POWERNAP for Longevity in Wireless Sensor Networks

Onur Soysal, Sami Ayyorgun, Murat Demirbas

Abstract—**Idle-listening is the biggest challenge for the energy-efficiency and longevity of multihop wireless sensor network (WSN) deployments. Existing coordinated sleep/wakeup scheduling protocols eliminate idle-listening, but become inapplicable quickly for complex packet arrival patterns. We present a novel coordinated sleep/wakeup protocol POWERNAP that addresses this problem without the overhead of employing control messages for sleep/wakeup scheduling. Our insight is to piggyback the seed of the pseudo-random generator that encodes the sleep/wakeup scheduling information onto the data packets to enable any recipient/snooper to compute its sleep/wakeup schedule from this seed. In essence, POWERNAP trades off extra computation for avoiding expensive control message transmissions. We show through simulations and real implementation on TelosB motes that POWERNAP eliminates the idle-listening problem without any control message overheads and achieves low-latency and self-stabilizing fault-tolerant relaying of data packets.**

Index Terms—**Multihop relaying, Load-balancing, Coordinated sleep/wakeup**

I. INTRODUCTION

ENERGY efficiency is a major challenge in wireless sensor network (WSN) deployments. In multihop WSN deployments, the radio is often the critical component that drains most of the energy. Even while "idly listening" to the channel in order to detect any potential transmissions addressed to itself, the radio wastes the same amount of energy as in packet transmissions. In order to eliminate idle listening, it is crucial to shut down the radios and power them back only when they are needed. This simple requirement, however, leads to several challenges for multihop data relaying.

1) *How do we coordinate the sleep/wake-up times of nodes?* To achieve communication, both the transmitter and the receiver should be awake at the same time. Thus, the sleep/wakeup protocol should be aware of the communication patterns and eliminate idle listening by waking up pairwise nodes only when a packet transmission is occurring between the two.
2) *How do we load-balance the relaying duties among nodes?* Due to redundant deployment and fault-tolerance reasons, there are multiple nodes within the transmission range of any node. If the node always chooses the same neighbor for relaying, the energy of that neighbor depletes quickly, while the other neighbors are left underutilized. The depletion of energy in an unbalanced manner often leads to partitioning and ends the lifetime of the multihop relaying deployment prematurely. Thus, the sleep/wakeup protocol should load-balance the relaying duties among multiple relay paths in the network.
3) *How do we ensure that the death of one node does not disable other nodes in the relay path?* Even when there are multiple relay paths, if the relay paths are determined statically beforehand in a node-disjoint manner, the death of one node in a path will thrash the entire path and put unproportionally more strain on the network. Thus, the sleep/wakeup protocol should employ as many permutations of relay paths as possible to achieve graceful fault-tolerance, and should be able to handle the resultant complex packet arrival patterns for the wakeup scheduling of the relay nodes.
4) *How do we achieve low-latency in multihop relaying?* Since many applications require real-time guarantees, we also care about reducing the latency in multihop relaying. Thus, the ideal sleep/wake-up protocol should introduce almost no latency in relaying by aligning the wakeup times of the relay nodes in the path for seamless forwarding of the packet.

Your name is with xyz Department…

A. Related work

There have been several attempts to mitigate the energy-efficiency problem. However, none of these solutions is adequate in answering all of these questions.

BMAC [6] provided low-power-listening (LPL) to the receiver nodes and shifted the coordination burden to the sender node, which needs to send a long preamble to ensure that receiver nodes detect the presence of the transmission. BMAC, however, still suffers from the idle listening problem because all receivers switch to the receive state after a detection of a preamble to be able to receive the transmission even though the transmission is not addressed to them. Low-power-probing (LPP) [**?**] alleviates this problem by a role-reversal: in LPP the sender needa to keep awake until the receiver it is interested in transmitting to wakes up and transmits a probe. Both LPL and LPP fail to achieve challenge 4, the low-latency requirement in multihop relaying.

SMAC [8] uses rendezvous to enable the motes in a singlehop to synchronize their sleep/wakeup times, and leads to synchronized singlehop clusters in the network. Of course, this also leads to wasteful listening as in reality there may not be a packet transmitted at every rendezvous time. SMAC also fails to satisfy challenge 4, the low-latency requirement due to the misaligned sleep-wakeup schedules of nodes in different singlehop neighborhoods.

A TDMA approach can reduce the idle listening problem by allowing the nodes to sleep most of the time, except in their timeslots. However, since nodes do not know when they may receive messages they need to wake up and check the channel