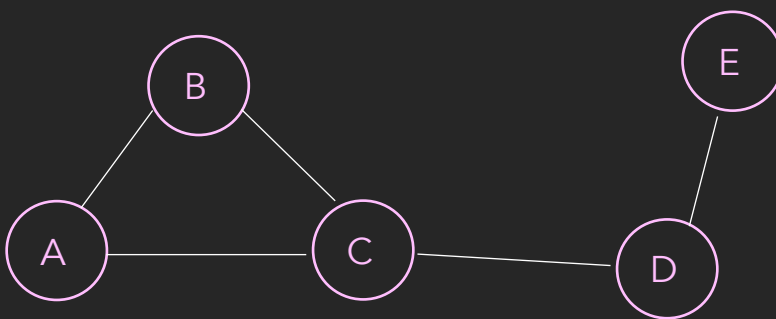# Power Management in Wireless Mac

1) Describe the PAMA problem. How was it solved by signaling?

Solution

☐ What PAMA Power-Aware Multi-Access Protocol) does is Power off for the duration of transmission (known from RTS/CTS) while any neighbor is transmitting to someone else

☐ This arises in a problem as if a node wakes up and hears transmission from its neighbors, it won't know for how much more to sleep as it will miss their RTS.

☐ For example:

    o Suppose A sends RTS to B and C where B is the intended receiver
       $\Rightarrow C$ decides to go to sleep for the duration in the header

    o Now suppose while A is transmitting to B; later, D starts transmitting to E
       $\Rightarrow C$ is off and won't see RTS from D
         ▪ Once it wakes up, A and B finished but D may be still transmitting to E so it can't send anything and should sleep
         » Problem it won't know for how much more to sleep

☐ There are two solutions to this problem and both require a separate control channel for signaling; hence, protocol becomes $PAMAS$ ($PAMA$ with Signaling)
  → Probing (Let C ask nodes around it for how much to sleep $L, L/2, L/4, ...$)
     ▪ They can use their control channel to respond during data transmission
  → Keep control channel on while sleeping
     ▪ Thus, C never misses RTS/CTS in the first place

**2)** How to avoid probing in PAMAS? Mention pros and cons.

Solution

The other option besides probing is to keep the control channel on while sleeping (i.e., only turn of the data channel).

Pros	Cons
Each node knows for how much more to sleep if any new transmission occurs without needing to probe	Not possible if the same hardware is used for both the control and the data channels. Besides, the control channel may be subject to interference.

**3)** How does ATIM window achieves power saving in IEEE 802.11 Ad Hoc Mode?

Solution

- Time is divided into beacon intervals
- Each beacon interval begins with an ATIM window (announcement traffic indication message)
- No RTS/CTS and one channel

ATIM window

Beacon interval
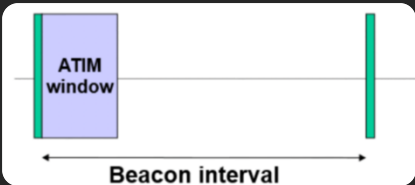
- If the node has no packets to transmit and it didn't get an $ATIM$ request during the $ATIM$ window then it sleeps and thus, power saving is achieved
  - Else, if it got an $ATIM\ Req$ then it should respond with $ATIM\ Ack$ during the $ATIM$ window
  - Likewise, if it has something to send then it should establish an $ATIM\ Req$ and get back an $ATIM\ Ack$ during the $ATIM$ window
    - In both these cases, the node stays up to engage in communication after the $ATIM$ window

- The following graphic shows an example for communication between B and A. Note how C was able to save power as it didn't need to communicate.

Node A

ATIM Req   ATIM Ack   Data   Ack

Node B

Node C   Sleep
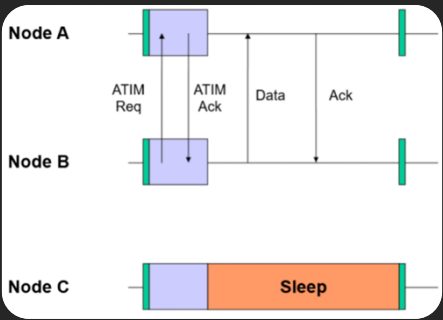
4) Having a small ATIM window is the perfect choice. Agree or disagree.

> Solution

☐ Disagree

→ If the *ATIM* window is too small then a node must be too early to catch it from the beginning so most transmissions will be delayed to the next beacon interval (as won't receive acks) which will cause more collisions causing more delays so overall delays increase hurting efficiency.

This was interpreted form the slides "Delay increases, specially in high load conditions since ATIM requests will collide so nodes won't receive ACKs and will wait for the next beacon interval."