

# 无线传感器网络自适应 MAC 协议

范清峰, 张瑞华

FAN Qing-feng, ZHANG Rui-hua

山东大学 计算机科学与技术学院, 济南 250101

School of Computer Science and Technology, Shandong University, Jinan 250101, China

E-mail: ifqf@mail.sdu.edu.cn

FAN Qing-feng, ZHANG Rui-hua. Adaptive MAC protocol for wireless sensor networks. *Computer Engineering and Applications*, 2010, 46(3): 76-78.

**Abstract:** This paper proposes a new MAC protocol, ATMAC which can be adaptive to the traffic load. Based on the TMAC protocol, aimed to low energy consumption, low latency, several scheme is used to make nodes sleep more when their traffic is lower, while be awake more when their traffic is high, such as adaptive duty-cycle, adaptive contention window, and privileged data transfer queue. Moreover, the duty-cycle of nodes refer to in low and high traffic can be different even they are in one virtual cluster. So the energy consumption of idle listening and latency can be reduced while increasing the throughput. The simulation results with ns-2 show that ATMAC outperform TMAC in energy consumption and latency.

**Key words:** MAC; low energy consumption; low latency; duty-cycle; wireless sensor networks

**摘要:**提出了一种根据无线传感网络流量自动调节节点睡眠-活动时间比例的 MAC 协议-ATMAC, 在无线传感器网络 TMAC 协议的基础上, 以低能耗、低延迟为目标, 主要采用自适应、多级别的占空比及自适应竞争窗口, 数据优先级队列使节点在流量较小时能更多地处于睡眠状态以节省能量, 而在流量较大时, 传输所涉及的节点可相对长时间地进入活动状态, 且大流量和小流量节点所采取的占空比可以不同, 从而节省低流量节点用于空闲侦听的能耗, 降低数据传输的延迟, 增大网络的吞吐量。仿真结果显示新协议在能量消耗、数据延迟等方面要超过 TMAC。

**关键词:** MAC 协议; 低能耗; 低延迟; 占空比; 无线传感器网络

DOI: 10.3778/j.issn.1002-8331.2010.03.023 文章编号: 1002-8331(2010)03-0076-03 文献标识码: A 中图分类号: TP393

## 1 引言

随着通信技术、嵌入式计算技术和传感器技术的飞速发展和日益成熟, 具有感知、计算和通信能力的微型传感器节点开始在世界范围内出现。由于传感器节点电池供电, 处理能力、存储能力和通讯能力等都十分有限。这决定了无线传感器网络在设计时要以能量效率为首要目标, 尽量延长网络的寿命。

无线传感器网络 MAC 协议是保证无线传感器网络高效通信的关键网络协议之一。MAC 协议为了减少能量消耗, 通常采用“侦听/睡眠”交替的无线信道使用策略。当有数据收发时, 节点开启无线通讯模块, 使节点进入发送或接收状态, 如果没有数据则关闭无线通讯模块进入睡眠状态以节省能量。但如果没有一定的协调机制, 节点在邻居节点发送给自己数据时处于睡眠状态, 则会错过接收数据, 造成数据的丢失。所以, 邻居节点必须采用协调的侦听/睡眠周期, 同时睡眠或唤醒。根据文献[1] MAC 层能量浪费有四个方面: 碰撞、空闲侦听、串音、控制信息相关能量消耗。目前, 提出的大多 MAC 协议都考虑减少额外能量消耗, 经典的协议如 SMAC (Sensor-MAC)<sup>[2-3]</sup>、TMAC (Time-out-MAC)<sup>[4]</sup>、PMAC (Pattern-MAC)<sup>[5]</sup>等。

SMAC<sup>[2-3]</sup>协议是在 802.11MAC 协议<sup>[1]</sup>的基础上, 针对无线

传感器网络能量优先考虑的特点而设计的, 主要目标是减少能耗, 提高良好的扩展性。它主要采用了以下机制来避免能量的浪费: 周期性侦听/睡眠的低占空比工作方式、流量自适应侦听、串音避免以及消息分片传递。虽然 SMAC<sup>[2-3]</sup>在能量节省方面做了很多工作, 但仍有问题需要解决, 如它忽视了无线传感器网络流量变化较大的事实, 采用了固定大小的占空比, 这在网络流量变化较大时, 并不能达到很好的效果。占空比设置的大小会影响到能量节省和数据延迟的大小, 如何确定合适的占空比是这篇文章将要解决的问题之一。

TMAC<sup>[4]</sup>是在 SMAC<sup>[2-3]</sup>的基础上提出了一种自适应调整活期长度的算法。其调整的准则是: 只要有邻节点处于激活状态, 本节点就不进入休眠期。在该准则下它定义了几种延长激活期的事件, 休眠状态下, 节点在每个时隙的起始时刻激活, 每发生一次延长激活期的事件, 节点就将激活期延长一个 TA 时间。反之如果在 TA 时间内没有发生任何一个延长激活期的事件, 节点则转入休眠。其调整活期期的目的是尽量提高活期期的利用率, 在一个时隙中传输更多分组, 并在没有数据收发时及时转入休眠。然而它付出了侦听其他节点通信和时延的代价, 也就是说一对节点通信时, 它们的邻居都将延长激活期, 但

作者简介: 范清峰, 男, 研究生, 主要研究方向: 无线传感器网络; 张瑞华, 女, 博士, 硕士生导师, 主要研究方向: 无线传感器网络、嵌入式系统。

收稿日期: 2008-07-28 修回日期: 2008-10-21

可能只有这些节点中的小部分节点需要参加数据转发。而监听范围之外需要参加数据转发的节点无法获得延长激活期的信息,可能会提前转入休眠,也会出现由于数据转发中断带来的时延。文献[4]增加 FRTS 防止了这种提早休眠现象的出现,但也只能在一跳范围内改善转发。

PMAC 也是一种自适应调度协议,它引入了模式(pattern)的概念,节点可以根据自身和邻居节点的流量产生模式,并在此基础上改变自己的调度。与 SMAC 相比,在负载较轻时,PMAC 具有较好的节能性,而负载较大时,PMAC 能获得较大的吞吐量。

## 2 ATMAC 协议描述

这一章将给出 ATMAC 协议描述细节,在实际环境中,无线传感器网络流量是多变的,特别是在生态环境监测中,网络中长时间无事件发生,一旦有数据,数据大量猝发产生。

在提出的协议中,节点所采用的占空比是根据网络流量决定的,当流量比较小时,节点采取低占空比,流量比较大时,节点采取高占空比。除此之外,还对竞争窗口大小进行了改善。在节点收到数据并需要转发,而其自身也有数据需要传送时,采取优先传递中转数据的原则。

### 2.1 自适应占空比

在提出的协议中,采取占空比随流量大小改变的机制,协议中节点所采取的占空比如图 1 所示。TMAC 在每个唤醒周期内动态改变唤醒时间的长度,在 TMAC 的基础上,加入动态调节唤醒周期间隔的算法。算法中,节点判断自身待发送数据队列中数据的个数  $ContfQueue$ ,在每个唤醒周期结束前,根据变量  $ContfQueue$  的大小判断通信密度的大小,并会影响下个唤醒周期存在与否的选择。根据  $ContfQueue$  的大小设置该周期内通信量密度  $Com\_Density$ 。最近几个周期内通信量密度的大小  $Com\_Density\_avg$  决定下个阶段单位时间内周期的个数。这里  $Com\_Density$  的取值为 000,001,011,111 四种。000 代表通信密度为 0,001 代表通信密度为低,011 和 111 分别代表通信密度为中和高(这里预定义  $ContfQueue$  的几个门限值  $ContfQueueTh\_low$ ,  $ContfQueueTh\_mid$ ,  $ContfQueueTh\_high$  在 TCL 脚本中给出接口  $setContfQueueTh\_low$ ,  $setContfQueueTh\_mid$  和  $setContfQueueTh\_high$ )。

算法描述如下:

算法 1 (计算一个周期内通信密度)

```

If (ContfQueue>0
    and ContfQueue<ContfQueueTh_mid)
    Com_Density=001;
Elseif (ContfQueue>ContfQueueTh_mid
    And ContfQueue<ContfQueueTh_high)
    Com_Density=011;
Elseif (ContfQueue>ContfQueueTh_high)
    Com_Density=111;
Else
    Com_Density=000;

```

算法 2 (计算最近周期内通信密度)

$Com\_Density\_sum$ =取最近几个周期内通信密度相加(如 4 个周期, 011+111+001+110)

$num\_1=Com\_Density\_sum$  中 1 的个数

$Com\_Density\_avg=num\_1/(4*3)$ ;

算法 3 (计算下个阶段单位时间内周期的个数)

```

If(Com_Density_avg=1)

```

下个唤醒周期前增加一个唤醒周期

```

Elseif(Com_Density_avg>0.75)

```

保持唤醒周期状态

```

Else

```

去掉下个唤醒周期

可以看出,  $Com\_Density$  只是记录每个周期通信量,间接影响决定下个阶段单位时间内唤醒周期个数的变量  $Com\_Density\_avg$ 。

所以,在 ATMAC 协议中,节点唤醒周期及状态可如图 1 所示。

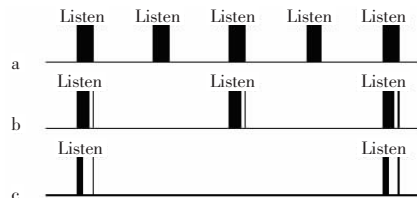


图 1 ATMAC 中节点占空比

如图 1 中 b、c 节点所示,在 ATMAC 中,当无数据或数据量很小时,节点沿用 TMAC 的机制快速进入睡眠,与 TMAC 不同,将节点的唤醒周期延长(算法 1~3)。这样节点不仅减少了一个唤醒周期内的能量,而且减少了唤醒周期的个数、唤醒-睡眠状态转换的能量消耗。

### 2.2 自适应的竞争窗口

竞争窗口的大小会影响通信的质量,特别是在多跳的网络中。竞争窗口选择过大时,数据在每个节点所产生的时延都会增大,竞争窗口过小时,数据碰撞的几率也将增大。所以,需要一个兼顾剩余能量并随流量变化自动选取的竞争窗口。

在 SMAC 中,节点所采用的竞争窗口  $CW$  大小为固定值(如实验中数据  $SYNC\_CW=31$ ,  $DATA\_CW=63$ ),这同样不能适应流量不断变化的网络环境。在新协议中,在文献[7]的基础上考虑节点剩余能量,采取如下算法确定竞争窗口的大小。加入一个实体  $times\_of\_collision$ ,记录最近时间内节点碰撞次数。可将竞争窗口分为几个等级  $lv1, lv2, \dots, lvn$ ,网络初始化时,竞争窗口设置为一个较小值,在碰撞次数增加到一定值时(或节点剩余能量  $RemainEnergy$  减小到一定程度),将竞争窗口加大一倍,减少碰撞(或减少节点参与通信的几率,从而延长节点和整个网络的生命周期)。在流量降低时,竞争窗口并不立即减小一倍,而是判断  $times\_of\_collision$  是不是减小一定程度,这样避免瞬间碰撞次数减小引起的错误,导致碰撞窗口减小,而使碰撞次数再次加大,从而有效避免二次冲突的产生。具体描述如下:

```

If (times_of_collision>threshold1 && RemainEnergy>energylv1 &&
cwlv! =lv1)

```

```

{
    CW=min((CW+1)*2, CW_max);
    cwlv=lv1;
}

```

```

Elseif (times_of_collision>threshold2 && RemainEnergy>energylv2
&& cwlv<lv2)

```

```

{
    CW=(CW+1)*2;
    cwlv=lv2;
}

```

```

...

```

```

Else

```

```

{
    CW=max((CW+1)/2,CW_min)
    cwlv=lv_n
}
    
```

### 2.3 数据优先级队列

在节点发送队列中,可以分为两类数据,一个是节点自身产生的数据,另一个是节点需要转发的数据。针对两种不同的数据,将发送队列中数据定义两个发送优先级,需要转发的数据包具有较高的优先调度权,本地产生的数据需要等待转发数据包结束后再进行发送,这样既可以保证数据报传送的成功率,又可以减少一个数据包传输中的延迟。

### 3 协议仿真

在这部分中,将给出新协议的能量和延迟仿真结果。使用现行的开源网络仿真软件 ns2,对新协议及 SMAC、TMAC 进行仿真比较,并得出数据结果。实验中所采用的两种网络拓扑结构,一种比较简单,为线形拓扑结构如图 3 所示;另外一种较复杂为网状拓扑结构,如图 4 所示。

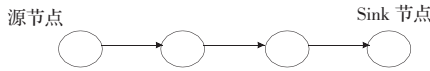


图 2 实验中采用的线形网络拓扑结构图

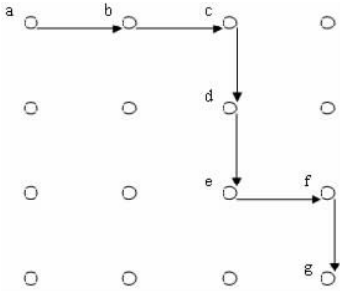


图 3 实验中采取的网状网络拓扑结构图

实验中采用 CBR 作为数据发生器,通过改变其发送间隔来改变其发送速率。把 SMAC 和 TMAC 的占空比设置为 10%。对于 ATMAC 节点,在脚本中,加入设置节点唤醒周期间隔最大值和最小值的接口, setCycleInterval\_low\_ 和 setCycleInterval\_high\_, 节点的唤醒周期间隔必须在这两个设置的值之间,以保证延迟或能量消耗单方面过大而影响另外一个方面的结果。表 1 是实验中采取的仿真环境参数。

表 1 仿真参数

| 参数                  | 值       | 意义                    |
|---------------------|---------|-----------------------|
| Bandwidth           | 20 kb/s | 网络带宽                  |
| SYNC_CW             | 31      | SMAC、TMAC 中 SYNC 竞争窗口 |
| DATA_CW             | 63      | SMAC、TMAC 中数据竞争窗口     |
| Slot Time           | 1 ms    | 时槽长度                  |
| Initialenergy       | 100 J   | 初始能量                  |
| Listening power     | 0.4 W   | 侦听能耗                  |
| Receiving power     | 0.4 W   | 接收能耗                  |
| Transimission power | 0.5 W   | 发送能耗                  |
| Sleep power         | 0.005 W | 睡眠能耗                  |

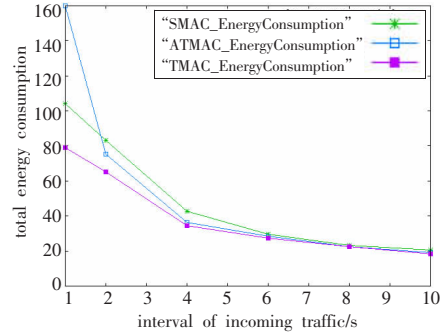
### 3.1 仿真结果

如图 4(a)所示,在线形网络拓扑结构中,所有节点参与通讯,在数据流量比较大的情况下,ATMAC 所消耗的能量要略高于 SMAC 和 TMAC 协议,这是因为,为了降低数据的延迟,而

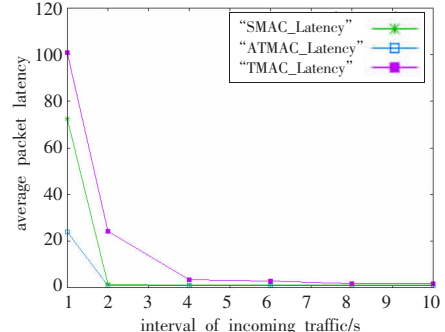
自动将节点的占空比增大,从而增加了能量的消耗。但从数据延迟结果来看,如图 4(b)所示,ATMAC 协议以牺牲一定的能量为前提,而大幅降低了数据的延迟。

线形网络拓扑结构中,所有节点全部参与通信,这在现实的无线网络中是不太实际的,所以,再对一个网状网络(图 2)进行仿真。比较三种协议的能耗和延迟。

在网状网络结构下(图 4)各协议表现出的能耗性能如图 5

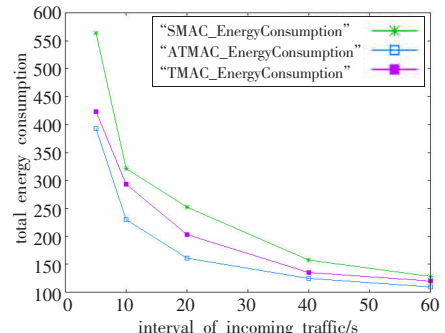


(a)能量消耗图

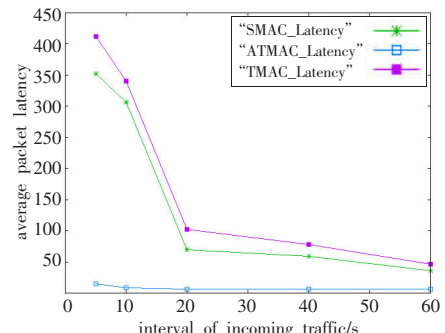


(b)数据延迟仿真图

图 4 线形网络拓扑结构能量消耗和数据延迟仿真结果图



(a)能量消耗图



(b)数据延迟仿真图

图 5 网状网络能量消耗和数据延迟仿真结果图