PH-MAC: A Periodically Hybrid MAC Protocol for Wireless Sensor Networks

Liansheng Tan [1], Jintao Meng [2], Jie Li [3], Han-Chieh Chao [4]
[1] The Research School of Information Sciences and Engineering, The Australian National University, Australia
[2] Department of Computer Science,Central China Normal University, China
[3] Department of ComputerScience, Graduate School of Systems and Information Engineering, University of Tsukuba, Japan
[4] Electronic Engineering Department & CSIE Institute, College of Electrical Engineering & Computer Science National Ilan University, Taiwan, R..O.C.
liansheng.tan@raise.anu.edu.au, meng.jintao@gmail.com, lijie@cs.tsukuba.ac.jp, hcc@niu.edu.tw

Abstract

This paper proposes a novel protocol termed periodically hybrid MAC protocol (PH-MAC) for wireless sensor networks. In order to save energy and maximize the throughput we introduce the channel access mechanism TDMA into the PH-MAC. PH-MAC divides the whole transmission process into various PH-MAC periods, and each PH-MAC period is further divided into synchronization period, allocation period and TDMA period. The synchronization period is used to synchronize the neighboring nodes to prevent long-time clock drift; the allocation period is used to allocate timeslots to every node for communication; and the TDMA period is used for data communication. We present the designing techniques of PH-MAC, and compare the performance of PH-MAC with the simplified 802.11 DCF (no duty cycle), S-MAC (fixed duty cycle) and T-MAC (dynamic duty cycle) through simulation. The simulation results show that, PH-MAC not only improve the throughput of the network, but also save nearly half of the energy that T-MAC usually consumes.

Keywords: Sensor networks, MAC, TDMA, PH-MAC.

1 Introduction

For wireless sensor networks, energy is usually wasted in the processes of collision, overhearing, control packet overhead and idle listening. In this paper, we proposed a novel protocol termed periodically hybrid MAC protocol (PH-MAC). PH-MAC is designed to minimize the waste of energy in all the above processes. Besides energy efficiency, through PH-MAC, we are able to guarantee throughput and bandwidth utilization by allocating timeslots effectively in the contention period. PH-MAC proposes a new architecture of MAC protocol for data communication. PH-MAC is contention-based at allocation period and TDMA-based at the TDMA period. It uses contention period to allocate the timeslot to every node that needs to transmit data, and the TDMA part to transmit actual data. For every node at TDMA period, it just needs to transmit data or receive data when the incoming data coming or outgoing data outgoing. What we want to do is to get the boring contention work done once and forever, though we could not keep it forever, just a moment (period) is enough for us. As we also need to deal with network changes and clock drift, so the PH-MAC runs periodically to adjust to those changes online. Under this architecture we get rid of contention, overhearing, idle listening and control packet overhead at the TDMA part, which could save much energy for the network.

Clearly, the energy used in the contention part is all wasted. So prolonging the TDMA part will save more energy, while the whole period will be prolonged, this will reduce the latency of the network. So we need to tradeoff latency with energy consumption.

The current MAC [1-4], [9], [10], which is designed for wireless sensor networks, can be broadly divided into contention-based and slotted protocol [4], [8]. Though a contention-based protocol is promising in terms of its simplicity, flexibility and robustness, It will suffer from contention, idle listening, overhearing, and this will waste lots of energy; while a slotted based protocol have no collisions, it requires an efficient time schedule, clock synchronization and adjusting to network topology changes.

The standardized IEEE 802.11 distributed coordination function (DCF) [1] is an example of the contention-based protocol, and is mainly built on the research protocol MACAW [2]. It is widely used in ad hoc wireless networks because of its simplicity and robustness to the hidden terminal problem. However the energy consumption in the wireless sensor network used this MAC is very high.

In order to reduce the waste of energy from collision, overhearing, idle listening and control packet overhead, S-MAC [3] induces a duty cycle into the 802.11, and use virtual clustering, message passing, collision avoidance, overhearing avoidance to save more energy. As this MAC protocol must predetermine the length of active part, which could not adjust to the network with variable load. So T-MAC [4] was proposed to dynamically adjust to the variable load by inducing the TA threshold, which determines the minimal amount of idle listening per frame. This could minimize the idle listening and also dynamically adjust to the variable load.

We present PH-MAC, a periodically hybrid energy efficient MAC protocol for wireless sensor networks that minimizes energy waste. To demonstrate the effectiveness and measure the performance of our new MAC protocol, we have compared PH-MAC with other three existing MAC protocols (simplified 802.11 DCF, S-MAC and T-MAC) on network throughput, energy consumption and latency. Three simulators are designed for each of these protocols using C++.

2 Design of PH-MAC

Our MAC protocol is designed to reduce energy consumption and adjust to network changes and variable network load. This protocol is designed to reduce all energy consumption from all the sources that we have mentioned, i.e., idle listening, collision, overhearing and control overhead. In order to achieve the design goal, we have developed the MAC that consists of three major periods: synchronization period, allocation period and TDMA period. Before describing them we need to discuss our assumptions about the wireless sensor network and its applications.

2.1 Network Assumptions

Sensor networks are consisted of many small sensors and deployed in an ad hoc fashion. We expect all the sensors are the same, which means that all sensors have equal bandwidth, process capability and energy.

All sensor networks take advantage of short-range multi-hop communication instead of long-range communication to conserve energy [5]. Most communication will be between nodes as peers, rather than to a single base-station or sink node. So routing algorithm must be used to collect information in the network layer, i.e., directed diffusion [11], LEACH [12] and GAF [6]. As some sensors will fail when it is working, the routing algorithm must be robustness to the failure. Some sensors will move slowing along with the environment, i.e., sensors deployed on the sea, the routing algorithm could self-adjusting to the network topology changes periodically by inducing the periodical MAC protocol. We expect that the routing algorithm at the network layer will construct a routing tree to collect data. This routing tree will let every node knows its parent node which the data must be sent to and its child nodes which the data must be received from. This is what we need in the MAC design.

2.2 The Scheme of PH-MAC

The basic scheme of PH-MAC is shown in Figure 1, in this figure, PH-MAC is divided into three periods, they are synchronization period, allocation period, and TDMA period.

In section III-A, we have assumed every node will know its parent node and its child nodes once the routing tree is constructed. In the synchronization period, all nodes in the network maintain the same schedule by communicating with SYNC packets; in the allocation period, each node requests timeslots from his parent node for sending data, and allocate timeslots to his child node for receiving data; in the TDMA period, every node transmit packets in the timeslots that its parent node allocate to it, and receive data from its child nodes in the timeslots that it allocates to its child nodes.

2.3 Synchronization Period

In the synchronization period, the main task of the network is to synchronize the clock of all active nodes in the network, so as to minimize the effect of clock drift.

Clock synchronization is referred to virtual clustering, which is proposed by the authors of the S-MAC protocol [3]. While in our protocol, only the sink node who is the root node in the routing tree can start to send the SYNC packet at the start of one period, the other node in the network hears SYNC from another node, then follows the schedule in the SYNC packet and transmits its own SYNC packet. This will let all nodes in the network follow the sink's schedule. Any node that has not heard a SYNC packet must not transmit any data in this period, and this node will wait for the next synchronization period to receive the SYNC packet.

2.4 Allocation Period

In allocation period, the main work of the network is to allocate timeslots to every node who need to transmit data. In this period all nodes must contention for the channel to transmit packets, whereas there are two differences compared to 802.11 DCF:

1) The payload of allocation packet is not the data that the node needs to transmit, but the bitmap of the node's TDMA part;

2) All the packets follow the sequence of RTS/CTS/RTA/ALC/NOT, not the sequence of RTS/CTS/DATA/ACK as in the 802.11 DCF.

Every node has a bitmap, which will be used to describe local channel usage of TDMA period, every bit in the bitmap was mapped to a timeslot of TDMA period, that is to say, the first bit in the bitmap means whether the first timeslot in the TDMA period has been used by any node include himself. So at begin of every period, all the timeslots are not used by any node, all bits in every node's bitmap should be zero.

Figure 2 shows the time schedule of RTS/CTS/RTA/ALC/NOT which is used in the allocation part to allocate timeslot.

We adopt the RTS/CTS mechanism to address the hidden terminal problem [3], what's more, by inducing network allocation vector (NAV) [1] we can reserve the channel to transmit the incoming allocation packets, e.g. RTA, ALC, NOT. After occupying the channel by using RTS/CTS mechanism, the node that has packets to transmit, needs to load his current bitmap state and the number of needed timeslots into the payload of the request to allocate (RTA) packet, and send it to the destination node. When the destination node receives the RTA packet, it gets the bitmap of the source node, by Boolean or the source node's bitmap and the destination node's bitmap together, the destination node get the bitmap of all available timeslots, then the destination node can allocate some of the available timeslots to the source node, and load the corresponding bitmap and the number of allocated timeslots to the allocation (ALC) packet. After received the ALC packet, the source node must first update its bitmap by Boolean or its bitmap with the bitmap in the ALC packet and then load the bitmap into the notice (NOT) packet, and broadcast the NOT packet to let its neighbors know that it has occupied those timeslots. If any neighbor of source and destination nodes has received RTA, ALC or NOT packet which is not routing to himself, it must update its bit-map by Boolean or its bitmap with the bitmap in the packet. The above procedure is given in Algorithm 1 as a pseudocode.

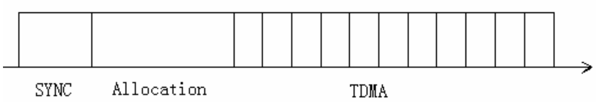

Figure 1 Scheme of PH-MAC: SYNC period, Allocation period, and TDMA period

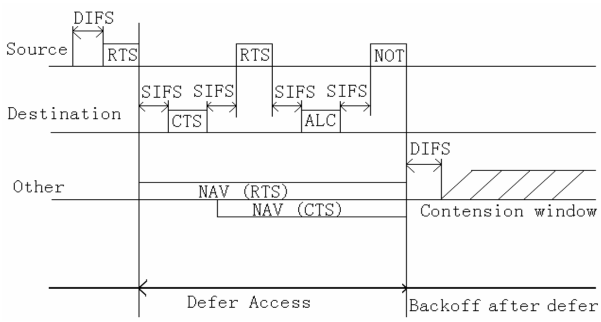

Figure 2 RTS/CTS/TC/ALC/NOT and NAV setting of the allocation period of PH-MAC

Algorithm 1 Distributed allocation algorithm used in the allocation period. Every node in the network will maintain a bitmap and node state.

```
If the channel is idle, then
    If node have packets to send and have not got
    any timeslot allocated then
        If node state is 0
            Send out a RTS packet
            node state ← 1, and set a timeout timer;
        end if
    end if
    If timeout occurred then
        node state ← 0
    end if
If the channel is not idle, then
    If node get a packet suffered collision then
        Discard this packet.
    end if
    If node get a packet correctly then
        if packet is not addressed to this node then
            if this packet is RTS or CTS then
                Update NAV to reserve channel.
            end if
            if this packet is RTA or ALC or NOT then
                Update the node's bitmap by Boolean
                or the bitmap in the packet and the
                node's bitmap.
            end if
        if packet is addressed to this node then
            if this packet is RTS and node state = 0 then
                Send out CTS to reply
                node state = 2, and set a timeout
                timer.
            end if
            if this packet is CLS and node state = 1 then
                Load the bitmap and the number of
                needed timeslots into the RTA packet,
                and send it.
                node state ← 3, and set a timeout
                timer.
            end if
            If packet is RTA packet and node state=2 then
                Boolean or the bitmap in the RTA packet
                and the node's bitmap, this node will get
                the available bitmap could be used to
                allocate.
                According to the available bitmap, this
                node will try to allocate the needed
                timeslots to the source node.
                Load the allocated timeslots to ALC
                packet
                Send out the ALC packet
                node state ← 0, and set a timeout timer.
            end if
            If the packet is ALC packet and node state=
                3then
                Load the allocated bitmap in the ALC packet
                into the NOT packet, and send it out.
                node state ← 0;
            end if
        end if
end if
```

One example of such schedule is shown in Figure 3. In Figure 3, (a) shows all nodes' initial bitmap; (b) shows the source node are sending a RTA packet to the