Efficient Information Exchange in Single-Hop Multi-Channel Radio Networks

Weijie Shi[1], Qiang-Sheng Hua[1], Dongxiao Yu[2], Yuexuan Wang[1], and Francis C.M. Lau[2]

[1] Institute for Theoretical Computer Science, Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, 100084, P.R. China
[2] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong, P.R. China
swj05652@gmail.com, {qshua,amywang}@mail.tsinghua.edu.cn, {dxyu,fcmlau}@cs.hku.hk

Abstract. This paper studies the information exchange problem in single-hop multi-channel radio networks, which is to disseminate k messages stored in k arbitrary nodes to the entire network (with n nodes) with the fewest timeslots. By using $\Theta(\sqrt{n})$ channels, the previous best result [9] showed that this problem can be solved in $\Theta(k)$ time slots with high probability even if k is unknown and no bounds on k are given. Under the same assumptions but by using $\Theta(n)$ channels, this paper presents a novel randomized distributed algorithm called Detect-and-Drop that can solve the information exchange problem in $O(\log k \log\log k)$ time slots with high probability. Thus by allowing using more channels, our proposed algorithm contributes an exponential improvement in running time compared to that in [9]. The simulation results corroborate the analysis result.

Keywords: information exchange, single hop, multiple channels, randomized algorithm, distributed algorithm.

1 Introduction

Recent advances in wireless technology have made multi-radio multi-channel wireless networks possible practically [13]. Compared with using only one single channel, by using multiple channels, can we improve the performance of various common communication primitives such as broadcast [3] and information exchange [9,6]? The information exchange problem is to disseminate k messages stored in k arbitrary nodes to the entire network (with n nodes) with the fewest timeslots, which is also known as the multiple-message broadcasting problem [11] or the many-to-all communication problem [2]. The information exchange problem generalizes two well-known problems—broadcast ($k = 1$) and gossip ($k = n$). In this paper we restrict ourselves to single-hop networks, where each node in the network can communicate directly with every other node. The information exchange problem in single-hop networks may also take the form of the

X. Wang et al. (Eds.): WASA 2012, LNCS 7405, pp. 438–449, 2012.

k-selection problem [4,14] or the contention resolution problem [18] where each of the k contenders has to exclusivly access a shared communication channel at least once.

To our best knowledge, the first study on the information exchange problem under multiple channels is due to Holzer, Pignolet, Smula and Wattenhofer [9]. In their paper, by using $\Theta(\sqrt{n})$ channels, the authors showed that the problem can be solved in $O(k)$ time slots with high probability even if k is unknown and no bounds on k are given. Since their paper restricts that each node can only receive one piece of information in each time slot, an obvious lower bound for the information exchange is $\Omega(k)$. Thus their proposed algorithm is asymptotically optimal. However, since each node can be equipped with multiple radios [13], the node can actually receive multiple messages simultaneously on multiple channels. Bearing this in mind, this paper seeks to find if there are efficient distributed algorithms that can solve the information exchange problem in $o(k)$ time slots. Our proposed algorithm answers this question affirmatively.

2 Our Contribution

In this paper, we present a randomized distributed algorithm which can complete the information exchange in $O(\log k \log\log k)$ time slots with probability at least $1 - 1/k^c$ for some constant $c > 0$ where k is the number of nodes that hold a message.[1] Our algorithm does not assume any information on the number of nodes n or the number of messages k. Although our algorithm uses more channels ($\Theta(n)$) than that ($\Theta(\sqrt{n})$) in [9], we can exponentially reduce the time needed for accomplishing the information exchange.

3 Related Work

The information exchange problem and its variants have been extensively studied in the past decades, both for single-hop networks [18,4,14,5,9,17] and multi-hop networks [11,12,8]. By taking advantage of the collision detection ability [16] (which distinguishes between background noise and collision), Martel [15] presented a randomized adaptive protocol for the information exchange problem that works in $O(k + \log n)$ time in expectation. As argued by Kowalski in [14], this protocol can be improved to $O(k + \log\log n)$ in expectation using Willard's expected $O(\log\log n)$ selection protocol [17]. Without assuming the collision detection ability, Fernández Anta, Mosteiro, and Muñoz [5] proposed the Exp BackOn/BackOff algorithm which can solve the information exchange problem in $O(k)$ timeslots with high probability. Also without the collision detection ability, Yu et al. gave a randomized distributed algorithm for the dynamic version of the information exchange problem, where the messages may arrive in an adversarial pattern, in $O(k + \log^2 n)$ time slots with high probability.

[1] Throughout the paper, log means $\log_2$.

For multiple channels, to our best knowledge, the only efficient distributed algorithms for the information exchange problem in single-hop networks was proposed by Holzer, Pignolet, Smula and Wattenhofer in [9]. This paper gave both randomized and deterministic distributed algorithms. For the information exchange problem in multi-hop networks, there are no known efficient distributed algorithms that take advantage of multiple channels. For more references on the distributed algorithms for the information exchange problem in single-hop radio networks, please refer to [5,18,9].

The rest of the paper is organized as follows. Section 4 describes the network model and defines the the problem. Then a simple algorithm called Multi-Channel BackOn/BackOff which is adapted from the Exp BackOn/BackOff algorithm will be presented in Section 5. Our main algorithm Detect-and-Drop is given and analyzed in Section 6 and Section 7, respectively. The empirical evaluation result is given in Section 8 and we conclude the paper with some open problems in Section 9.

4 Model and Problem Definition

We consider a single-hop radio network consisting of n nodes and multiple channels. It is assumed that there are $4n$ available channels (later we will see that our algorithm can solve the problem with $\Theta(n)$ channels). Without loss of generality, these channels are numbered by $1, 2, \ldots, 4n$.

Initially, k $(1 \leq k \leq n)$ different messages are assigned to k arbitrary nodes, one message per node. The information exchange problem is to deliver the k messages to all nodes in the shortest time.

It is assumed that nodes have no any prior information about n or k, nor any estimates of these parameters. The only prior knowledge given to nodes is the linear relation between the number of channels and the number of nodes. So when a node selects a channel, it may actually select an unavailable channel and it will never know whether this channel is valid or not. Time is slotted into synchronous time slots. At the beginning of each time slot, a node can choose a channel and send a message via it. A message can be received if and only if there is exactly one node transmitting on a channel. If multiple nodes transmit on a channel simultaneously, a collision occurs and none of these transmissions can be correctly received. We assume that nodes are primitive and have no ability to detect collisions. At the end of the time slot, all nodes receive the successful broadcasts on multiple channels. This assumption is practical, since each node can be equipped with multiple radios [13]. It is also assumed that the sender can get feedback information from the channel on whether its transmission is successful.

5 Multi-Channel BackOn/BackOff Algorithm

In this section, we present a variant of the Exp BackOn/BackOff algorithm in [5] which can complete the information exchange in $O(\log^2 k)$ time with high probability. In the algorithm, nodes have two states: ACTIVE and IDLE. Nodes that have a message to transmit are in state ACTIVE initially. In the main algorithm (Algorithm 2), by estimating on the number of active nodes which increases exponentially, active nodes iteratively run a subroutine as described in Algorithm 1 trying to transmit their messages. In Algorithm 1, active nodes adopt a balls-into-bins strategy to achieve the appropriate transmission probabilities. An active node will set its state as IDLE after successfully transmitting its message.

Algorithm 1. Back-Off SubRoutine ($DROP(w)$)

Require: $w > 0$
while $w \geq 1$ **do**
  Broadcast on channel i ($1 \leq i \leq w$) with probability $1/w$.
  if Broadcast Success **then**
    Set IDLE
  end if
  $w \leftarrow w \cdot (1 - \delta)$
end while

Algorithm 2. Multi-Channel BackOn/BackOff

for $i = \{1, 2, \ldots\}$ **do**
  Run: $DROP(2^i)$
end for

Using an analysis for the algorithm similar to that for the single channel case in [5], if w satisfies $k \leq w \leq 2n$, all active nodes have successfully transmitted their messages after executing subroutines $DROP(w)$ and $DROP(2w)$ for $O(\log k)$ time slots with high probability. We present the following result without giving the detailed proof.

Lemma 1. *For constant* $0 < \delta < 1/e$, *with probability at least* $1 - 1/k^c$ *for some constant* $c > 0$, *the information exchange can be completed in* $O(\log w)$ *time slots after* w *satisfies* $k \leq w \leq 2n$.

Since before w satisfies the condition $k \leq w \leq 2n$, in each iteration, it takes $O(\log k)$ time to execute the subroutine described in Algorithm 1. Then based on Lemma 1, the following theorem can be obtained.

Theorem 1. *For constant* $0 < \delta < 1/e$, MULTI-CHANNEL BACKON/BACKOFF *can complete the information exchange process in* $O(\log^2 k)$ *time slots with probability at least* $1 - 1/k^c$ *for some constant* $c > 0$.

6 Detect-and-Drop Algorithm

In this section, based on the subroutine $DROP$ given in Algorithm 1, we present a faster randomized distributed algorithm which can complete the information exchange in $O(\log k \log\log k)$ time with high probability. As shown in Algorithm 5, by doubly exponentially increasing the estimates on the value of k, active nodes iteratively execute the subroutine LDD given in Algorithm 4. The subroutine LDD consists of two parts. First, active nodes execute the subroutine $DETECT$ given in Algorithm 3 to get all possible estimates of k. Then for each obtained possible estimate, active nodes try to transmit their messages by calling the subroutine $DROP$ as described in Algorithm 1 twice. Next we will describe the algorithm in more details.

With the estimate s, in the execution of the subroutine $DETECT(s)$, active nodes broadcast on $\log s$ channels and set the transmission probabilities on these channels exponentially. In each round during the execution of the subroutine $DETECT(s)$, if on some channel i, the number of successful broadcasts exceeds $\log s$, 2^i will be seen as a possible estimate of k. If the input $s \geq k$, there is a channel i satisfying $k \leq 2^i < 2k$. In the analysis, we will show that with high probability, there will be at least $\log s$ successful broadcasts on channel i after $6 \log s$ rounds. Thus the proper estimate of k will be included in the output set $ResultSet$. Furthermore, we will also show that when $s \leq k^2$, there are at most $2 \log\log k$ channels on which more than $\log s$ successful broadcasts occur. This will ensure that the time complexity of our information exchange algorithm is at most $O(\log k \log\log k)$.

For each possible value 2^i obtained after executing the subroutine $DETECT$, all active nodes will try to transmit their messages by calling subroutines $DROP(2^i)$ and $DROP(2^{i+1})$. As discussed above, when $s \geq k$, with high probability, an estimate 2^j which locates in the interval $[k, 2k)$ is output by the subroutine $DETECT$. By Lemma 1, all active nodes can successfully broadcast their messages after executing the subroutines $DROP(2^j)$ and $DROP(2^{j+1})$ for $O(\log k)$ time slots.

Furthermore, in order to get an s as the input of $DETECT(s)$ in Algorithm 5 such that $s \geq k$, s is doubly exponentially increased instead of the traditional exponential increase. To guarantee the correctness and efficiency of the algorithm, we will show that an estimate of k in the interval $[k, k^2]$ is needed. The doubly exponential increase on one hand accelerates the estimation process; on the other hand, it ensures that there must be an estimate s in the interval $[k, k^2]$.

Algorithm 3. Detect SubRoutine ($DETECT(s)$)

Require: $s > 0$

1: **for** $j = 1$ **to** $6 \log s$ **do**
2:     Broadcast on channel i $(1 \leq i \leq \log s)$ with probability $1/2^i$.
3:     Count the number of messages received on each channel
4: **end for**
5: **return** $ResultSet = \{i|$ more than $\log s$ messages received on channel $i\}$

Algorithm 4. A Loop of Detect-and-Drop ($LDD(s)$)

Require: $s > 0$
Run: $DETECT(s)$, Get $ResultSet$
for all $i \in ResultSet$ **do**
  Run: $DROP(2^i)$
  Run: $DROP(2^{i+1})$
end for

Algorithm 5. Detect-and-Drop

for $i = \{1, 2, \ldots\}$ **do**
  Run: $LDD(2^{2^i})$
end for

7 Analysis

In this section, we analyse the correctness and efficiency of our information exchange algorithm. Specifically, we show that with probability $1 - \frac{1}{k^c}$ for some constant $c > 0$, the information exchange can be completed in $O(\log k \log \log k)$ time. Before starting the proof for the main theorem, some commonly used inequalities are in order.

Lemma 2. *If* $0 < p \leq 1/2$, *then* $(\frac{1}{4})^{1/p} \leq 1 - p \leq (\frac{1}{e})^{1/p}$.

Lemma 3. *If* n *is a sufficiently large natural number,* $\sqrt{2\pi n} e^{n \log n - n} \leq n! \leq (\sqrt{2\pi n} + \sqrt{\pi/50n}) e^{n \log n - n}$.

By computing the differential, the following lemma can be easily proved.

Lemma 4. *If* $1/2 \leq x \leq 1$, *then* $1/4 \leq x/4^x \leq 1/3$. *If* $x > 0$, *then* $f(x) = x/e^x$ *is monotonically increasing on* $(0, 1)$ *and monotonically decreasing on* $(1, +\infty)$.

Lemma 5. *For sufficiently large* k, *the following statements hold with probability at least* $1 - 1/k^c$ *for some constant* $c > 0$.

(i) *If* $s \geq k$, *there exists* $i \in ResultSet$ *such that* $k \leq 2^i < 2k$.

(ii) *If* $k \leq s \leq k^2$, *the* $ResultSet$ *returned by the subroutine* $DETECT(s)$ *satisfies:* $|ResultSet| \leq 2 \log \log k$.

Proof. (i) Next we prove that if channel i satisfies $k \leq 2^i < 2k$, with large probability, at least $\log s$ broadcasts succeed in a total of $6 \log s$ broadcasts. Each node's transmitting probability on this channel is $p = 1/2^i$. So the probability that only one node transmits (and it can succeed) is:

$$P_{once} = kp(1-p)^{k-1} \geq kp(1/4)^{kp} \geq 1/4 \tag{1}$$

The first inequality is by Lemma 2. So the probability that less than $\log s$ broadcasts succeeding is:

$$P_{fail} \leq \log s \cdot \binom{6\log s}{\log s}(1-P_{once})^{5\log s} \tag{2}$$

We have the following inequality by using Lemma 3.

$$\binom{6\log s}{\log s} \leq s^{5\ln 5 - 4\ln 4 + 0.01} \tag{3}$$

Thus, $P_{fail} \leq \log s/s \leq 1/k^{\frac{1}{2}}$ for large enough k.

(ii) Suppose that i is the required value in (i). We show that with high probability, for any channel j, $1 \leq j \leq i - \log\log k$ or $i + \log\log k \leq j \leq \log s$, $j \notin ResultSet$.

If $1 \leq j \leq i - \log\log k$, the transmission probability of a node on j is $p \geq \frac{\log k}{2k}$. In one round, the probability that a successful broadcast occurs is

$$P_{once} = kp(1-p)^{k-1} \leq 2kp(\frac{1}{e})^{kp} \leq \log k(\frac{1}{e})^{\log k/2} \leq \frac{\log k}{k^{3/4}} \tag{4}$$

And the probability that at least one successful broadcast occurs on these $\log\log k$ channels in $6\log s$ rounds is

$$P_{fail} \leq 6\log s \cdot \log\log k \cdot P_{once} \leq \frac{6\log s \log k \log\log k}{k^{3/4}} \tag{5}$$

which is small enough to guarantee that on channel $1, 2, \ldots, i - \log\log k$, no successful transmission occurs with probability $1 - 1/k^c$ for some constant $c > 0$.

If $i + \log\log k \leq j \leq \log s$, the transmission probability of a node on j is $p \leq \frac{1}{k\log k}$; similarly,

$$P_{once} \leq \frac{2}{\log k}(1/e)^{1/\log k} \leq 2/\log k \tag{6}$$

The probability that more than $\log s$ successful transmissions occur on j is

$$P_{fail} \leq 5\log s \cdot \binom{6\log s}{\log s}(2/\log k)^{\log s} \tag{7}$$

Notice that $k \leq s \leq k^2$, so we get that for sufficiently large k,

$$P_{fail} \leq 10\log k \cdot (60/\log k)^{\log k} \leq 1/k^3 \tag{8}$$

Thus, for each channel $j \geq i + \log\log k$, the probability that channel j is included in $ResultSet$ is at most $1/k^3$. Since there are at most $\log s \leq 2\log k$ such channels, the probability that none of these channels is included in $ResultSet$ is at least $1 - \frac{1}{k^2}$.

Combining everything together, we have shown that the size of $ResultSet$ is not larger than $2\log\log k$ with probability $1 - \frac{1}{k^c}$ for some constant $c > 0$. □

Lemma 6. *For large enough k, the following statements hold with probability at least $1 - 1/k^c$ for some constant $c > 0$.*
(i)If $s \geq k$, $LDD(s)$ completes the information exchange.
(ii) If $s \leq k^2$, $LDD(s)$ takes $O(\log s \log\log k)$ time slots.

Proof. (i) Obviously, all nodes get the same $ResultSet$. So they run $DROP()$ at the same time. By Lemma 5, there is an i in $ResultSet$ satisfying $k \leq 2^i < 2k \leq 2n$. Then by Lemma 5, $LDD(s)$ completes the information exchange.

(ii) In the subroutine $LDD(s)$, $DETECT(s)$ takes $O(\log s)$ time. Furthermore, by Lemma 5, $|ResultSet| \leq 2 \log\log k$. And each $DROP$ subroutine takes at most $O(\log s)$ time. All together, the total time for executing $LDD(s)$ is $O(\log s \log\log k)$. □

Theorem 2. *For large enough k, with probability larger than $1 - 1/k^c$ for some constant $c > 0$,* Detect-and-Drop *completes the information exchange in $O(\log k \log\log k)$ time.*

Proof. Assume $s' = 2^{2^j}$ is the largest number satisfying $s' \leq k$. Then $2^{2^{j+1}} = s'^2 \leq k^2$. So there exists an integer i that satisfies $k \leq 2^{2^i} \leq k^2$. By Lemma 6, the total running time is

$$\sum_{i=1}^{\log\log k} O(\log(2^{2^i}) \log\log k) \in O(\log k \log\log k) \tag{9}$$

which completes the proof. □

Indeed, for any known constant $c > 0$, Detect-and-Drop can solve the information exchange problem with cn channels in $O(\log k \log\log k)$ time. Here we use the case with n channels as an example. All nodes can split one round in the original Detect-and-Drop into four rounds. If a node broadcasts on Channel $4k + j$ in the lth round in the original algorithm, it broadcasts on Channel k in the $4l + j$th round now. The new algorithm takes four times that of the original time. Then we can get the following result

Corollary 1. *If the number of the channels $|C|$ is known as a function of n and $|C| = f(n) \in \Theta(n)$, then* Detect-and-Drop *can complete the information exchange in $O(\log k \log\log k)$ time.*

8 Simulation

In this section, we report our simulation of the original Exp BackOn/BackOff algorithm [5], the Multi-Channel BackOn/BackOff algorithm and the Detect-and-Drop algorithm. The simulation measures the average number (10 trials for each experiment) of time slots that the algorithms take until they complete the information exchange, for different values of k ($k = 10^4, 10^{4\frac{1}{3}}, 10^{4\frac{2}{3}}, 10^5, 10^{5\frac{1}{3}}, \ldots$). The constant is chosen to be $\delta = 0.366$. We set

$n = k$ and provide $2n$ and $4n$ available channels, respectively. We modify the detecting times in Algorithm 3 from $6 \log s$ to $4 \log s$ because this is actually enough to get a correct *ResultSet* in practice while not affecting the time complexity. No failure cases occur in all these simulations, which verifies that the success probabilities of these algorithms are all indeed very high.

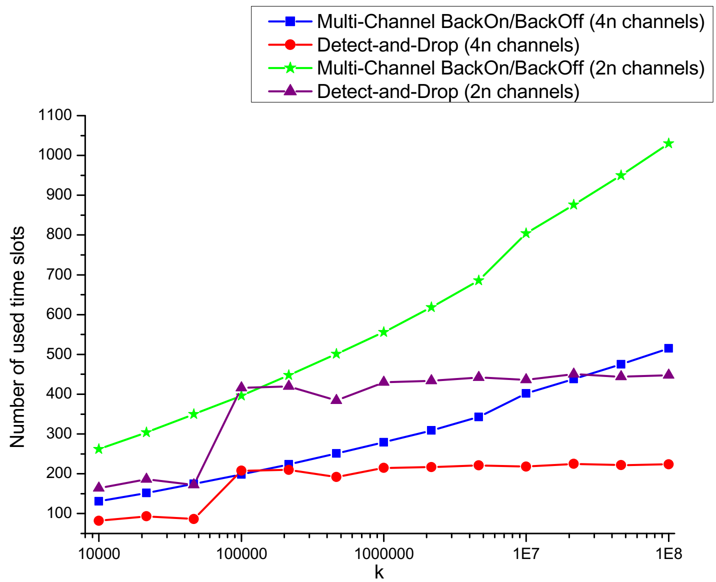

Fig. 1. Number of time slots to solve the information exchange problem

Fig. 1 and Table 1 show the result of the simulation. Table 2 reveals the hidden constant of Detect-and-Drop calculated by the ratio of time slots to the value of $\log k \log \log k$.

The simulation result speaks for the huge advantage of utilizing multi-channel technique. Among the five scenarios that have been simulated, the Exp BackOn/BackOff algorithm which uses only one channel takes the largest number of time slots, as shown in Table 1. For example, when $k = 10000$, it takes 71125 time slots to complete the information exchange with Exp BackOn/BackOff while our Detect-and-Drop algorithm only needs 82 time slots. Moreover, overall speaking, the algorithms that use $2n$ channels use more time slots than those that use $4n$ channels, as shown in Fig. 1.

Our Detect-and-Drop algorithm performs much faster than the Multi-Channel BackOn/BackOff algorithm in the simulation, as shown in Fig. 1, which verifies that our main algorithm Detect-and-Drop can greatly reduce the time complexity. In addition, the larger the k value, the more the number of reduced time slots. Finally, the running time of Detect-and-Drop increases slowly after $k \geq 10^5$, while the running time of Multi-Channel BackOn/BackOff increases rather rapidly.

As shown in Fig. 1, when $k \in \{10^{4\frac{2}{3}}, 10^5\}$, there is a jump in the running time of Detect-and-Drop. This is because for $k \geq 10^5$, Algorithm 5 needs one more execution round of Algorithm 4. With the same execution rounds of Algorithm 4, the running time grows slowly when k gets larger. (The running time is almost the same for $10^5 \leq k \leq 10^8$, as shown in Table 1.)

Table 1. Number of time slots to solve the information exchange problem

k	10^4	$10^{4\frac{1}{3}}$	$10^{4\frac{2}{3}}$	10^5	$10^{5\frac{1}{3}}$	$10^{5\frac{2}{3}}$	10^6
Exp BackOn/BackOff (single channel)	71125	1.56e+5	3.24e+5	6.58e+5	1.35e+6	2.74e+6	5.61e+6
Multi-Channel BackOn/BackOff ($2n$ channels)	262	304	350	396	448	501	556
Detect-and-Drop ($2n$ channels)	164	186	172	416	420	384	430
Multi-Channel BackOn/BackOff ($4n$ channels)	131	152	175	198	224	251	279
Detect-and-Drop ($4n$ channels)	82	93	86	208	210	192	215
k	$10^{6\frac{1}{3}}$	$10^{6\frac{2}{3}}$	10^7	$10^{7\frac{1}{3}}$	$10^{7\frac{2}{3}}$	10^8	
Exp BackOn/BackOff (single channel)	1.13e+7	2.27e+7	7.93e+7	1.68e+8	3.37e+8	7.74e+9	
Multi-Channel BackOn/BackOff ($2n$ channels)	618	686	804	876	950	1030	
Detect-and-Drop ($2n$ channels)	434	442	436	450	444	448	
Multi-Channel BackOn/BackOff ($4n$ channels)	309	343	402	438	475	515	
Detect-and-Drop ($4n$ channels)	217	221	218	225	222	224	

Table 2. Hiding Constant of Detect-and-Drop: time slots/ $\log k \log \log k$

k	10^4	$10^{4\frac{1}{3}}$	$10^{4\frac{2}{3}}$	10^5	$10^{5\frac{1}{3}}$	$10^{5\frac{2}{3}}$	10^6
Hiding Constant	1.65	1.68	1.41	3.09	2.86	2.41	2.50
k	$10^{6\frac{1}{3}}$	$10^{6\frac{2}{3}}$	10^7	$10^{7\frac{1}{3}}$	$10^{7\frac{2}{3}}$	10^8	
Hiding Constant	2.35	2.24	2.07	2.01	1.87	1.78	

Table 2 shows that the hidden constant is quite small and keeps dwindling after $k \leq 10^5$. This is because Algorithm 3 has time complexity $O(\log k)$ with a large constant coefficient, which is dominated by the time complexity of Algorithm 5 ($O(\log k \log \log k)$) which has a relatively small constant coefficient.

9 Conclusion

In this work, for the information exchange problem in single-hop multi-channel radio networks, we have proposed a novel randomized distributed algorithm called Detect-and-Drop. This algorithm uses $\Theta(n)$ channels and can solve the information exchange problem in $O(\log k \log \log k)$ time slots with high probability. Our algorithm does not need any knowledge on the number of nodes n in the network and the number of messages k. Compared with the state-of-the-art algorithm in [9] which can solve the same problem in $\Theta(k)$ time slots using

$\Theta(\sqrt{n})$ channels, our algorithm contributes an exponential improvment. There are many interesting and meaningful future work topics that are related: (1) By using multi-radio multi-channels, it would be interesting to see if we can use $o(n)$ channels (sublinear number of channels in terms of the number of nodes n) to solve the information exchange problem in $o(k)$ time slots (sublinear number of channels in terms of the number of messages k); (2) similar to the work in [18] which considers adversarial (arbitrary) arrival patterns of the messages, it is still open whether our result also holds for this dynamic version of the information exchange problem; (3) it is meaningful to extend our work to other scenarios, such as the simple multiple-access channels where the channel cannot provide any feedback information to the sender and the multi-hop networks; (4) it is still open to design a deterministic distributed algorithm that can solve the information exchange problem in sublinear time complexity under multiple-channels; (5) it will be worthwhile to see if we can apply our technique to many other related problems, such as the wake-up problem [10], the broadcast problem [1], the local broadcasting problem [19], etc.

Acknowledgements. This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00302, the National Natural Science Foundation of China Grant 61103186, 61073174, 61033001, 61061130540, the Hi-Tech research and Development Program of China Grant 2006AA10Z216, and Hong Kong RGC-GRF grants 714009E and 714311.

References

1. Bar-Yehuda, R., Goldreich, O., Itai, A.: On the Time-Complexity of Broadcast in Multi-hop Radio Networks: An Exponential Gap Between Determinism and Randomization. J. Comput. Syst. Sci. 45(1), 104–126 (1992)
2. Chlebus, B.S., Kowalski, D.R., Radzik, T.: Many-to-Many Communication in Radio Networks. Algorithmica 54(1), 118–139 (2009)
3. Dolev, S., Gilbert, S., Khabbazian, M., Newport, C.: Leveraging Channel Diversity to Gain Efficiency and Robustness for Wireless Broadcast. In: Peleg, D. (ed.) DISC 2011. LNCS, vol. 6950, pp. 252–267. Springer, Heidelberg (2011)
4. Anta, A.F., Mosteiro, M.A.: Contention Resolution in Multiple-Access Channels: k-Selection in Radio Networks. In: Thai, M.T., Sahni, S. (eds.) COCOON 2010. LNCS, vol. 6196, pp. 378–388. Springer, Heidelberg (2010)
5. Fernández Anta, A., Mosteiro, M.A., Ramón Muñoz, J.: Unbounded Contention Resolution in Multiple-Access Channels. In: Peleg, D. (ed.) DISC 2011. LNCS, vol. 6950, pp. 225–236. Springer, Heidelberg (2011)
6. Gilbert, S., Kowalski, D.R.: Trusted Computing for Fault-Prone Wireless Networks. In: Lynch, N.A., Shvartsman, A.A. (eds.) DISC 2010. LNCS, vol. 6343, pp. 359–373. Springer, Heidelberg (2010)
7. Goldberg, L.A.: Design and analysis of contention-resolution protocols, EPSRC Research Grant GR/L60982, `http://www.csc.liv.ac.uk/~leslie/contention.html` (last modified October 2006)
8. Haeupler, B., Karger, D.R.: Faster information dissemination in dynamic networks via network coding. In: PODC, pp. 381–390 (2011)

9. Holzer, S., Pignolet, Y.A., Smula, J., Wattenhofer, R.: Time-optimal information exchange on multiple channels. In: FOMC, pp. 69–76 (2011)
10. Jurdziński, T., Stachowiak, G.: Probabilistic Algorithms for the Wakeup Problem in Single-Hop Radio Networks. In: Bose, P., Morin, P. (eds.) ISAAC 2002. LNCS, vol. 2518, pp. 535–549. Springer, Heidelberg (2002)
11. Khabbazian, M., Kowalski, D.R.: Time-efficient randomized multiple-message broadcast in radio networks. In: PODC, pp. 373–380 (2011)
12. Khabbazian, M., Kuhn, F., Kowalski, D.R., Lynch, N.A.: Decomposing broadcast algorithms using abstract MAC layers. In: DIALM-PODC, pp. 13–22 (2010)
13. Kodialam, M.S., Nandagopal, T.: Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In: MOBICOM, pp. 73–87 (2005)
14. Kowalski, D.R.: On selection problem in radio networks. In: PODC, pp. 158–166 (2005)
15. Martel, C.U.: Maximum Finding on a Multiple Access Broadcast Network. Inf. Process. Lett. 52(1), 7–15 (1994)
16. Schneider, J., Wattenhofer, R.: What Is the Use of Collision Detection (in Wireless Networks)? In: Lynch, N.A., Shvartsman, A.A. (eds.) DISC 2010. LNCS, vol. 6343, pp. 133–147. Springer, Heidelberg (2010)
17. Willard, D.E.: Log-Logarithmic Selection Resolution Protocols in a Multiple Access Channel. SIAM J. Comput. 15(2), 468–477 (1986)
18. Yu, D., Hua, Q.-S., Dai, W., Wang, Y., Lau, F.C.M.: Dynamic Contention Resolution in Multiple-Access Channels. In: Koucheryavy, Y., Mamatas, L., Matta, I., Tsaoussidis, V. (eds.) WWIC 2012. LNCS, vol. 7277, pp. 232–243. Springer, Heidelberg (2012)
19. Yu, D., Wang, Y., Hua, Q.-S., Lau, F.C.M.: Distributed local broadcasting algorithms in the physical interference model. In: DCOSS, pp. 1–8 (2011)