Self-Organizing WSN Protocol for Real-Time Communication Requirements

János Sallai, Péter Horváth, Xenofon Koutsoukos
Institute for Software Integrated Systems, Vanderbilt University, USA
Email: janos.sallai@vanderbilt.edu

***Abstract*—In this paper we propose a distributed, low-power, self-organizing MAC scheme for low-power wireless sensor and control applications. In such applications, nearly periodic traffic needs to be handled so that the communication delays are minimized and the transmission is as predictable as possible. We propose Asynchronous Random Schedules with Collision Forecast (ARS/CF) for this purpose, together with a multi-channel extension based on an improved modulation scheme to improve capacity scaling of the basic ARS/CF scheme. We analyze and simulate the basic single-hop ARS/CF implementation with regard to throughput and delay when used both in single- and multi-channel settings. Finally, we consider the implementation of ARS/CF in a multi-hop setting.**

I. Introduction

Designing wireless communications platforms for time-critical applications, such as real-time industrial process control networks, sensor networks etc. is not entirely straightforward. As network size increases, the network predictability, in terms of delay guarantees and bandwidth allocations, tends to decrease. The tradeoff between scalability and predictability in today's wireless ad-hoc, sensor and control networks limits the scale of safety-critical application deployments to up to a few dozen to few hundred nodes. Current wireless mesh networking technologies that have been designed specifically with predictability as an objective rely on (1) fixed infrastructure nodes and gateway nodes with special hardware and dedicated roles in a hierarchical network, (2) predefined communication schedules that are settled at deployment time or calculated and distributed by a centralized coordinator entity in advance, and (3) global control to manage the network. These characteristics result in inflexible deployments that are hard to extend because of constraints on the topology and the need of recomputing the communication schedules. Also, such systems are prone to single points of failure, in particular at higher levels of the network hierarchy.

At the other end of the spectrum are contention-based protocols that do offer scalability, at the cost of defaulting to providing best-effort services. They do not provide guarantees on the timing of network primitives, which is insufficient in a large class of application scenarios. In typical wireless nodes both transmission and reception are equally costly operations in terms of power consumption. Typical battery-powered nodes get depleted more quickly due to the wasted transmission and reception cycles due to the channel contention.

We claim that features common to wired real-time networking services, such as global synchrony, permanence of assigned communication slots within a radio frame, strict periodicity at the communications layer, or deployment-time knowledge of communication timing are not crucial, and are often not explicitly required by the applications, and are relied on simply for convenience. By rethinking the application requirements and redefining the problem by using a slightly weaker set of assumptions at the application layer, we can design a communications stack that redraws the predictabilty-scalability tradeoff curve, significantly reducing the predictability tradeoff in large-scale wireless ad-hoc applications.

We claim that a large class of timing-critical applications remains feasible if the above assumptions are relaxed, as long as the radio stack can provide (1) timing guarantees (static properties such as worst-case delays, and dynamic ones such as time to next communication opportunity), (2) guarantees that quality of service properties are met (guaranteed bandwidth and latency), and (3) if the topology is allowed to change, a means for the application layer to adapt to topology changes.

In this paper we present a central element of our proposed smart network stack for time-critical networked applications: the media access protocol and an associated improved physical layer design. The key to our proposed MAC approach is pseudo-random seed exchange, a lightweight cooperation mechanism that makes it possible to compute shared schedules without global control. Contrary to many seed-exchange MAC protocols, our scheme enables the nodes to accurately predict their next successful transmission epochs, requires only the exchange of small amounts of information and results in large power savings in steady state by minimizing transceiver wake-ups due to unnecessary transmissions and receptions.

The work in [1] reports the fundamentals of our collision forecasting scheme, Asynchronous Random Schedules with Collision Forecast (ARS/CF) and demonstrates its successful proof-of-concept implementation and evaluation results using TinyOS with Iris wireless motes. Because of its proof-of-concept nature, ARS/CF is a single-hop MAC protocol, and assumes a fully connected network topology. This paper extends the results presented in [1] by (1) considering multi-hop extension, (2) proposing and analyzing a multi-channel extension to ARS/CF to improve the scalability of the scheme, (3) providing more accurate throughput formulas for the single-hop case, with simulation validation, and (4) providing simulation results to gain better insight into the performance of basic ARS/CF variants.

The paper is organized as follows. Section II provides a brief survey of state-of-art distributed MAC protocols based on seed exchange. Section III summarizes the features of our collision forecast scheme. Section IV describes extensions to the basic protocol to increase the number of nodes the system can accommodate, and considers techniques to extend ARS/CF over multiple hops. Section V revisits the performance evaluation for the single-channel case and extends it to the multi-channel variant. Finally, Section VI presents simulation results to validate our approach.

II. Related work

There are existing MAC approaches that aim at providing transmission schedules in a distributed manner. The topology-independent scheduling algorithms in [2] exploits the algebraic structures generated by unique node identifiers to derive schedules that guarantee at least one successful transmission within a frame. Similarly, [3] uses a different algebraic structure to maximize the minimum throughput within the network in a topology-independent way. Being robust against topology changes, their throughput is low, comparable to ALOHA. Both protocols inherently calculate with collisions, therefore their energy efficiency deteriorates significantly under heavy load. Thanks to the topology-independence, signalling requirements are low. These protocols assume accurate network-wide time synchronization. The f-MAC protocol in [4] requires no global clock synchronization and provides certain delay and throughput guarantees by retransmitting the same short frames multiple times. However, establishing such guarantees mean a tradeoff between delays and throughput, and has a profound impact on the power consumption, both on the transmitter and receiver side.

On the other hand, topology-dependent distributed protocols, generally based on leader election, are also feasible. SEEDEX [5] announces the schedules by exchanging pseudo-random seed values, and selects time slots for potential transmission based on the known neighbor schedules. In NAMA [6], nodes are contending for a certain time slot within a well-defined time slot structure by broadcasting one-hop neighborhood information. Finally, NOMAD [7] exchange detailed lists of future transmissions between the nodes to enable a deterministic scheduling, leading to predictable inter-packet delays. All these distributed election protocols require continuous monitoring of neighbor's broadcasts, and involve exchange of potentially large amounts of data, therefore they are not very energy efficient.

III. Collision forecasting

The underlying idea of ARS/CF can be summarized as follows. For more details, refer to [1]. Each node has an independent pseudo-random communication schedule, that describes an infinite sequence of transmission intents in its local clock time. Such a schedule can be succinctly represented by a seed-timestamp pair, and an algorithm relying on a pseudo-random number generator that deterministically computes the time assignments of all future communication slots. Through sharing these seeds within the collision domain, nodes run a

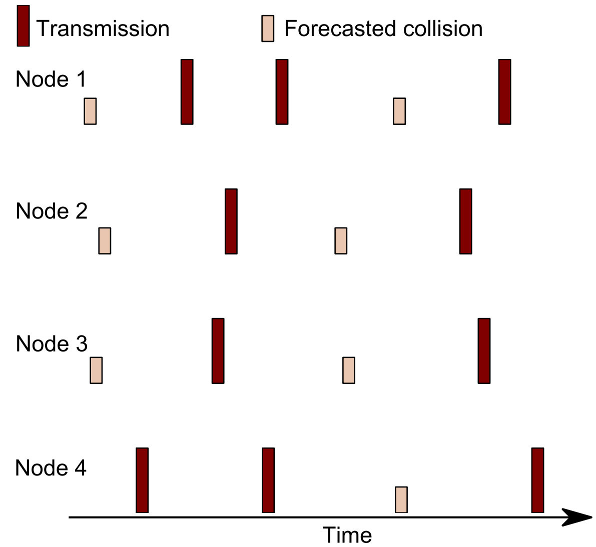

Fig. 1. Illustration of the ARS/CF protocol

collision forecasting (CF) algorithm locally that reconstructs, merges, and prunes the collected schedules of imminent collisions. Since individual nodes execute the CF algorithm on an identical set of inputs, the resulting schedules will necessarily be identical across the collision domain, and therefore are guaranteed to be free of collisions. Fig. 1. shows the example operation with four nodes, with successful transmission attempts (dark bars) as well as abandoned transmission intents due to forecasted collisions (light bars).

The computed schedule remains valid as long as the network topology remains unchanged. That is, after carrying out pairwise seed exchange between all pairs of nodes within the collision domain, the nodes are aware of all future communication times within their neighborhood. Most importantly, the timing assignment of these slots are guaranteed. Being aware of the transmit schedules of all the nodes in the neighborhood, collisions can be forecasted and avoided.

A. Schedule maintenance

One of the simplest pseudo-random number generators is the linear feedback shift register (LFSR). The next value of the shift register is completely determined by its current state. It is this state that needs to be communicated among the nodes within the collision domain, and this state is being used to define the time instant of the next transmission intent.

Nodes maintain a neighbor table with transmission times and random seed values for each neighbor, as well as for the node itself. The neighbor table is kept up-to-date, such that the transmission time value for each neighbor always contains the time of the next transmission anticipated from the particular neighbor. To compute collisions, all we have to do is identifying pairs of entries in the neighbor table with transmission times that differ less than the time it takes to transmit a packet. This time is referred to as the *collision windows size*. For entries with collisions forecasted, the packet transmission time, as well as the random seed, is updated to their next values applying the LFSR shift operation, canceling the respective transmit or receive actions.

An important property of collision forecasting is that it is deterministic: since the same information is available at each node, the decisions cancel a particular transmission or wake-up will be identical across the neighborhood.

An other important and unique property of our proposed ARS/CF scheme is the possibility to shape the distribution of the time intervals between consecutive transmission intents. If we denote the current LFSR state (an integer number) by s, and its normalized value by $\tilde{s}$, therefore $\tilde{s}$ is a uniformly distributed random number over $(0, 1]$. If, on average, uniform packet inter-arrival times are desired for the application (e.g. for sensor or control purposes), we generate the packet intervals from the normalized LFSR state as follows:

$$T_s = T_{min} + \tilde{s}(T_{max} - T_{min}), \tag{1}$$

where T_{min} and T_{max} are the minimum and maximum time interval between two consecutive transmission attempts. Therefore, this mapping yields to inter-packet times drawn from a uniform distribution over $(T_{min}, T_{max}]$, with average $T_{average} = \frac{T_{min}+T_{max}}{2}$. Alternatively, the following mapping would yield to exponential inter-packet intervals:

$$T_s = T_w - T_{average} \cdot \log(\tilde{s}), \tag{2}$$

where T_w is the length of the collision window.

Assuming that every node is programmed with the same procedure that generates future packet intervals using a current LFSR state, only the transmission time of a message and the LFSR state that was used to generate the packet interval preceding the message need to be stored for each neighbor. This information is sufficient to compute future packet transmission attempt times, and hence, it is a compact representation of a (virtually infinite) transmission schedule.

The protocol is in principle not limited to one seed (pseudo-random generator) per node, but each node might be able to advertise multiple seeds with straightforward modification of the original scheme. This allows nodes with higher traffic needs to multiply the average number of transmission opportunities.

B. Advantages of ARS/CF

Collision forecasting offers power savings in multiple aspects. First, from the sender's point of view, a transmission is suppressed when a collision is predicted, so no power is spent on transmitting packets that would collide. Second, from the receiver's aspect, the radio is not turned on to receive packets that could be corrupted as a result of a collision. Since all the receivers in the neighborhood that forecast the collision cancel their receiver wake-ups, this typically accounts for more power saving than suppressing the two (or more) colliding transmissions.

One important difference between TDMA schedules and the merged and pruned schedules in our proposed approach is that while a TDMA schedule is strictly periodic, the common schedule in the CF approach is not, as it is derived from merged pseudo-random sequences. We can, however, design the schedule generation and collision forecasting algorithms in a way that statistical properties on QoS parameters are met at design-time, and design the application in a way that it adapts to the schedule as it becomes available run-time. By appropriate application design, ARS/CF can be an efficient communication platform for systems with real-time communication requirements, such as networked control systems and other networked cyber-physical systems.

The protocol does not require global clock synchronization, the nodes use their local clocks to time stamp the received seed values, which is a major advantage over many existing collision-free MAC scheme. This insensitivity to clock offsets is supported by the findings of our single-hop proof-of-concept implementation [1].

IV. Extensions to basic ARS/CF

The basic ARS/CF scheme presented in [1] works for single-hop networks and assumes that only one node can access the channel at a time. In this work we consider qualitatively the necessary steps to lift these limitations.

A. Accommodating multiple simultaneous transmissions

With the evolution of low-cost, low-power transceiver components it becomes feasible to share the channel for simultaneous transmissions from multiple nodes by employing appropriate modulation schemes. Code Division Multiple Access (CDMA) is a widely investigated scheme in the sensor network context, however, it did not find widespread usage in practical deployments. Recently, Interleaved Frequency Domain Multiple Access [8], [9] has been proposed as a favorable alternative to CDMA. IFDMA can be regarded as a frequency-spread OFDM scheme, thus accommodating multiple users by spreading their information over orthogonal sets of subcarriers in an interleaved manner, allowing simultaneous transmissions. On the other hand, it is a battery-efficient single-carrier scheme, lends itself to low-complexity channel equalization and adaptive modulation is straightforward to implement. The latter two are distinct advantages over conventional ZigBee/IEEE 802.15.4-based transceivers. A similar scheme is Localized Frequency Domain Multiple Access (LFDMA), in which adjacent subcarriers are assigned to the same terminal. The generation of these signals using an M-point DFT and N-point IDFT ($N > M$) is illustrated in Fig. 2. With IFDMA, if we denote $Q = \frac{N}{M}$, the rth user uses IDFT subcarriers $r, r+Q, r+2Q, \ldots, r+(M-1)Q$, whereas with LFDMA, the rth user uses subcarriers $rQ, rQ+1, rQ+2, \ldots, (r+1)Q$, with $r = 0, 1, \ldots, Q-1$.

These xFDMA-like (IFDMA or LFDMA) schemes are well suited for ARS/CF. Choosing which set of subcarriers will be assigned to a certain node before the frequency spreading can be derived from the states of the pseudo-random generators by a simple mapping. Collision only occurs if two nodes would use the same set of subcarriers on the same carrier frequency in within overlapping collision windows. Therefore xFDMA fits within our deterministic collision prediction framework.

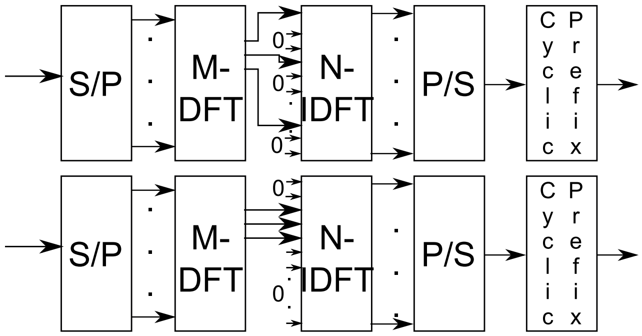

Fig. 2. Illustration of IFDMA (upper) and LFDMA (lower) signal flow

It seems to be natural to further improve the performance of the system by using multiple carrier frequencies ("RF channels") in a pseudo-random frequency hopping manner, and deriving the used carrier frequency index from the same pseudo-random number. We leave as a future work to study the practical implications of using multiple channels over ARS/CF, but the subsequent analysis should equally apply to this case as well.

B. Multi-hop operation

The basic ARS/CF scheme assumed a fully connected network. Extensions to multi-hop network topographies are possible. The most straightforward solution involves seed dissemination in a way similar to the well-known RTS/CTS mechanism: the transmitting node places not only its own current seed value into the transmitted packet, but also that of the destination node. The destination node acknowledges the reception by advertising its own and the sender's seed. Therefore, schedule information from a two-hop environment becomes available to every node with extremely low signaling overhead.

Clearly, relying only on 2-neighbor data yields to suboptimal power consumption. This is straightforward to understand on an example assuming a line network with nodes A, B, C and D, if only next-hop neighbor can hear each other. If node A learns its transmission could collide with a transmission from its second neighbor C, the transmission intent is canceled. However, C might be forced to cancel its overlapping transmission intent due to a collision with D, therefore A could have used the canceled intent to communicate with B. The lack of information about the three-hop neighborhood also causes unnecessary receiver wake-ups.

Obtaining seeds and timestamps from a 3-hop neighborhood mitigates this problem. On the other hand, this information can realistically be obtained by relying on higher-layer data transmission.

V. Performance evaluation

In this section we extend our previous results in [1] and provide the accurate throughput of the single-channel ARS/CF scheme, in which we drop the usual approximations (Poisson traffic and very low per-node activity).

We start with analyzing the expected throughput, defined as the number of over-the-air packets within the collision domain per unit time. The number of nodes in the system is N, and each of them executes the protocol with identical parameters. We also assume there are no packet losses in the system. We denote the CDF of the time intervals elapsed between successive transmission intents by $F(x)$. The average rate of transmission intents per node is denoted by λ, therefore the average time between successive intents is $1/\lambda$.

Clearly, if the successful transmission starts at time t, there must not be another transmission taking place in the interval $[t - T_w, t + T_w]$, for which the probability amounts to [10]

$$P_{succ} = \left[1 - \lambda \int_0^{2T_w} [1 - F(u)]\, du\right]^{N-1}. \tag{3}$$

For transmission intents generated according to the uniform mapping in (1), $F(x)$ takes the following form:

$$F(x) = \begin{cases} 0 & \text{if } x < T_{min}, \\ \frac{x - T_{min}}{T_{max} - T_{min}} & \text{if } T_{min} \le x \le T_{max}, \\ 1 & \text{if } x > T_{max}. \end{cases} \tag{4}$$

Exploiting that in a practical system $T_w < T_{min}$, (3) can be evaluated as

$$P_{succ} = (1 - 2T_w\lambda)^{N-1}, \tag{5}$$

and taking into account that the average number of transmission intents per unit time is $N\lambda$, the effective throughput is given as

$$S_1 = N\lambda[1 - 2T_w\lambda]^{N-1}. \tag{6}$$

As it can be easily proven by the binomial theorem, for large N and small T_w, (6) yields the well-known ALOHA throughput. Essentially, ARS/CF behaves like ALOHA, with a significant advantage: collisions in fact do not happen, the transceivers do not wake up, therefore transmission and reception energy does not get wasted.

Similarly, the loss in throughput due to abandoned transmission intents can be quantified using (5) as

$$W_1 = N\lambda \left[1 - (1 - 2T_w\lambda)^{N-1}\right]. \tag{7}$$

Now turning to the K-channel case (K might be equal to Q with the proposed xFDMA schemes), we realize that transmissions fall onto the same frequency-domain resource with probability $1/K$ when the channel index is derived from the psudo-random seed. Therefore the collision probability in (3) need to be modified accordingly. The throughput then turns out to be

$$S_K = N\lambda \left[1 - \frac{2T_w\lambda}{K}\right]^{N-1}, \tag{8}$$

and the loss of throughput due to the abandoned attempts is

$$W_K = N\lambda \left[1 - \left(1 - \frac{2T_w\lambda}{K}\right)^{N-1}\right]. \tag{9}$$

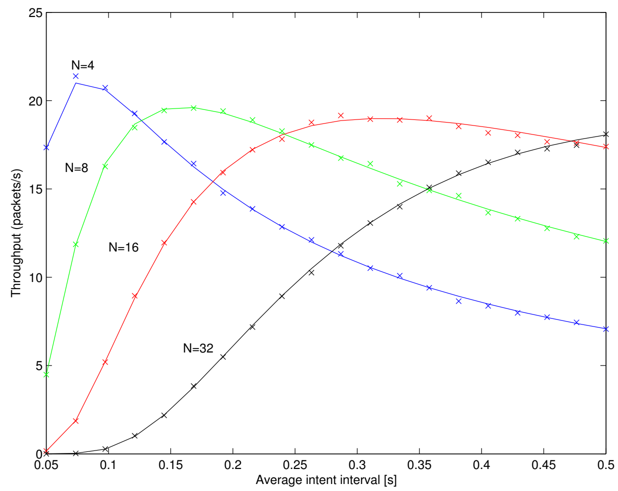

Fig. 3. Network throughput vs. mean intent interval for $K = 1$ available channel, $T_w = 10$ ms, uniform interval distribution. Crosses represent simulation outcomes.

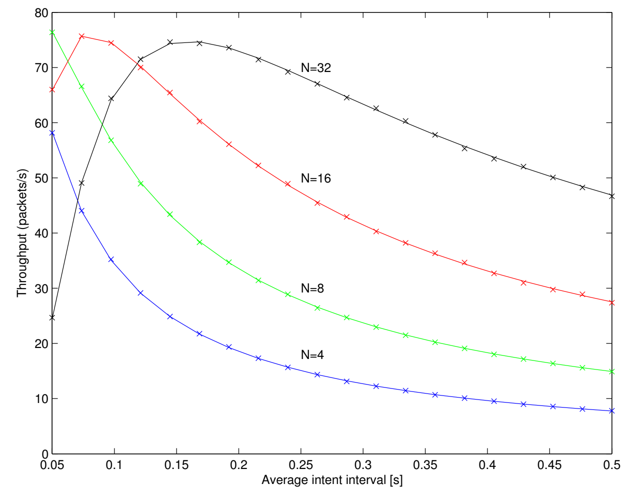

Fig. 4. Network throughput vs. mean intent interval for $K = 4$ available channels, $T_w = 10$ ms, uniform interval distribution. Crosses represent simulation outcomes. Note the different throughput scale.

VI. Simulation results

A. *Throughput and delay of single-hop ARS/CF*

We developed a discrete event simulation model for our proof-of-concept ARS/CF implementation. In these simulations we assume that the nodes exhibit identical protocol behavior and no packets get lost completely. At the beginning of each simulation round, we assume nodes obtain the time stamp and seed value from all other nodes, then construct their schedule and cancel transmissions that would collide. Each data point has been obtained from 100 independent simulation drops. We assume all the nodes belong to the same collision domain, i.e., there are no hidden nodes.

First we validate the single-hop throughput analysis presented in Section V. Fig. 3. shows the throughput, defined as the successfully transmitted over-the-air packets per unit time vs. the average intent interval for the single-channel case, if the nodes attempt the transmission according to the uniform inter-packet mapping of (2), depending on the number of users within the collision domain. Very good agreement can be observed between the throughput predicted by (6) and the simulated values even for small number of users and high traffic.

Figure 4. depicts the simulated throughput versus the average intent interval if $K = 4$ channels are available for simultaneous transmission. As expected, increasing the number of channels provides efficient capacity scaling.

Next, we assess the delay behavior of the basic ARS/CF scheme. The average time between successful transmissions is shown in Fig. 5. As intuitively expected, for low average packet intervals, throughput is low and therefore one node seldom gets to transmit. Increasing the average packet interval decreases the delay until an optimum is reached, beyond which transmission attempts tend to succeed often. The mean delay is insensitive to the choice of T_{min} and T_{max}.

These results point out an important future research direction, namely, to find efficient solutions within the ARS/CF framework to improve further the predictability in terms of inter-packet times. We should point out the fact that the proof-of-concept implementation does not try to resolve conflicting intents but simply all nodes abandon those transmission opportunities. These abandoned windows render the channel heavily underutilized. In contrast to contention-based protocols, where real collisions occur, these empty windows still can be exploited under a smart secondary collision resolution algorithm. Such a secondary scheduling algorithm can still be completely deterministic within the collision domain. Proper choice of this scheduling algorithm can ensure fairness over a shorter time horizon, for example, taking into account the time elapsed since the last transmission opportunity to grant conflicting windows would improve the above mentioned outliers in delay. Windows marked as abandoned could also be used for retransmission on a contention basis.

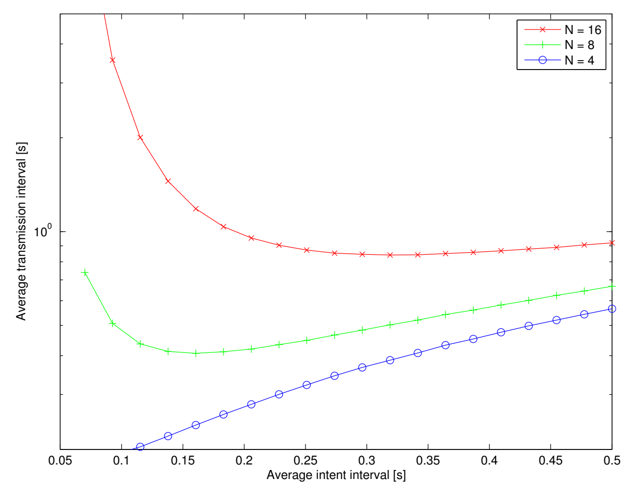

Fig. 5. Average delay between successful transmissions.

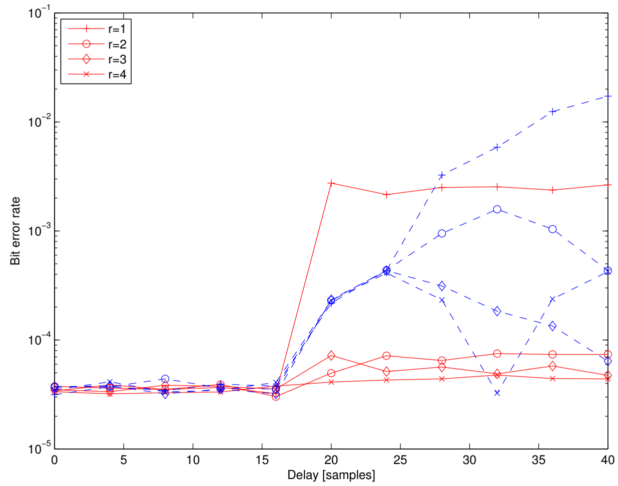

Fig. 6. Error rates for asynchronous transmissions over IFDMA/LFDMA. Solid lines: LFDMA, dashed lines: IFDMA

B. Multi-channel PHY

We also investigate the feasibility of using xFDMA to allow multiple users to transmit approximately simultaneously on orthogonal frequency resources, determined by their pseudo-random seed, as suggested in section IV-A. To our knowledge, performance of xFDMA under significant time offsets has not been widely investigated. We performed link-level evaluation of IFDMA and LFDMA with 64 subcarriers, of which 8 subcarriers are assigned to a single user. Hence, up to 8 users could use the channel at the same time. The nodes use a cyclic prefix that is 1/4 of the symbol duration, i.e., 16 symbols. We assume two nodes are transmitting simultaneously, with $r = 0$ assigned to the reference node, and one other node using $r = 1$ to $r = 4$, respectively. We plot the resulting bit error ratio of the reference node in Fig. 6 when its signal-to-noise ratio is 10 dB over the additive white Gaussian noise (AWGN) channel. Clearly, ensuring synchronicity between nodes such that the channel impulse response and the amount of asynchronism is smaller than the duration of the cyclic prefix results in no loss in communication performance, independent of the FDMA scheme or the r value of the other node. However, ARS/CF not requiring accurate network-wide time synchronization, it is imperative to investigate the possible loss due to asynchronism exceeding the length of the cyclic prefix. As we can see in Fig. 6, LFDMA copes with this kind of interference very well, except for the case when the interfering transmission is on the immediate neighboring set of subcarriers ($r = 1$). IFDMA, having its subcarriers interleaved, understandably exhibits moderate performance degradation. It is left for future work to study low-complexity blind interference cancellation schemes to improve IFDMA in this respect.

VII. Conclusions

We proposed a predictable, scalable wireless MAC protocol that can be adapted to the communication needs of time-critical sensor and control applications. We analyzed the throughput of both the single- and multi-channel proof-of-concept ARS/CF, and demonstrated its similarity with ALOHA, although pointing out several advantages in term of power consumption and effective channel occupancy compared to traditional contention-based protocols. We demonstrated the suitability of channel hopping and multiple-access modulation schemes, to be organically integrated with ARS/CF for capacity improvement. Analysis of the inter-packet arrival times revealed the need for secondary scheduling within ARS/CF to exploit the channel capacity which remains unused due to abandoned transmission intents.

As the single-hop proof-of-concept implementation has been successfully demonstrated the viability and feasibility of our proposed solution, the most important future direction is developing low-complexity extensions for multi-hop wireless operation that are robust against the imperfections of the wireless transmission and preserve the property of the original protocol of not requiring global clock synchronization.

Acknowledgement

This work is supported in part by the National Science Foundation (CNS-1035655), U.S. Army Research Office (ARO W911NF-10-1-0005) and Lockheed Martin. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

References

[1] J. Sallai, "Collision forecasting: A low-power MAC with traffic and power shaping," in *The 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, Agia Napa, Cyprus, July 2012, pp. 39–46.

[2] I. Chlamtac, A. Faragó, and H. Zhang, "Time-spread multiple-access (TSMA) protocols for multihop mobile radio networks," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 804–812, 1997.

[3] J.-H. Ju and V. O. K. Li, "An optimal topology-transparent scheduling method in multihop packet radio networks," *IEEE/ACM Trans. Netw.*, vol. 6, no. 3, pp. 298–306, 1998.

[4] U. Roedig, A. Barroso, and C. J. Sreenan, "f-MAC: A deterministic media access control protocol without time synchronization," in *Wireless Sensor Networks*. Springer Berlin Heidelberg, 2006, vol. 3868, pp. 276–291.

[5] R. Rozovsky and P. R. Kumar, "SEEDEX: a MAC protocol for ad hoc networks," in *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, ser. MobiHoc '01, New York, NY, USA, 2001, pp. 67–75.

[6] L. Bao and J. J. Garcia-Luna-Aceves, "A new approach to channel access scheduling for ad hoc networks," in *7th Annual International Conference on Mobile Computing and Networking*, New York, NY, USA, 2001, pp. 210–221.

[7] J. Garcia-Luna-Aceves and A. N. Masilamani, "Nomad: Deterministic collision-free channel access with channel reuse in wireless networks," in *8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, Salt Lake City, Utah, June 2011.

[8] H. G. Myung and D. J. Goodman, *Single Carrier FDMA*. John Wiley & Sons, 2008.

[9] T. Frank, A. Klein, and E. Costa, "IFDMA: A scheme combining the advantages of OFDMA and CDMA," *IEEE Wireless Commun. Mag.*, vol. 14, no. 3, pp. 9–17, 2007.

[10] D. Sant, "Throughput of unslotted ALOHA channels with arbitrary packet interarrival time distributionsr," *IEEE Trans. Commun.*, vol. 28, no. 8, pp. 1422–1425, 1980.