Dynamic Contention Resolution in Multiple-Access Channels

Dongxiao Yu[1], Qiang-Sheng Hua[2], Weiguo Dai[2], Yuexuan Wang[2,⋆], and Francis C.M. Lau[1]

[1] Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong, P.R. China
[2] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, 100084, P.R. China
amywang@mail.tsinghua.edu.cn

Abstract. Contention resolution over a multiple-access channel can be modeled as a k-selection problem in wireless networks, where a subset k of n network nodes want to broadcast their messages over a shared channel. This paper studies a dynamic version of this problem, which assumes that k messages arrive at an arbitrary set of k nodes (contenders) asynchronously and the message arrival pattern is determined by an online adversary. Under this harsh and more practical assumption, we give a randomized distributed algorithm which can guarantee any contender deliver its message in $O(k + \log^2 n)$ rounds with high probability. Our proposed algorithm neither relies on collision detection, nor a global clock or any knowledge about the contenders, not even its size k. Furthermore, we do not assume the channel can provide any kind of feedback information, which makes our protocol work in simple channels, such as the channels used in wireless sensor networks.

Keywords: Contention Resolution, Multiple-Access Channel, Radio Networks, Distributed Algorithm, Randomized Algorithm.

1 Introduction

Contention resolution is a fundamental operation for both wired and wireless networks, which as a problem has been extensively studied for many years. For a multiple-access channel—a broadcast channel that allows a multitude of users to communicate with each other by sending messages onto the channel—contention resolution can be modeled as a k-selection problem in radio networks [1,5], where each node in a subset k of n network nodes wants to exclusively access a shared communication channel at least once. In terms of message transmissions, this means the k nodes want to broadcast their messages to the single-hop network with n nodes. Due to shared nature of the channel, if two or more users send a message simultaneously, then their messages interfere with each other, and the messages will not be transmitted successfully. The goal of a contention resolution

⋆ Corresponding author.

Y. Koucheryavy et al. (Eds.): WWIC 2012, LNCS 7277, pp. 232–243, 2012.

protocol is to minimize the time for the nodes to access the shared channel. The most well-known example of contention resolution is ALOHA which was designed around 40 years ago. In ALOHA, when a sender encounters a collision, the sender will wait a random amount of time and send again. Due to its simplicity, this random back-off idea was later adopted by the current WiFi protocols (IEEE 802.11 protocols).

The static k-selection problem, where the k messages are ready at their respective nodes before the protocol starts, has been intensively studied; the state-of-the-art randomized static k-selection protocol, given in [5], needs $O(k+\log^2 n)$ rounds for all active nodes (nodes having messages to transmit) to successfully transmit their messages on the shared channel with high probability.[1] However, this protocol is uniform, i.e., all active nodes use the same transmission probability in the same communication step, which will not work when the message arrival pattern is arbitrary and a global clock is not available. The reason for this failure is that nodes do not know other nodes' statuses, not even the number of contenders. How to derive an efficient protocol for arbitrary message arrivals without a global clock is still open [5]. Furthermore, although previous work have considered the channel without collision detection—i.e., nodes can not distinguish between the case of no transmission and that of collision (multiple transmissions), they all assume that either the nodes can receive feedback information from the channel if the transmission is not successful, or a node by itself knows whether its transmission is successful, such that the nodes can decide whether it should quit the protocol after a transmission. This assumption is crucial for the correctness of these protocols. However, in some real wireless networks, such as wireless sensor networks, the channel can not provide any feedback. The protocol we introduce in this paper can work for these networks.

1.1 Related Work

The static k-selection problem has been studied since 1970s [2,10,17]. With the availability of collision detection, Martel [16] presented a randomized adaptive protocol with running time of $O(k + \log n)$ in expectation. In [14], it was shown that this protocol can be improved to $O(k + \log\log n)$ in expectation by making use of the expected $O(\log\log n)$ selection protocol in [20]. An $\Omega(\log\log n)$ expected time lower bound was also given in [20] for uniform selection protocols. Without collision detection, the state-of-the-art randomized protocol was presented in [5], which can solve the static k-selection problem in $O(k + \log^2 n)$ rounds with high probability. Given that k is a trivial lower bound, the protocol in [5] is asymptotically optimal for $k \in \Omega(\log^2 n)$. Furthermore, the result in [15] on the lower bound of the expected time needed to get the first message delivered implies an $\Omega(\log n)$ expected time lower bound for randomized k-selection protocols. In a recent paper [6], an $O(k)$ randomized protocol was proposed even without knowing n. However, the error probability of this protocol is $\frac{1}{k^c}$, rather

[1] We say an event succeeds with high probability if the error probability is at most $\frac{1}{n^c}$ for some constant $c > 0$.

than $\frac{1}{n^c}$. The performance of several kinds of randomized backoff k-selection protocols are analyzed in [1]. Apart from above work on static k-selection, the authors of a recent paper [11] showed that the $\Omega(k + \log n)$ lower bound for randomized protocols can be subverted when multiple channels are available. There are also several studies on the k-selection problem with dynamic packet arrivals, e.g., in stochastic model [7] and in the adversarial queuing model [1,3,14]. To the best of our knowledge, there have been no results on randomized protocols for k-selection with arbitrary message arrivals.

As for deterministic solutions for the static k-selection problem, the technique of tree algorithms, which models the protocol as a complete binary tree where the messages are placed at the leaves, has been used to produce adaptive protocols with running time $O(k \log(n/k))$ in [2,10,17]. All these protocols rely on collision detection. A lower bound of $\Omega(k \log_k n)$ is shown in [9] for this class of protocols. For oblivious algorithms, where the sequence of transmissions of a node does not depend on the received messages, by requiring prior knowledge on k and n, Komlòs and Greeberg [13] gave an $O(k \log(n/k))$ protocol without collision detection. The lower bound of $\Omega(k \log(n/k))$ for oblivious protocols was given in [4]. This lower bound also holds for adaptive algorithms without collision detection. If collision detection is unavailable, making use of the explicit selector given in [12], Kowalski [14] presented an oblivious deterministic protocol with $O(k poly \log n)$ running time. For more details of the contention resolution algorithms in the past four decades, interested readers are referred to an excellent online survey maintained by Goldberg [8].

1.2 Our Result

In this paper, we present the first known randomized distributed protocol for dynamic contention resolution in a multiple-access channel with arbitrary message arrivals. In particular, we show that each node can deliver its message on the shared channel in $O(k + \log^2 n)$ rounds with high probability. When applying to the static scenario, the proposed protocol has the same asymptotical time bound as the state-of-the-art result in [5]. Based on the trivial $\Omega(k)$ lower bound, our protocol is asymptotically optimal if $k \in \Omega(\log^2 n)$. The channel considered in this work is even more practical than the simple multiple-access channel in [1] since we assume that the channel in this paper does not provide any feedback information to nodes. Furthermore, the number of contenders is also unknown to the nodes and there is not any collision detection mechanism being assumed. There are three challenges in this dynamic version of the k-selection problem: The first one is how to let any node know that its transmission has succeeded without feedback from the channel, such that the node knows when to quit the protocol and as a result the contention is reduced; the second one is how to ensure that newly activated nodes would not interfere with other nodes' processing; the third one is how to coordinate the nodes' transmission probabilities such that each node can quickly deliver its message for any arbitrary pattern of message arrivals and in the absence of a global clock. In our protocol, we meet these challenges by electing a leader. Specifically, the leader elected serves three functions: first, it sends acknowledgement messages to inform nodes of their successful transmissions; second, it periodically transmits a trigger to notify newly activated nodes when they can start executing the protocol; third, it transmits controlling messages to adjust other nodes' transmission probabilities according to the transmission situation of the shared channel.

2 Preliminaries

Communication Model (The Multiple-Access Channel). We consider a single-hop radio network consisting of n nodes, in which each node is potentially reachable from any other node in a communication step. Communication is done in synchronized rounds, which means that all the nodes' clocks tick simultaneously at the same rate. However, we do not assume the existence of a global clock. So in a round, clock values may be different among the nodes. A node sets its clock value to 1 after being activated, and increases it at every round. If exactly one node is transmitting on the shared channel, all nodes receive the message transmitted by this node at the end of the round. A collision occurs when multiple messages are transmitted concurrently, which means that none of these messages can be successfully received by any node. Collision detection is not assumed, i.e., nodes can not tell apart the case of no transmission and that of collision (more than one node transmit concurrently). We further assume that the shared communication channel does not provide any feedback information in any case.

An external mechanism is assumed that generates messages and assigns them to nodes with the purpose of broadcasting them on the channel. For each node, there are three status: *idle*, *active* and *passive*. Before being assigned a message, a node is *idle* and does nothing. It becomes *active* if it is assigned a message and is ready to transmit on the shared channel. After successfully transmitting its message, the node switches its status to *passive*. In a round, active nodes can either transmit or listen, while passive nodes can only listen on the channel. We say that a node is activated in round i if it is assigned a message in round i. Each node has a unique but arbitrary ID, and it knows its own ID and the parameter n. Nodes have no other prior information about the network. Initially, a node does not know any other nodes' statuses, nor the number of nodes that compete with it for the channel.

Problem Definition. Suppose that a subset k of the set of n network nodes are activated by message arrivals. The k-selection problem is to make each of the k nodes deliver its message on the shared communication channel as quickly as possible.

Dynamic Setting. In this work, we consider a dynamic version of the k-selection problem. Messages may arrive at the nodes asynchronously and the arrival pattern is arbitrary, even in the worst case. In particular, we assume that the message arrivals are controlled by an on-line adversary. The adversary knows the protocol, but does not know the future random bits. Obviously, such an adversary is much stronger than the usually assumed oblivious adversary (which decides the message arrivals off-line).

Complexity Measure. We define the efficiency measurement of dynamic k-selection protocols as follows. We assume that comparing to the communication time consumed, the computation time cost is negligible. So we only care about the time efficiency in terms of communication rounds. Formally, we define the process latency of a node v as the length of the period between its activation time and the completion time (when it completes the message transmission and becomes passive). The time complexity of a k-selection protocol is the maximum value of all nodes' process latencies over any message arrival pattern. When messages arrive at nodes simultaneously, the above defined time complexity is just the same as that for the static k-selection protocols.

Finally, we give some inequalities and a Chernoff bound as follows which will be used in the protocol analysis.

Lemma 1. *([19]) Given a set of probabilities $p_1, \ldots, p_n$ with $\forall i : p_i \in [0, \frac{1}{2}]$, the following inequalities hold:*

$$(1/4)^{\sum_{k=1}^{n} p_k} \leq \prod_{k=1}^{n} (1 - p_k) \leq (1/e)^{\sum_{k=1}^{n} p_k}. \tag{1}$$

Lemma 2. *([19]) For all n, t, with $n \geq 1$ and $|t| \leq n$, it holds that:*

$$e^t(1 - \frac{t^2}{n}) \leq (1 + t/n)^n \leq e^t. \tag{2}$$

Lemma 3. *(Chernoff Bound) Suppose that X is the sum of n independent $\{0, 1\}$-random variables X_i's such that for each i, $Pr(X_i = 1) = p$. Let $\mu = E[X]$. Then for $0 \leq \epsilon \leq 1$,*

$$Pr(X \leq (1 - \epsilon)\mu) \leq e^{-\frac{\epsilon^2 \mu}{2}} \tag{3}$$

3 Algorithm

3.1 Algorithm Description

In this section, we propose our randomized contention resolution algorithm. In the algorithm, due to the assumption that the channel can not provide any feedback, a leader is elected to be responsible for acknowledging successful transmissions. By transmitting controlling messages, the leader also takes the responsibilities of adjusting other nodes' transmission probabilities and informing newly activated nodes when to start executing the algorithm. In particular, the leader and active non-leaders iteratively execute a 3-round scheme which is shown in Algorithm 1. The first round is for active non-leaders to transmit their messages. The second and the third rounds are used for leader's transmissions. In the first round, each active non-leader transmits its message with a specified transmission probability. If only one node u sends in the first round, the leader v can successfully receive the message. Then in the second round, v sends an acknowledgement message to inform u of the successful transmission. After receiving the acknowledgement message, u adjusts its status as passive and quit the algorithm. To deal with arbitrary arrivals of messages, in the third round, the leader also transmits a controlling message. If v's clock value is not $r_l + i \cdot 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ for some integer $i > 0$, where r_l is the last round before v starts executing the 3-round scheme and α is a constant given in Algorithm 1,[2] the controlling message transmitted by v would only carry a trigger which is to inform newly activated nodes to start executing the 3-round scheme from the next round. Otherwise, besides the trigger, the controlling message also contains information on how to adjust the transmission probabilities of other active nodes. Specifically, if v transmitted less than $8 \log n$ *Ack* messages in the past $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds, which means that nodes' transmission probabilities are not large enough to get many successful transmissions, v makes all active nodes double their transmission probabilities. Otherwise, it makes all nodes halve the transmission probability. The leader is elected through carrying out the MIS (maximal independent set) algorithm in [19], which can correctly compute a maximal independent set[3] in $O(\log^2 n)$ rounds with high probability. A newly activated node will first wait for at most three rounds. If it did not receive the trigger, which means that the leader has not been elected, then this newly activated node would start executing the MIS algorithm to compete for becoming a leader. Otherwise, it iteratively executes the 3-round scheme after receiving the trigger.

3.2 Analysis

In this section, we prove the correctness and efficiency of the proposed algorithm. Specifically, we show that for any node u, with high probability, it can successfully transmit its message on the shared channel after being activated for at most $O(k + \log^2 n)$ rounds. First, we state the correctness and the efficiency of the MIS algorithm in the following lemma which is proved in [19].

Lemma 4. *([19]) After executing the MIS algorithm for $O(\log^2 n)$ timeslots, a maximal independent set can be correctly computed with probability at least $1 - O(n^{-1})$.*

From the above lemma, a leader can be correctly elected after $O(\log^2 n)$ rounds with high probability. In the following, we assume that the leader is correctly computed; the error probability will be considered in the proof of the main theorem. Let's denote the elected leader as v. Next we give a lemma which

[2] Here we only give a value for α such that the proposed algorithm is correct and has the stated asymptotically running time bound with high probability. Since a different value of α only affects the time complexity of our algorithm by a constant factor, so we do not optimize the value we choose for α.

[3] For single hop networks, the MIS contains only one node.

Algorithm 1. 3-Round Scheme

Initially, $p_u = \frac{2^{-\alpha-1}}{n}$; $\alpha = 1$

3-round scheme for the leader v

1: listen

2: **if** v received a message from a non-leader u
  then transmit Ack_u
  end if

3: transmit a controlling message

3-round scheme for an active non-leader u

4: **if** u has a message to transmit
  then transmit the message with probability p_u
  end if

5: listen
  if u received Ack_u
  then quit the execution of the algorithm
  end if

6: listen
  if the received controlling message contains information on how to adjust the transmission probability
  then adjust the transmission probability accordingly
  end if

states that in any round the sum of transmission probabilities of all active nodes is bounded by a constant.

Lemma 5. *Assume that the leader is correctly elected. In any round during the execution of the algorithm, with probability* $1 - \frac{1}{n}$*, the sum of transmission probabilities of active nodes is at most* $2^{-\alpha}$*.*

Proof. Assume that r is the first round that the sum of transmission probabilities of active nodes exceeds $2^{-\alpha}$. Denote r_a as the last round until r in which the leader v transmits a controlling message to adjust other active nodes' transmission probabilities. By the algorithm, the leader v transmits a controlling message to adjust other active nodes' transmission probabilities every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds. So r_a must be in the interval $(r - 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n, r]$. Since r is the first violating round, in any round before r, for the sum of transmission probabilities of active nodes, we have $\sum_u p_u \leq 2^{-\alpha}$. Furthermore, by the algorithm and the definition of r_a, for nodes that are activated after round $r_a - 3$, in any round until r, the sum of transmission probabilities of these nodes is at most $\frac{2^{-\alpha-1}}{n} \times n = 2^{-\alpha-1}$. Then in any round during the interval $I = (r_a - 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n, r_a]$, for the sum of transmission probabilities of nodes that have been activated before $r_a - 2$, we have $\sum_u p_u \geq 2^{-\alpha-2}$, since each such node can only double their transmission probabilities once in round r_a. Before any violating round, there must be such an interval I. Next we show that during I, with probability $1 - n^{-2}$, v transmits at least $8 \log n$ *Ack* messages. Then by the algorithm, v makes all active nodes halve their transmission probabilities in round r_a, which leads to a contradiction.

Claim. During the interval I, with probability $1 - n^{-2}$, the leader v transmits at least $8 \log n$ *Ack* messages.

Proof. For a round r^*, denote the set of active nodes as A_{r^*}. During I, in the first round r_1 of each execution of the 3-round scheme, the probability P_{one} that there is only one node transmitting is

$$\begin{aligned} P_{one} &= \sum_{u \in A_{r_1}} p_u \prod_{w \in A_{r_1} \setminus \{u\}} (1 - p_w) \\ &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1} \setminus \{u\}} p_w} \\ &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1}} p_w} \end{aligned} \tag{4}$$

The second inequality is by Lemma 1. Note that the function $f(x) = x \left(\frac{1}{4}\right)^x$ is monotone increasing in the range $[2^{-\alpha-2}, 2^{-\alpha}]$. So we have

$$\begin{aligned} P_{one} &\geq \sum_{u \in A_{r_1}} p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_{r_1}} p_w} \\ &\geq 2^{-\alpha-2} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha-2}} \end{aligned} \tag{5}$$

By the algorithm and the above equation, during the interval I, in expectation, there are at least $16 \log n$ active non-leader nodes successfully transmitting their messages on the shared channel, since $\frac{1}{3}$ of the rounds in the interval I are used for non-leaders' transmissions. Then using the Chernoff bound in Lemma 3, the probability that v transmits less than $8 \log n$ *Ack* messages during I is at most $e^{-\frac{1}{8} \cdot 16 \log n} = n^{-2}$. □

By the above claim and the algorithm, with probability $1 - n^{-2}$, v makes active nodes halve their transmission probability in round r_a. Thus for all nodes which have been activated before $r_a - 2$, with probability $1 - n^{-2}$, the sum of transmission probabilities is at most $\sum_u p_u \leq \frac{1}{2} \times 2^{-\alpha} = 2^{-\alpha-1}$ in any round during the interval $I^{'} = [r_a, r_a + 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n)$, since v will not transmit another controlling message to adjust the nodes' transmission probabilities during $I^{'}$. Obviously, r is in $I^{'}$. Then combining the fact that the sum of transmission probabilities of all nodes that are activated after the round $r_a - 3$ is at most $2^{-\alpha-1}$ in round r, we have $\sum_{u \in A_r} p_u \leq 2^{-\alpha-1} + 2^{-\alpha-1} = 2^{-\alpha}$. So with probability $1 - n^{-2}$, r is not the first violating round.

To complete the proof, we still need to bound the number of potential violating rounds. From the above argument, before each potential violating round, with probability $1 - n^{-2}$, there are $\Omega(\log n)$ successful transmissions for active non-leader nodes during the corresponding interval I. So there are at most $O(\frac{k}{\log n})$ potential violating rounds and thus with probability at least $1 - n^{-1}$, none of these rounds are the first violating round. This completes the proof. □

Theorem 1. *For any node u, with probability $1 - O(n^{-1})$, it can successfully transmit its message on the shared channel after carrying out the algorithm for $O(k + \log^2 n)$ rounds.*

Proof. By the algorithm, if u is activated before the leader is elected, it needs to take part in the MIS algorithm. By Lemma 4, this process needs at most $O(\log^2 n)$ rounds with probability $1 - O(n^{-1})$. After that, u starts iteratively executing the 3-round scheme. If u is activated after the leader election process, it starts the 3-round scheme execution after waiting for at most three rounds. Next we bound the number of rounds needed for u in executing the 3-round scheme before receiving the Ack_u message from the leader v. From then on, we assume that the leader v is correctly computed, which means that there is only one leader. And we assume that in any round, the sum of transmission probabilities of active nodes is at most $2^{-\alpha}$. The error probability will be considered at last.

For a round, we call it successful if there is only one non-leader node transmitting in it. By the algorithm, when u executes the 3-round scheme, u adjusts its transmission probabilities every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds. So after starting executing the 3-round scheme, for every $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds, either a constant of these rounds are successful, which makes u halves its transmission probability, or u doubles its transmission probability once. So after at most $2 \cdot \frac{k-1}{8\log n} \cdot 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n + 3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log^2 n$ rounds, u has a constant transmission probability $2^{-\alpha-1}$, since there are at most $k-1$ contenders when u is active. Next we show that u can successfully transmit its message in the subsequent $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds with probability $1 - n^{-2}$. Denote P_{only} as the probability that u is the only transmitting node in a round r. Denote A_r as the set of active nodes in round r. Then we have

$$\begin{aligned} P_{only} &= p_u \prod_{w \in A_r \setminus \{u\}} (1 - p_w) \\ &\geq p_u \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_r \setminus \{u\}} p_w} \\ &\geq 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{\sum_{w \in A_r} p_w} \end{aligned} \tag{6}$$

By Lemma 5, $\sum_{w \in A_r} p_w \leq 2^{-\alpha}$. Then

$$P_{only} \geq 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}} \tag{7}$$

By the algorithm, u transmits in $\frac{1}{3}$ of the $3 \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n$ rounds. Thus the probability P_{no} that all of these $2^{\alpha+6+2^{-\alpha-1}} \log n$ transmissions are unsuccessful is at most

$$\begin{aligned} P_{no} &\leq (1 - 2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}})^{2^{\alpha+6+2^{-\alpha-1}} \log n} \\ &\leq e^{-2^{-\alpha-1} \cdot \left(\frac{1}{4}\right)^{2^{-\alpha}} \cdot 2^{\alpha+6+2^{-\alpha-1}} \log n} \\ &\leq n^{-2} \end{aligned} \tag{8}$$

The second inequality is by Lemma 2. So u will successfully transmit its message in the subsequent $3 \cdot 2^{\alpha+6+2^{-\alpha-1}}$ rounds with probability $1 - n^{-2}$. This means that after executing the 3-round scheme for $O(k + \log^2 n)$ rounds, with probability $1 - n^{-2}$, u can successfully transmits its message on the shared channel. This claim is true for any node with probability $1 - n^{-1}$.

Finally, we combine everything together. Based on the above argument, we know that for any node u, with probability $1 - O(n^{-1})$, it takes at most $O(k + \log^2 n)$ rounds in executing the MIS algorithm and the 3-round scheme. Furthermore, note that the above argument is under the assumptions that the leader is correctly computed and the sum of transmission probabilities of active nodes is upper bounded by $2^{-\alpha}$ in any round. By Lemma 4 and Lemma 5, these two assumptions are true with probability $1 - O(n^{-1})$. Thus any node u can successfully transmit its message after being activated for $O(k + \log^2 n)$ rounds with probability $1 - O(n^{-1})$, which completes the proof.

4 Conclusion

In this paper, we solve the dynamic contention resolution problem in the multiple-access channel, also called the dynamic k-selection problem, where the message arrival pattern is determined by an online adversary. To the best of our knowledge, our protocol is the first one considering an arbitrary pattern of message arrivals. We show that the proposed protocol can make each node successfully deliver its message on the shared channel in $O(k + \log^2 n)$ rounds with high probability, which is optimal when $k \in \Omega(\log^2 n)$. Our protocol neither relies on collision detection, nor on a global clock or any knowledge about the number of contenders k. In addition, we do not assume the channel can provide any feedback information which is commonly used in existing contention resolution protocols. Thus our contention resolution protocol can be applied to a variety of wireless networks such as wireless sensor networks without such a function. Interesting future work include how to extend our result to wireless networks without even an estimation of the number of network nodes n. Furthermore, it is also meaningful to analyze the performance of our protocol in the adversarial queueing model [1].

Acknowledgements. The authors thank the anonymous reviewers for their helpful comments. 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. Bender, M.A., Farach-Colton, M., He, S., Kuszmaul, B.C., Leiserson, C.E.: Adversarial contention resolution for simple channels. In: Proc. of the 17th Ann. ACM Symp. on Parallel Algorithms and Architectures, SPAA, pp. 325–332 (2005)
2. Capetanakis, J.: Tree algorithms for packet broadcast channels. IEEE Trans. Inf. Theory IT-25(5), 505–515 (1979)
3. Chlebus, B.S., Kowalski, D.R., Rokicki, M.A.: Adversarial queuing on the multiple-access channel. In: Proc. of the 25th ACM Symposium on Principles of Distributed Computing, PODC, pp. 92–101 (2006)
4. Clementi, A., Monti, A., Silvestri, R.: Selective families, superimposed codes, and broadcasting on unknown radio networks. In: Proc. of the 12th Ann. ACM-SIAM Symp. on Discrete Algorithms, SODA, pp. 709–718 (2001)
5. Fernández Anta, A., 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)
6. Fernández Anta, A., Mosteiro, M.A., Muñoz, J.R.: Unbounded Contention Resolution in Multiple-Access Channels. In: Peleg, D. (ed.) Distributed Computing. LNCS, vol. 6950, pp. 225–236. Springer, Heidelberg (2011)
7. Goldberg, L.A., Jerrum, M., Kannan, S., Paterson, M.: A bound on the capacity of backoff and acknowledgment-based protocols. SIAM Journal on Computing 33, 313–331 (2004)
8. 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)
9. Greenberg, A., Winograd, S.: A lower bound on the time needed in the worst case to resolve conflicts deterministically in multiple access channels. Journal of the ACM 32, 589–596 (1985)
10. Hayes, J.F.: An adaptive technique for local distribution. IEEE Trans. Comm. COM-26, 1178–1186 (1978)
11. Holzer, S., Pignolet, Y.A., Smula, J., Wattenhofer, R.: Time-Optimal Information Exchange on Multiple Channels. In: Proc. of the 7th ACM SIGACT/SIGMOBILE International Workshop on Foundations of Mobile Computing, FOMC, pp. 69–76 (2011)
12. Indyk, P.: Explicit constructions of selectors and related combinatorial structures, with applications. In: Proc. of the 13th Ann. ACM-SIAM Symp. on Discrete Algorithms, SODA, pp. 697–704 (2002)
13. Komlòs, J., Greenberg, A.: An asymptotically nonadaptive algorithm for conflict resolution in multiple-access channels. IEEE Trans. Inf. Theory 31, 303–306 (1985)
14. Kowalski, D.R.: On selection problem in radio networks. In: Proc. of 24th Ann. ACM Symp. on Principles of Distributed Computing, SPAA, pp. 158–166 (2005)
15. Kushilevitz, E., Mansour, Y.: An $(D\log(N/D))$ lower bound for broadcast in radio networks. SIAM Journal on Computing 27(3), 702–712 (1998)

16. Martel, C.U.: Maximum finding on a multiple access broadcast network. Inf. Process. Lett. 52, 7–13 (1994)
17. Mikhailov, V., Tsybakov, B.S.: Free synchronous packet access in a broadcast channel with feedback. Problemy Peredachi Inform. 14(4), 32–59 (1978)
18. Mitzenmacher, M., Upfal, E.: Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press (2005)
19. Moscibroda, T., Wattenhofer, R.: Maximal independent sets in radio networks. In: Proc. of 24th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing, PODC, pp. 148–157 (2005)
20. Willard, D.E.: Log-logarithmic selection resolution protocols in a multiple access channel. SIAM Journal on Computing 15, 468–477 (1986)