Improving the Reliability of WiDom in a Single Broadcast Domain

Nuno Pereira[1], Björn Andersson[1], Eduardo Tovar[1] and Paulo Carvalho[2]

[1]IPP Hurray Research Group - CISTER/ISEP
Polytechnic Institute of Porto, Portugal
{npereira, emt, bandersson}@dei.isep.ipp.pt

[2]Department of Informatics
University of Minho, Braga, Portugal
pmc@di.uminho.pt

Abstract

WiDom is a previously proposed prioritized medium access control protocol for wireless channels. We present a modification to this protocol in order to improve its reliability. This modification has similarities with cooperative relaying schemes, but, in our protocol, all nodes can relay a carrier wave. The preliminary evaluation shows that, under transmission errors, a significant reduction on the number of failed tournaments can be achieved.

1. Introduction

A MAC protocol that supports static priority scheduling for a wireless channel was recently proposed and dubbed WiDom [1]. This protocol allows for designers to apply the well-developed scheduling theory for CAN and for non-preemptive uniprocessor scheduling. But the usefulness of WiDom goes beyond real-time communication; it was recently identified how WiDom can be used for computing aggregated quantities efficiently [2]. For example, the maximum among sensor readings in a distributed system can be computed with a time-complexity that is dependent on the size of the value domain of sensor readings but remarkably, that time-complexity is independent of the number of sensor nodes. This is possible because (i) the sensor reading can be used as a priority and (ii) the MAC protocol elects the node with the highest priority.

We say that WiDom operates correctly if the node that is granted access to the wireless channel is the node with the highest priority among the nodes that requested to transmit. Previous work, which implemented and tested WiDom, has shown that it is possible to achieve correctness with 99.9% probability [1]. This figure of the reliability of correctness (99.9%) was obtained in a controlled lab environment with (i) nodes located close to each other in space and (ii) nodes were still. With the two already mentioned exciting properties of WiDom (the possibility of scheduling wireless traffic and computing aggregated quantities), we are motivated to also achieve good reliability of WiDom in order to use it also in environments that are harsher.

Therefore, in this paper, we propose a modification to WiDom in order to improve its reliability. The research literature offers different types of WiDom; in this paper only consider the simplest one where it is assumed that an external node sends synchronization pulses on an out-of-band channel in order to create timeslots for the computed nodes. One way to implement that is to use a sensor node platform that is equipped with an AM receiver that detects signals from a atomic clock. Such AM receivers are used in the FireFly sensor platform [3] and it can receive time-sync signals with a continental wide coverage. Transmitters for such signals are deployed in Europe and in the USA [4] and used for a variety of applications.

The main idea to improve reliability is to re-broadcast dominant bits; this concept has already been used by us in order to tackle the so-called hidden node problem [5]. Nonetheless, in this paper we show that the idea is also useful for improving the reliability of the protocol. To demonstrate this, we present preliminary simulation results.

The remainder of this paper is organized as follows. Section 2 provides a background on WiDom. Section 3 discusses the fault scenarios for which WiDom is most vulnerable. The findings from this discussion are used to improve the reliability of WiDom, reported in Section 4, where the improved version of the protocol and simulation results are presented. Finally, Section 5 summarizes the main discussions, conclusions and future work.

2. Background

The main idea of the WiDom protocol is that among all computer nodes that request to transmit on the channel, the one with the highest priority is selected by the protocol and this node will be granted the right to transmit on the wireless channel. The operation of WiDom is as follows. All nodes synchronize their clocks meaning that they have a time reference. After that, all nodes go through a contention resolution phase - called tournament in WiDom - such that after the contention resolution phase a node with the highest priority is declared as a winner; all other nodes are declared as losers. Then the winner node transmits its data packet. This procedure is repeated indefinitely.

Several versions of WiDom have been proposed. One version is for wireless networks in a single broadcast domain, meaning that every transmission reaches every other computer node [1]. Another version is designed for networks with multiple broadcast domains, that is, a single broadcast from a computer node does not necessarily reach all computer nodes [5]. These versions of WiDom used elaborate schemes for time

```
when an out-of-band synchronization pulse is received do:
    t := read_clock
    <pak,prio> := peek_highest_prio_packet
    winner := TRUE
    winner_prio := 0
    for j := 0 to npriobits-1 do
       if bit j in prio = 0 then
          send_carrier( t+j*(H+G)+H)
          set bit j in winner_prio to 0
       else
          if heard_carrier( t+j*(H+G)+H) then
             set bit j in winner_prio to 0
             winner := FALSE
          else
             set bit j in winner_prio to 1
          end if
       end if
       delay until t+j*(H+G)+H+G
  end for
  Proceed to send/receive, according to variable winner
```

```
procedure send_carrier(tstop_send : time) is
begin
   switch transceiver to TX mode
   transmit an unmodulated carrier wave
   delay until tstop_send
   stop transmitting an unmodulated carrier wave
end


function heard_carrier(tstop_sense: time) return boolean is
begin
   switch transceiver to RX mode
   while read_clock<tstop_sense do
      if a carrier was detected then
         return true
      end if
   end while
   return false
end
```

Figure 1. Baseline WiDom. *G* and *H* are constants that depend on the transceiver used.

synchronization in order to be able to operate without any other infrastructure. Specifically, they did not depend on any out-of-band receiver for receiving time-synchronization signals. Previous work [1, 5] stated the versions of WiDom in a precise way using timed automaton and taking many real-world effects into account, such as clock-drift, finite execution speed of the processor, detection a carrier pulses, transmit/receive switching time, etc. In order to simplify the discussion, we will, in the remainder of the paper, assume that (i) an external computer node transmits pulses in a separate frequency band, where each pulse indicates the beginning of a time slots, (ii) the duration of the timeslot is equal to the time it takes to run a tournament in the MAC protocol, and (iii) all nodes are in a single broadcast domain.

We will now briefly describe the protocol. Figure 1 presents a simple algorithm that describes the version of WiDom that we will use as a baseline in our discussion. Every computer node executes this pseudo-code, operating as follows. On each computer node, there is a queue, called output queue, which stores all packets to be transmitted. Each packet in the output queue has an associated priority. The packets are sorted in descending order of priority; that is the highest priority packet is in the head of the queue. The convention in CAN and WiDom is that low numbers represent high priorities. For example, the number zero corresponds to the highest priority.

A computer node waits until it receives an out-of-band pulse. Then it dequeues the highest priority packet from the output queue, obtaining the priority of this packet. The priority is represented as an integer and the computer node inspects each bit of this integer starting with the most significant bit and ending with the least significant bit. If the bit under inspection is “0” then we say that the bit is dominant; if the bit is “1” then we say that the bit is recessive. In the presence of a dominant bit, then the computer node switches the transceiver to transmission mode and transmits an unmodulated carrier wave for a fixed duration. If the bit is recessive then the computer node switches the transceiver to reception mode and it performs carrier sensing for a fixed duration. During this carrier sensing, the computer node may detect that an unmodulated carrier was transmitted or it may detect that no unmodulated carrier was transmitted.

A computer node has a variable `winner` which is initialized to true before the most significant bit is inspected. Whenever a computer node has a recessive bit and detects an unmodulated carrier it sets its variable `winner` to false. After the least-significant bit has been inspected, the protocol switches the transceiver to transmission mode and transmits the packet, only if `winner` is true; otherwise the protocol switches the transceiver to reception mode and receives the packet. Then the WiDom protocol waits until a new out-of-band pulse is received. As noted earlier, we assume that this out-of-band pulse is transmitted periodically, with a period defined in function of the time it takes to run a tournament.

There are several time constants that need to be selected in order for this protocol to work. They follow from the same spirit as in previous work [1, 5]. There are two parameters in the configuration of the transceiver that have an impact on our future discussion though. When the transceiver performs carrier sensing, it measures the amount of energy in the frequency band being used and compares this energy to a threshold. If the measured energy is above this threshold then it is perceived as an unmodulated carrier wave; otherwise it is considered as no unmodulated carrier wave. We assume that this threshold is set sufficiently high so that whenever no computer nodes send a packet, it holds that no transceiver declares that it has detected an unmodulated carrier. This was done in previous work [1]. Note that frequently it is possible to trade effective communication range for a lower false positive probability, by raising the detection threshold.

There is also a threshold for detecting the out-of-band pulse that indicates the beginning of a new time slot. If the energy in the out-of-band pulse exceeds the threshold then it triggers the algorithm in Figure 1. We will assume that the threshold is sufficiently high so that, at those moments, when the external node sends no out-of-band signal, it holds that no computer nodes perceives that an out-of-band signal is received.

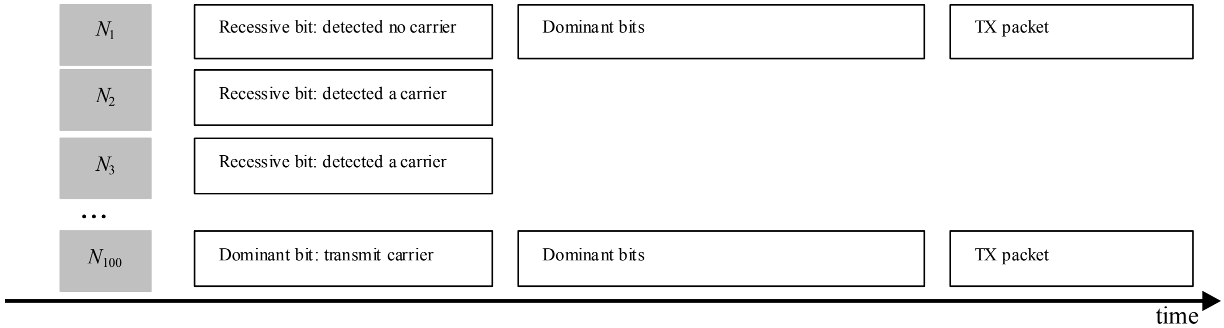

Figure 2. A scenario which illustrates the vulnerability of baseline WiDom. The system comprises 100 computer nodes in a single broadcast domain and their location is as shown to the left. In the most significant bit, node N_{100} is dominant and the other nodes are recessive. The node N_{100} broadcasts a carrier wave and all other nodes are within broadcast range. There is a small probability that a node does not detect this carrier though; in this particular scenario it happened that node N_1 did not detect the carrier wave. Consequently, the variable winner at node N_1 is true even after the most significant bit. The remaining priority bits of N_1 and N_{100} are all dominant so at the end of the conflict resolution phase, both N_1 and N_{100} have winner equal to true. Therefore, both of them transmit and they collide. In this scenario, WiDom did not operate correctly.

3. Vulnerability

Let us consider the following four possible fault-scenarios: (i) an out-of-band signal was transmitted but there was a computer node which did not perceive it; (ii) no out-of-band signal was transmitted but there was a computer node which perceived an out-of-band signal; (iii) a carrier wave was transmitted by one of the computer nodes but there was another computer node which did not perceive this unmodulated carrier wave; and (iv) no carrier wave was transmitted but one computer node perceived an unmodulated carrier wave.

Attending to the configuration assumed in Section 2, the fault scenarios (ii) and (iv) cannot occur. Let us assume that fault-scenario (i) occurs. Then the consequence is that the computer node which did not perceive this unmodulated carrier wave will not participate in the contention resolution phase and hence holds that this node has `winner`=false. If the computer network is large there will be computer nodes in every contention that act in this way. Consider for example 100 computer nodes and a probability of unheard carrier to be 10^{-4} (which is reasonable considering the probabilities gathered in [1] on experiments on normal carrier sensing with inband carriers) then it follows that in each slot we can expect that approximately 1 computer node every 100 tournaments will not participate in the contention. This behavior is undesirable of course because its effect is similar to priority-inversion in uniprocessor scheduling. But the network as a whole will continue to make progress in the sense that one computer node will transmit and this transmission will be collision-free.

```
when an out-of-band synchronization pulse is received do:
   t := read_clock
   <pak,prio> := peek_highest_prio_packet
   winner := TRUE
   winner_prio := 0
   for j := 0 to npriobits-1 do
      if bit j in prio = 0 then
         send_carrier( t+j*(2*H+G)+H)
         dominant_exist := TRUE
         set bit j in winner_prio to 0
      else
         if heard_carrier( t+j*2*H+H) then
            dominant_exist := TRUE
            set bit j in winner_prio to 0
            winner := FALSE
         else
            dominant_exist := FALSE
            set bit j in winner_prio to 1
         end if
      end if
      if dominant_exist then
         send_carrier( t+j*(2*H+G)+2*H)
      else
         if heard_carrier( t+j*2*H+2*H) then
            dominant_exist := TRUE
            set bit j in winner_prio to 0
            winner := FALSE
         else
            dominant_exist := FALSE
            set bit j in winner_prio to 1
         end if
         delay until t+j*(2*H+G)+2*H+G
      end if
   end for
```

Figure 3. WiDom with reliability improvement. *G* and *H* are constants that depend on the transceiver used.

Scenario (iii) is even more adverse however. As we can observe from the scenario depicted in Figure 2, as the number of nodes increase, it becomes increasingly likely that at least one node does not detect a transmitted carrier.

4. The New Protocol

From the previous discussion, it follows that one important vulnerability of WiDom is the scenario where a node transmits a carrier wave and another node does not perceive this carrier wave. If the two nodes are closely located then the signal strength from the transmitted carrier wave at the receiving node will be large and hence it is very unlikely that the receiving node will not perceive the carrier. We can modify the WiDom protocol slightly to achieve this.

Figure 3 shows the new protocol. It operates in a similar fashion as the protocol in Figure 1 but with one exception. Every priority bit is represented by two time intervals. During the first of these time intervals, nodes operates as in Figure 1 but in the later part of the time interval, a computer node transmits a carrier if it heard a carrier earlier in the time interval of this bit. In the example in Figure 2, it implies that when node N_2 perceives a carrier wave in the most significant bit then N_2 will broadcast a carrier in the later step of the time interval for the most significant bit. Since N_2 is close to N_1, it holds that N_1 will detect this carrier and set `winner` to false.

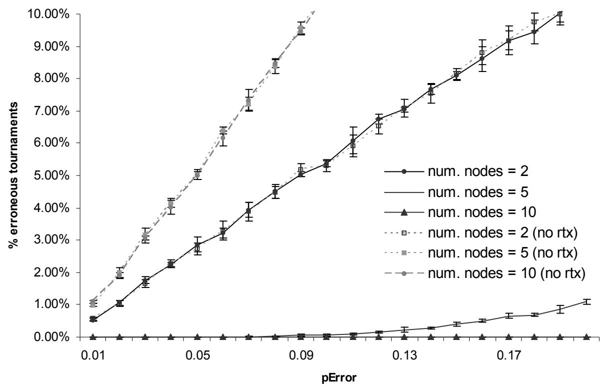

Figure 4. Simulation Results.

4.1. Experimental Evaluation

We have implemented a simulator to experimentally test how the new protocol would perform under different carrier detection failure rates. Both the previous version of the protocol and the modified one were implemented, which also allowed to us comparing their performances.

The simulations were performed using 10 priority bits during the tournament. We tested the protocol by varying probabilities of missing the detection of a carrier pulse and the number of nodes. For each scenario, 10 independent simulation runs were executed. Each node was setup with one message stream having a unique priority and an exponentially distributed inter-arrival time, with an expected value ranging between 0.01 and 1 second.

In all simulation runs, nodes perform more than 10000 tournaments. After each tournament, we detected whether the correctness properties collision-free, progress and prioritization were satisfied for all nodes in the network. Tournaments where any node in the network failed to satisfy one of the properties are named erroneous tournaments. These erroneous tournaments were caused by failure to detect a priority bit. The number of erroneous tournaments observed is plotted in Figure 4, were the previous version of WiDom is identified as "*no rtx*".

The experiments show that with the previous version, the number of failed tournaments increases very rapidly. This increase continues the more nodes we have in the network. Conversely, with the new protocol, as the number of nodes increases, we can observe that the new protocol performs markedly better. This is easily explained by the fact that, as more nodes exist in the network, it becomes more probable that a receiving node retransmits a dominant bit, previously not detected. We can also see that, for a network with 2 nodes, both the new protocol version and the previous one perform similarly. This is because, in this case, only one node is receiving at each time, thus the new protocol is basically reduced to the previous one, as retransmissions are not effective.

5. Discussion, Conclusions and Future work

We have shown how to improve the reliability of WiDom. The main idea is that a node which has perceived a carrier wave broadcasts a carrier wave in order to ensure that other nodes will perceive it with better reliability.

This improvement clearly helps the normal operation of WiDom where computer nodes contend for the channel and then the winning node transmits a packet. But it also contributes to achieve reliable communication of data bits. Consider for example a real-time system which monitors an important event (say the crash of a car or the outbreak of fire). A computer node that detects this event can request to transmit a packet with a given priority but then if it wins it does not send any data packet. The fact that the node wins the priority implies that the event has occurred. Since the information of the event is conveyed in the priority, we obtain the advantage that our new protocol will use other nodes to convey this information more reliably. This modification of the protocol decreases the throughput of the protocol by 50% (if both the tournament and the data are transmitted in the same way), but the protocol is still amenable to pre-runtime scheduling analysis.

This new protocol has similarities with cooperative relaying schemes [6, 7] which are typically employed to improve resource utilization. In our protocol however, it is not necessary to select the relaying node(s); all nodes can relay a carrier wave because the simultaneous transmission of two unmodulated carrier waves pose no problem.

References

[1] N. Pereira, B. Andersson, and E. Tovar, "WiDom: A Dominance Protocol for Wireless Medium Access", IEEE Transactions on Industrial Informatics, vol. 3, issue 2, pp. 120-130, 2007.

[2] B. Andersson, N. Pereira, E. Tovar, F. Pacheco, and N. Cruz, "A Scalable and Efficient Approach for Obtaining Measurements in CAN-Based Control Systems", IEEE Transactions on Industrial Informatics, vol. 4, issue 2, pp. 80-91, 2008.

[3] CMU, "FireFly-Real-Time Wireless Sensor Network Platform": Carnegie-Mellon University; Real-Time & Multimedia Systems Lab, 2007, online at: http://www.ece.cmu.edu/firefly/.

[4] "NIST Radio Station WWVB", http://tf.nist.gov/stations/wwvb.htm.

[5] N. Pereira, A. Rowe, B. Andersson, and E. Tovar, "Static-Priority Scheduling over Wireless Networks with Multiple Broadcast Domains", In proc. of the 28th IEEE Real-Time Systems Symposium (RTSS'07), pp. 447-458, Tucson, Arizona, USA, 2007.

[6] A. Wittneben, I. Hammerstroem, and M. Kuhn, "Joint cooperativediversity and scheduling in low mobility wireless networks," in Proc.Globecom, pp. 780–784, 2004.

[7] A. E. Khandani, J. Abounadi, E. Modiano, and L. Zheng,"Cooperative routing in wireless networks," in Proc. Allerton Conf.on Commun.,Control and Computing, 2003.