, and $T_i - \omega$ as the excess (or residual) time in an idle period on the channel. According to renewal theory, the deficit (elapsed) channel idle time for node has the probability distribution function (PDF) defined as $A(x) = P(\omega \leq x)$, while its pdf is $a(x) = \frac{dA(x)}{dx}$. Also, let $P(T_i > x) = T_i^c(x) = \int_{y=x}^{\infty} t_i(y)dy$. Then, the PDF $A(x)$ can be calculated as

$$A(x) = \frac{1}{\overline{T}_i} \int_0^x T_i^c(y)dy \quad (1)$$

and the pdf of deficit channel idle time as

$$a(x) = \frac{d}{dx} A(x) = \frac{T_i^c(x)}{\overline{T}_i} \quad (2)$$

Deficit and excess channel times have identical probability distributions but they are not independent [10].

IV. PROBABILITY OF LAGGING BETWEEN SEQUENCES

In the following, we will assume that the clock skew between the initiator and the follower is not destructive – i.e., the rendezvous may be accomplished regardless of the value of the skew. Three important events can occur on an idle channel, as shown in Fig. 1.

The arrival of the initiator node to the first channel η from the rendezvous sequence denotes the beginning of the rendezvous sequence. Let us denote this time as ρ ; it is

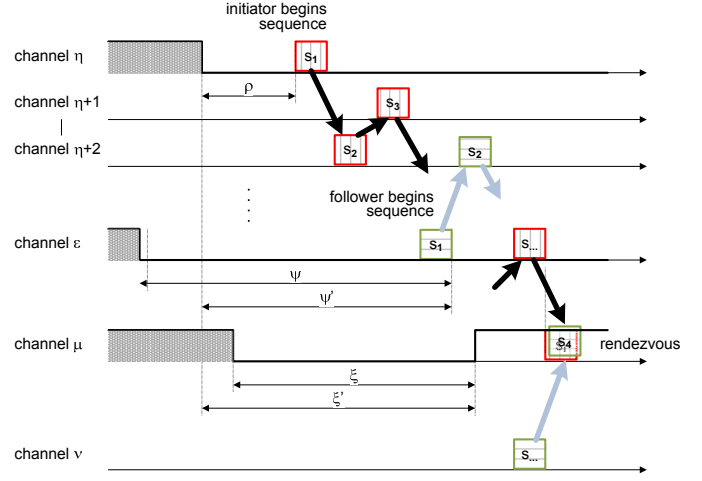


Fig. 1. Timing on initiator and follower node starting channels (η and ϵ , respectively) and rendezvous channel μ .

measured relative to the start of idle channel time for channel η .

The arrival of the follower node to the first channel ϵ of its rendezvous sequence denotes the beginning of its rendezvous sequence; it occurs at the time ψ after the beginning of the idle period on that channel. Since the beginning of the idle periods on channels η and ϵ are independent events, time of arrival of follower is also a random event with respect to the start of idle period at channel η . We will denote time of arrival of follower relative to start of η 's idle time as ψ' .

Given the lag of $0 \dots \Delta - 1$ r-slots between initiator and follower, and the (possible) difference between their individual sequences, rendezvous will occur at some channel μ . However, rendezvous may be interrupted by the onset of primary channel activity at time ξ relative to the beginning of the idle period on channel μ . Since the idle periods on channels η and μ are independent, the beginning of activity at channel μ will be a random event with respect to the start of idle time at channel η . Finally, let ξ' denote the beginning of primary user activity at rendezvous channel μ relative to the beginning of idle time at channel η .

Consider the follower node which arrives to the channel ϵ and begins its sequence; at that time, the initiator can be in any of the Δ r-slots of its sequence. The probability that the follower node comes to idle channel ϵ before the completion of i -th, but after the completion of $i - 1$ -th r-slot, conditioned upon the event that the initiator sequence begins at time $\rho = x$, is $P(\rho = x)P(x + (i - 1)\delta < \psi' < x + i\delta)$. By unconditioning for $0 \leq x \leq \infty$, we obtain the probability of lagging as

$$P'_{lag,i} = \int_{x=0}^{\infty} P(\rho = x)P(x + (i - 1)\delta \leq \psi' \leq x + i\delta)dx \\ = \int_{x=0}^{\infty} (A(x + i\delta) - A(x + (i - 1)\delta)) a(x)dx \quad (3)$$

where $i \in (1 \dots \Delta)$.

In order to obtain the probability distribution of the lag, we

need to scale those probabilities with their sum:

$$P_{lag,i} = \frac{P'_{lag,i}}{\sum_{i=1}^{sl} P'_{lag,i}} \quad (4)$$

Then, we need to find the probability that the channel on which rendezvous is to occur will become busy during the sequence. Assume that the channel changes its state at time $\rho \leq \xi' \leq \psi' + \Delta$. We note that primary activity on any channel after time ρ will require that initiator/follower remove that channel from the sequence, i.e. change the sequence. However, following derivations will assume that modifications to the sequence will not affect rendezvous.

The probability that channel μ will change to busy state, (during execution of the sequence) conditioned upon the events that the initiator sequence begins at time $\rho = x$, and that follower is lagging behind initiator by i r-slots, is $P(\rho = x)P(x \leq \xi' \leq x + i\delta + \Delta)$:

$$P'_{one,i} = \int_{x=0}^{\infty} P(\rho = x)P(x \leq \xi' \leq x + i\delta + \Delta)dx$$

$$P'_{one,i} = \int_{x=0}^{\infty} (A(x + i\delta + \Delta) - A(x))a(x)dx \quad (5)$$

Note that the probability of a channel changing state is obtained by using the probability that the residual idle channel time is larger than x , but smaller than $x + i\delta + \Delta$.

If the initiator sequence has started when all m channels from the sequence were idle (as is commonly assumed in the literature), the moment of sequence start is random with respect to the beginning of the idle period on each channel. Given that all channels have identical distribution of idle and active durations, expression (5) holds for any channel from the sequence.

V. PROBABILITY OF DESTROYED RENDEZVOUS AND TTR IN THE PRESENCE OF PRIMARY USER ACTIVITY

The last result can't be used directly since a rendezvous can occur in an arbitrary r-slot during the follower sequence, and we need to find the probability that rendezvous is destroyed by primary activity on the channel where rendezvous should occur according to the sequence.

In other words, we need to find the probability that the rendezvous channel μ becomes active before or at rendezvous point. Although the position of rendezvous depends on lag and follower's sequence, we will assume (somewhat optimistically) that the rendezvous can occur uniformly in any of the Δ r-slots in the follower sequence. The probability of primary activity on channel μ between the start of the initiator sequence (which lags i r-slots) and k -th slot of the follower's sequence can be calculated as

$$P_{one,i,k} = \int_{x=0}^{\infty} P(\rho = x) \frac{P(x \leq \xi' \leq x + (i+k)\delta)}{\Delta} dx$$

$$P_{one,i,k} = \int_{x=0}^{\infty} \frac{1}{\Delta} (A(x + (i+k)\delta) - A(x))a(x)dx \quad (6)$$

Again, since the initiator sequence has started when all m channels were idle, (6) holds for any channel from the sequence.

Probability that the initiator/follower sequence (which lags i r-slots) is broken by primary user activity on the rendezvous channel is

$$P_{one,i} = \sum_{k=1}^{\Delta} P_{one,i,k}$$

$$= \frac{1}{\Delta} \sum_{k=1}^{\Delta} \int_{x=0}^{\infty} (A(x + (i+k)\delta) - A(x))a(x)dx \quad (7)$$

By unconditioning $P_{one,i}$ on the lagging value i , we find the probability that a rendezvous will be destroyed as

$$P_d = \sum_{i=1}^{\Delta} P_{lag,i}P_{one,i} \quad (8)$$

Given the primary user activity, time-to-rendezvous becomes a random variable. To characterize its probability distribution, we begin by modeling the duration of rendezvous sequence broken by primary user activity. If the maximum time-to-rendezvous without primary user activity was TTR_m , the Probability Generating Function (PGF) of a broken follower sequence can be modeled using uniform distribution as $T_r(z) = \sum_{i=1}^{TTR_m} \frac{1}{TTR_m} z^i$. As the time between two consecutive rendezvous attempts is subject to different policies depending on the actual protocol (e.g., one or both sequences may be continued, restarted, or an attempt can be made to resynchronize them), we will just model it with a PGF $I(z)$. Then, the PGF for the maximum time-to-rendezvous becomes

$$TTR_M(z) = (1 - P_d)z^{TTR_m}$$

$$+ P_d T_r(z) I(z) (1 - P_d) z^{TTR_m}$$

$$\dots$$

$$+ (P_d T_r(z) I(z))^i (1 - P_d) z^{TTR_m}$$

$$= \sum_{i=0}^{\infty} (P_d T_r(z) I(z))^i (1 - P_d) z^{TTR_m}$$

$$= \frac{(1 - P_d) z^{TTR_m}}{1 - P_d T_r(z) I(z)} \quad (9)$$

Mean and variance of the maximum TTR become

$$\overline{TTR_M} = \left. \frac{d}{dz} TTR_M(z) \right|_{z=1} \quad (10)$$

$$var(TTR_M) = \left. \frac{d^2}{dz^2} TTR_M(z) \right|_{z=1} + \overline{TTR_M}^2 - \overline{TTR_M}^2 \quad (11)$$

So far we have been looking at primary user activity only on the channel on which the rendezvous will occur. However, we can take a more conservative approach and require that all of the channels visited during the initiator and follower sequences must be idle from the beginning of the initiator sequence up to and including the rendezvous r-slot. If this is not the case, the busy channel may be dropped from the

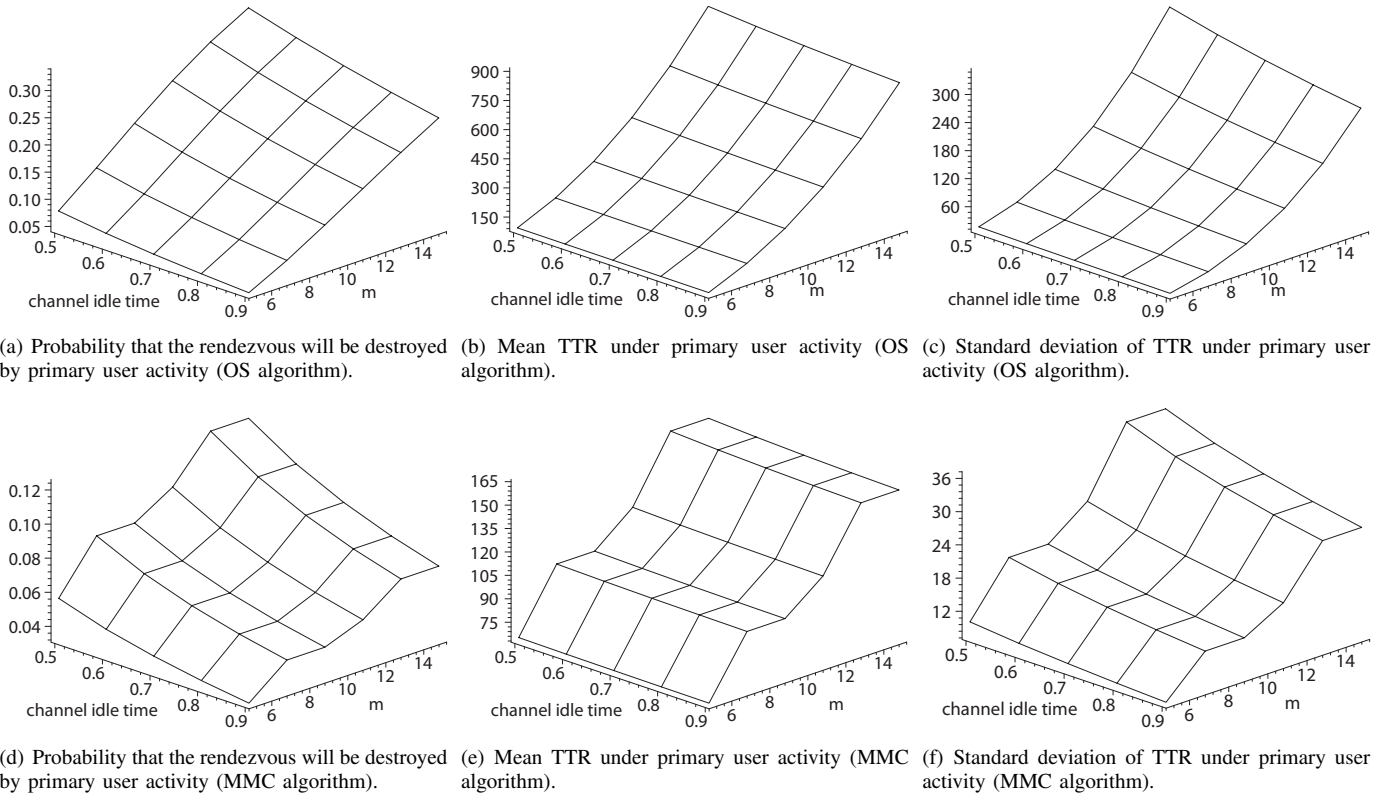


Fig. 2. Performance of two sequence-based rendezvous algorithms under primary user activity, variable number of channels and idle time on the primary channels.

sequence. The sequence itself will need to be recalculated (since the number of channels has changed) and, possibly, restarted. The drawback of this approach is that the sequences can easily exhaust the available channels, even though a busy channel may again become idle and should be re-attached to the pool used to generate the sequence; however, exclusion of such channels from the pool means that they will not be visited again until the sequence is reset with the original number of channels.

However, even with these changes, a rendezvous slot may still be affected by a sudden onset of primary user activity, and the analysis presented above remains valid, albeit the numerical values may change somewhat.

VI. EXPERIMENTAL EVALUATION

To evaluate the impact of primary user activity on the performance of sequence-based rendezvous algorithms, we have calculated the probability that the sequence based rendezvous will be destroyed by onset of activity of primary user between the moment when initiator starts the sequence and moment where rendezvous actually occurs. We have also calculated the expected value (i.e., mean) and variance of time-to-rendezvous when primary users are active.

Example rendezvous algorithms that we use are the orthogonal sequence (OS) algorithm proposed in [7], [18] and the modified modular clock (MMC) algorithm [12]. In the

former, the sequence is obtained by interspersing a random permutation of the channel set with individual channels from that permutation; it contains $m(m + 1)$ channels (and, consequently, rendezvous slots) which is also the upper bound for the rendezvous time TTR_m . In the latter, the sequence is dynamically generated using prime number modular arithmetic based on the smallest prime number larger than m . We stress that these two algorithms are used only as representatives for the general class of sequence-based approaches; other sequence-based algorithms will be affected by primary user activity in a similar manner.

To eliminate the impact of any actual technology, we have used an abstract time unit, and set the sequence slot to last for $\delta = 30$ time units. We have varied number of channels involved in sequence hopping from 5 to 15 in steps of 2. We assumed that channel idle and busy times due to primary user activity are exponentially distributed with average values \bar{T}_i and \bar{T}_a respectively. Total average cycle time of primary source on each channel is $T_{cyc} = \bar{T}_i + \bar{T}_a = 10000$ basic slots. The channel idle time was varied from 0.5 to $0.9T_{cyc}$. For both sequences, when a rendezvous is destroyed, we assume that the nodes wait for one sequence slot ($I(z) = z$) and continue their respective sequences.

We have then solved the analytical model outlined above using Maple 13 from Maplesoft, Inc. [15], to obtain performance indicators shown in Fig. 2. Diagrams in the upper

row correspond to the orthogonal sequence (OS) algorithm, while those in the lower row correspond to the modular clock (MMC) algorithm. The discontinuity in the diagrams for the MC algorithm stem from the need to select the smallest prime number greater than m , so that scenarios with different number of channels use the same prime number to generate the modular clock sequence.

As can be seen, the probability that a rendezvous will be destroyed is rather high, reaching as high as 0.3 (i.e., 30%) for the OS algorithm under the shortest channel idle time and the largest number of channels. The MMC algorithm is somewhat more resilient, with only up to about 12% rendezvous being destroyed through primary user activity. While it may seem counterintuitive that more channels should result in greater chance that a rendezvous will be destroyed, one should keep in mind that the length of the OS sequence increases quadratically with the number of channels – it is $m(m + 1)$ for m channels – which accounts for the increase.

Mean TTR also increases in a nonlinear fashion with the number of channels. For the OS algorithm, the increase is quite dramatic, while the MMC algorithm performs much better. Even more interesting is the increase in the standard deviation of TTR which, for the OS algorithm, amounts to about one-third of an already high mean TTR at $m = 15$ channels, and well beyond the ideal values which are quite small. Also, when the number of channels is higher, the standard deviation of TTR begins to show a dependence on the channel idle time, increasing further when the channel idle time approaches $0.5T_{cyc}$.

VII. DISCUSSION AND CONCLUSIONS

We stress once again that our goal was not to compare different sequence-based algorithms, as this is partially done in most of the paper that propose such algorithms. Instead, we aimed to evaluate the behavior of two representative algorithms under random primary user activity, a topic which does not show up too often in the literature. For example, neither of the experimental comparisons reported in [17] and [16] accounts for the presence of primary user activity. A comparative analysis that does include primary user activity was presented in [3], where conclusions similar to ours were reached on the basis of simulations, but without a theoretical model.

Overall, these results confirm our hypothesis that the presence of primary user activity may not only increase the mean time-to-rendezvous of all algorithms, including the sequence-based ones, but will actually invalidate their respective upper bounds obtained under ideal scenario with no primary user activity. Our future work will focus on the performance of probabilistic rendezvous algorithms.

REFERENCES

- [1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty. NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*, 50:2127–2159, 2006.
- [2] C. J. L. Arachchige, S. Venkatesan, and N. Mittal. An asynchronous neighbor discovery algorithm for cognitive radio networks. In *3rd IEEE Int'l Symp. on New Frontiers in Dynamic Spectrum Access Networks DySPAN 2008*, Chicago, IL, Oct. 2008.
- [3] K. Bian and J.-M. Park. Asynchronous channel hopping for establishing rendezvous in cognitive radio networks. In *Proc. INFOCOM*, pages 236–240, Apr. 2011.
- [4] K. Bian and J.-M. Park. Maximizing rendezvous diversity in rendezvous protocols for decentralized cognitive radio networks. *IEEE Transactions on Mobile Computing*, 12(7):1294–1307, 2012.
- [5] K. Bian, J.-M. Park, and R. Chen. Control channel establishment in cognitive radio networks using channel hopping. *IEEE J. on Selected Areas in Communications – Wireless Series*, 29(4):689–703, 2011.
- [6] C. Cormio and K. R. Chowdhury. Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping. *Ad Hoc Networks*, 8(4):430–438, 2010.
- [7] L. DaSilva and I. Guerreiro. Sequence-based rendezvous for dynamic spectrum access. In *3rd IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN 2008)*, pages 1–7, Oct. 2008.
- [8] Y. Duan, G. Liu, and Z. Cai. Opportunistic channel-hopping based effective rendezvous establishment in cognitive radio networks. In *Wireless Algorithms, Systems, and Applications*, pages 324–336, 2012.
- [9] R. Gandhi, C.-C. Wang, and Y. C. Hu. Fast rendezvous for multiple clients for cognitive radios using coordinated channel hopping. In *9th Conf. Sensor, Mesh and Ad Hoc Communications and Networks (SECON'2012)*, pages 434–442, June 2012.
- [10] D. P. Heyman and M. J. Sobel. *Stochastic Models in Operations Research, Volume I: Stochastic Processes and Operating Characteristics*. McGraw-Hill, New York, 1982.
- [11] B. Horine and D. Turgut. Link rendezvous protocol for cognitive radio networks. In *2nd IEEE Int'l Symp. on New Frontiers in Dynamic Spectrum Access Networks DySPAN 2007*, pages 444–447, Dublin, Ireland, Apr. 2007.
- [12] Z. Lin, H. Liu, X. Chu, and Y.-W. Leung. Jump-stay based channel-hopping algorithm with guaranteed rendezvous for cognitive radio networks. In *Proc. INFOCOM*, pages 2444–2452, Apr. 2011.
- [13] H. Liu, Z. Lin, X. Chu, and Y.-W. Leung. Ring-walk based channel-hopping algorithms with guaranteed rendezvous for cognitive radio networks. In *Proc. 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, GREENCOM-CPSCOM '10, pages 755–760, 2010.
- [14] H. Liu, Z. Lin, X. Chu, and Y.-W. Leung. Jump-stay rendezvous algorithm for cognitive radio networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(10):1867–1881, Oct. 2012.
- [15] Maplesoft, Inc. *Maple 13*. Waterloo, ON, Canada, 2009.
- [16] A. Robertson, L. Tran, J. Molnar, and E.-H. Fu. Experimental comparison of blind rendezvous algorithms for tactical networks. In *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, pages 1–6, June 2012.
- [17] M. Silvius, A. MacKenzie, and C. Bostian. Rendezvous MAC protocols for use in cognitive radio networks. In *IEEE Military Communications Conference (MILCOM 2009)*, pages 1–7, Oct. 2009.
- [18] N. C. Theis, R. W. Thomas, and L. A. DaSilva. Rendezvous for cognitive radios. *IEEE Transactions on Mobile Computing*, 10(2):216–227, Feb. 2011.
- [19] D. Yang, J. Shin, and C. Kim. Deterministic rendezvous scheme in multichannel access networks. *Electronics Letters*, 46(20):1402–1404, 30 2010.
- [20] J. Zhang and Z. Zhang. Initial link establishment in cognitive radio networks without common control channel. In *IEEE Wireless Communications and Networking Conference (WCNC 2011)*, pages 150–155, 2011.