Delay-Energy Aware Routing Protocol for Sensor and Actor Networks

Arjan Durresi, Vamsi Paruchuri
Department of Computer Science
Louisiana State University
Baton Rouge, Louisiana, USA
durresi@csc.lsu.edu

Leonard Barolli
Fukuoka Institute of Technology,
Fukuoka, Japan
barolli@fit.ac.jp

Abstract

We present a novel Delay-Energy Aware Routing Protocol (DEAP) for heterogeneous sensor and actor networks. DEAP enable a wide range of tradeoffs between delay and energy consumption. The two major components of DEAP are: (a) an adaptive energy management scheme that controls the wake-up cycle of sensors based on the experienced packet delay; and (b) a loose geographic routing protocol that in each hop distributes the load among a group of neighboring nodes. The primary result of DEAP is that it enables a flexible range of tradeoffs between the packet delay and the energy use. Therefore, DEAP supports delay sensitive applications of heterogeneous sensor and actor networks. DEAP is scalable to the change in network size, node type, node density and topology. DEAP accommodates seamlessly such network changes, including the presence of actors in heterogeneous sensor networks. Indeed DEAP takes advantage of actor nodes, and uses their resources when possible, thus reducing the energy consumption of sensor nodes. The performance of DEAP remains very good even in large networks, and it scales with density. Through analysis and simulation evaluations, we show that DEAP improves the packet delay and system lifetime compared to other protocols.

1. Introduction

Heterogenous Wireless sensor and actor networks (WSAN), supported by recent technological advances in low power wireless communications along with silicon integration of various functionalities such as sensing, communications, intelligence and actuations are emerging as a critically important disruptive computer class based on a new platform, networking structure and interface that enable novel, low cost, high volume applications [6, 5, 14, 17] such as nuclear, biological and chemical attack detection and protection, home automation, battlefield surveillance and environmental monitoring [6, 12, 27].

Sensor nodes in general are extremely small, low-cost, low energy that possess sensing, signal processing and wireless communication capabilities. Sensors usually gather information about the physical world. Actor nodes are nodes capable taking decisions and then perform appropriate actions. An example of actor nodes are robots able of sensing, communicating and performing actions. Actor nodes in general are equipped with larger energy sources than sensors. Heterogeneous ad-hoc wireless networks of large numbers of such inexpensive but less reliable and accurate sensors combined with few actors can be used in a wide variety of commercial and military applications such as target tracking, security, environmental monitoring and system control.

In wireless sensor networks, it is critically important to save energy. Current research on routing in wireless sensor networks mostly focused on protocols that are energy aware to maximize the lifetime of the network, are scalable to accommodate a large number of sensor nodes, and are tolerant to sensor damage and battery exhaustion [7, 8, 19, 30, 31, 33]. Since such energy consideration has dominated most of the research in sensor networks, the concepts of delay was not primary concern in most of the published work on sensor networks. However in WSANs, depending on the application, there may be a need to rapidly respond to sensor input. Moreover, to provide right actions, sensor data must still be valid at the time of acting. Therefore, the issue of real-time communication is very important in WSANs since actions are performed on the environment after the sensing occurs. Since packet queuing delay dominates its propagation delay, the goal of delay sensitive solutions would be to control and optimize queuing time per hop as well as number of hops. It is expected that reducing delay would require sacrifices on energy efficiency [34].

The design of a good delay sensitive power management protocol for heterogeneous WSANs should allow a flexible tradeoff between packet delay and the corresponding energy
consumption. Power management schemes should take advantage of actor nodes, and use their resources when possible. Another important attribute is the scalability to change in network size, node type, node density and topology.

Recent papers propose MAC, routing, and topology maintenance schemes that try to save energy based on aggressive power-off strategies. In fact, it has been recognized that the only way a node can save substantial energy is to power off the radio, since transmitting, receiving and listening to an idle channel are functions that require roughly the same amount of power. In [23] we have introduced Random Asynchronous Wakeup (RAW), an energy management scheme explicitly designed for wireless sensor networks. While reducing energy consumption was the primary goal in our design, our protocol achieves good scalability.

In this paper we present DEAP that enables lower packet delays and flexible tradeoffs between delay and energy consumption. DEAP is an extension of our previously presented Random Asynchronous Wakeup (RAW). DEAP distributes the load among the nodes in the forwarding set proportionally to their awake period, that is adapted to the queue produced by packets and the remaining energy. DEAP handles seamlessly the presence of actors, by using their resources at the advantage of other nodes with less energy. The final result of DEAP is lower packet delay and better, more flexible tradeoffs between delay and energy consumption. We present simulation of delay and delivery rate versus consumed energy by sensors.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 presents DEAP for sensor and actor networks. Section 4 describes our simulation model and discusses the simulation results. Section 5 concludes the paper.

2. Related work

Current research on sensor networking mostly is focused on solutions that try to maximize the lifetime of the network and are scalable to large networks. Therefore, other metrics such as delay, throughput, jitter were not primary concern of most of research work. However, with the introduction of actors [5] and the consequent increase interest in real-time sensitive communications, new challenges have been posed. While being able to control the use of energy remains a paramount design goal, it is becoming also important how to better trade it for lower delays.

The main sources of energy wastage are collisions, idle listening, overhearing and control packet overhead. All MAC [3],[33],[13],[29],[20] protocols contention based (like CSMA) or scheduled protocols (like TDMA) try to avoid collisions. The next major energy wastage source is idle listening, which occurs when the receiver is listening to the channel to receive possible data. Also sending, receiving and listening for control packets consume energy, which reduces the effective throughput.

One approach to prevent energy wastage due to above sources is to control the node receiver by setting it to sleep mode when no data is expected and to wake up mode when communication is expected (wakeup schemes) [32]. Wakeup schemes can be classified as synchronous and asynchronous. Synchronous wakeup approach is used by the IEEE 802.11 [3] ad hoc power save (PS) mode. An asynchronous wakeup scheme for mobile ad hoc networks by Zheng et al [36], builds on the block design problem in combinatorics.

The energy savings and wakeup delay can be improved by an additional wakeup or signaling radio. The PAMAS (Power Aware Multi-Access) protocol [24] is an adaptation of the basic mechanisms of IEEE 802.11 to a two-radio architecture. Since the power consumption of the wakeup radio is significantly lower than that of the radio transmission, it can be awake for the entire period, consuming little energy. STEM (Sparse Topology and Energy Management) [25] also uses two radios, one is used as a wakeup radio and other is used for data transmission.

S-MAC [33] is a protocol developed to address the energy issue in the sensor networks, building on contention-based protocols like IEEE 802.11. S-MAC follows a simple scheduling scheme that allows neighbors to sleep for long periods and to synchronize wakeups. S-MAC uses a fixed sleep interval regardless of traffic load. T-MAC [29] extends S-MAC by adjusting the length of time sensors are awake between sleep intervals based on communication of neighbors. Thus, less energy is wasted due to idle listening when traffic is light.

While many energy-aware MAC and routing protocols have been proposed for sensor networks, such the ones above mentioned, very little research has been done to combine real-time requirements and energy-awareness. Recently in [11] it was proposed an implicit prioritized access protocol for sensor networks that utilizes Earliest Deadline First scheduling algorithm to guarantee the delay for real-time traffic. RAP [21], another protocol uses a real-time scheduling policy for sensor networks. The scheduling is based on the prioritization on the traffic requirements and closeness to the gateway.

Sequential Assignment Routing (SAR) protocol includes QoS requirements in routing decisions [6, 26]. SAR creates multiple paths between sensor and sink, by considering energy resources and QoS metrics. Among these paths only one will be used and the rest as backup. While SAR enables better energy use than the minimum-energy metric algorithm, it suffers from overhead of maintaining information at each sensor. Akkaya and Younis [4] have proposed
a QoS aware routing protocol for sensor networks as an extension of [35]. SPEED, a QoS routing protocol for sensor networks that provides soft real-time end-to-end guarantees is proposed in [16]. The protocol uses geographic forwarding and allow applications to estimate the end-to-end delay, moreover SPEED can provide congestion avoidance when the network is overloaded.

Most of the routing protocols for sensor networks are designed to support query processing [18, 28, 10]. In WSAN, besides the query communication, other type of communication such as sensor-actor and actor-actor become very important. That is why we propose a new routing protocol that supports also these types of communications.

We have proposed a routing protocol based on Random Asynchronous Wakeup (RAW), a power management scheme [23]. While reducing energy consumption was the primary goal in our design of RAW, our protocol can also achieve good scalability and low latency.

3. Delay Energy Aware Routing Protocol

DEAP mainly consists of two components - Routing based on Forwarding Sets and the Random Wakeup Scheme. The routing methodology in DEAP is designed to take advantage of the fact that sensor networks are densely deployed. In conventional routing protocols, the shortest path between two nodes is computed proactively or reactively and a node forwards a packet only to the next node in the shortest path computed. A high node density results in the existence of several paths between two given nodes, whose path lengths are very close to the length of the shortest path. Thus, a packet can be forwarded to any such several paths in order to be delivered to the destination without affecting the path length and delay when compared to the shortest path. Our random wakeup scheme allows for a node to be active during a randomly chosen fixed interval in each time frame. This removes the necessity of time synchronization and makes the protocol implementation very simple. The wakeup time depends on the traffic load, which enable low delays. In this section, we first elaborate our routing methodology based on forwarding sets and then study the random wake up scheme. We then present the complete design of DEAP.

3.1. Routing based on Forwarding Sets

In the geographic routing protocol, a packet is forwarded to a neighboring node that is closest to the destination. However, in a sensor network, in which not all nodes might be active at a given point of time, a packet can be forwarded to the active neighbor that is closest to the destination, or the packet can be queued until the closest neighbor among the rest becomes active, and the packet can then be forwarded to this neighbor.

In this paper, we assume a modification of the geographic routing protocol such that a packet is sent to any of the active neighbors that meet a forwarding criterion (which is discussed later in the section). We define Neighboring Set and Forwarding Candidate Set as follows:

- The Neighbor Set of node $i$: This is the set of nodes that are inside the radio range $R$ of node $i$.

$$NS_i = \{ \text{node} \mid \text{distance(node, node } i) \leq R \}$$

- The Forwarding Candidate Set of node $i$: For a given destination, this is the set of potential neighboring nodes to which node $i$ can forward a packet.

We define Forwarding Candidate Set based on geographic distance to the destination.

**Forwarding criterion:** For a given source $s$ and destination $d$, a neighbor $k$ of $s$ is a node in FCS if:

$$D(k, d) < D(s, d) - Th$$

where, $D(i, j)$ is the geographic distance between nodes $i$ and $j$.

Thus, if a neighbor $k$ is closer to the destination by at least $Th$ than the node $s$ itself, then $k$ belongs to the Forwarding Candidate Set, as shown in Fig. 1. The d-FCS selection criterion guarantees that there would be no loops in the path. This is because a node always forwards a packet to a node that is closer to the destination than itself. At the same time, this simple criterion cannot guarantee the delivery of a packet to the destination in presence of holes. At high network densities, it can be safely assumed that holes would not exist. In case holes are present, the criteria for selection has to be extended based on the ideas presented in [15]. In this paper, we assume that no holes are present in the network.

Routing based on forwarding sets increases the path length. The $Th$ value limits the maximum path length, as with each transmission a packet traverses at least a distance of $Th$ towards the destination. Intuitively, because of increased path lengths, it might seem that Forwarding Set based routing adds additional overhead in terms of energy consumption. However, when combined with the random wake up scheme the total energy consumed by a sensor in DEAP is lower.

3.2. Random Wakeup Scheme

The idea is to have each node wake up once in every slot, be awake for a predetermined time, and then sleep again. To elaborate, consider time slots of fixed interval $T$ and the active time of $T_a$ for each sensor node in each time slot.
Thus, if there are \( m \) neighbors in the forwarding set of node \( S \) to which a packet destined to \( D \) can be transmitted to, then the probability that at least one of those nodes is awake, when \( S \) is awake is given by:

\[
P = 1 - \left(1 - \frac{2T_a}{T}\right)^m
\]

The scheme is self-adapted to traffic load. The duration of active period depends on the load level of the node. The queue size of a sensor is used as an indicator of the load of the node. Each sensor computes its active duration based on its queue size, thus being active for longer durations if it has large queue and being active for lesser duration if there is no queue.

Each sensor computes its active duration in the following manner. The wake up duration \( w_a \) of sensor \( x \) is calculated as:

\[
T_{a,n} = \begin{cases} 
T_{a,n-1} \cdot \frac{\alpha}{\beta} & \text{if } S_Q > 0 \\
\frac{T_{a,n-1}}{\beta} & \text{if } S_Q = 0 
\end{cases}
\]

where, \( S_Q \) is the size of the queue at sensor \( x \) and \( T_{a,n} \) is the current wake up interval and \( T_{a,n-1} \) is the previous one, \( \alpha \) and \( \beta \) are protocol parameters that determine the wake up duration.

Thus, whenever a sensor has some packets buffered, it stays awake for longer duration by a factor of \( \alpha \). But, when there are no packets buffered which indicates that the load on the sensor is low, the sensor is active for less time by a factor of \( \beta \). The values of \( \alpha \) and \( \beta \) determine the delay experienced by the packets and energy consumed by the sensors. The higher the value of \( \alpha \) is, the longer a sensor is active anticipating high loads, thus leading to lower delays and high energy consumption. The higher the value of \( \beta \) is, the longer will a sensor sleep assuming low traffic, thus resulting in lower energy consumption. We present the performance of our scheme for different values of \( \alpha \) and \( \beta \).

Also, it should be observed that though the sensors independently decide their active intervals and durations, the durations of neighboring sensors are highly correlated. This can be explained in the following manner: Assume that a sensor \( A \) has a very big queue which leads in the sensor being active for longer duration than its neighbors. But, once one of its neighbors in the forwarding region gets active, many packets are forwarded to that node, which results in that neighbor also being active for longer duration. Thus, until queue size of \( A \) comes down, it leads in longer duration of its neighbors, thus resulting in high correlation in active duration of neighboring nodes.

We consider the performance in presence of actor node which are considered to be powerful and having considerably higher energy levels than sensor nodes. Because of high energy levels, an actor is assumed to be active all the time. In such scenarios, a sensor transfers of its packets to an actor, if it is present in its neighborhood, thus maintaining low queue size load. Thus, presence of an actor node results in the neighboring sensors to sleep for longer durations. Thus, this protocol is self-adaptive to the heterogeneous conditions prevailing in the network.

### 3.3. Neighbor Discovery

The neighbor discovery procedure operates as follows. Whenever a node \( i \) wakes up, it broadcasts a beacon message piggybacking its own \( id \), the \( starttime \) of its wakeup period and other information subject to channel contention/resolution rule. The energy level of the sensor is also included in the beacon message. To implement the protocol, each node keeps a neighbor list in which each entry has fields as shown in Table 1.

<table>
<thead>
<tr>
<th>Node id</th>
<th>clock</th>
<th>Schedule</th>
<th>Lifespan</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Entry in the neighbor list maintained by each node

Whenever a new neighbor is discovered, a new entry is added. Also, among the neighbors of \( i \), the node \( j \) that has been awake for the longest period sends a beacon message to \( i \) as an acknowledgement and also piggybacks its neighbor list. All nodes that receive the acknowledgement update their neighbor lists according to neighbor list of \( j \). This ensures consistency in neighbor lists of all nodes.

### 3.4. Packet Forwarding

We use a greedy geographical routing protocol that forwards packet to an active neighbor that is closest to the destination at each hop. Whenever a node \( i \) (or) has a packet destined to node \( d \), it selects a node \( k \) from its 1-hop neighbor...
list, such that $k$ is closer to $d$ than any other active neighbor of $i$ and $k$ is closer to the destination by at least $Th$. The threshold $Th$ limits the length of a path to a maximum of $D(s, d) \times \frac{R}{Th}$. $D(s, d)$ is the distance between the source and the destination, while $R$ is the transmission range of the sensors.

4. Performance Evaluation

We have developed a simulator using OMNET++, a discrete-event simulation framework [2, 9, 22], to evaluate the performance of our protocol. All simulations were based on a network of dimension $5R \times 5R$, where $R$ stands for the transmission range of sensor node. Various node densities were considered. The model parameters and limits on transmission bit rates and energy ratings are set according to Crossbow MICA2 sensor nodes [1]. Power consumption in the model is based on the amount of the current draw that Crossbow MICA2 sensor node’s radio transreceiver uses, as shown in Table 2 [1].

In our setup, 40 nodes were made to generate traffic to random destinations at different rates varying from 1 packet/sec up to 16 packets/sec. Each data packet had a size of 64 bytes including a header of 12 bytes of header information and hence length beacon and other control packets are assumed to be 12 bytes. Nodes were randomly deployed with uniform distribution with various densities. The energy consumption for switching the radio from idle to sleep modes and vice versa is assumed to be negligible and hence not considered. Also, the location is assumed to be available via GPS or other localization means and thus is not simulated. We primarily focus on delay experienced by the packets, delivery percentage and total energy consumed by the sensors. In presence of actor nodes, the energy expended by them is not taken into consideration as we assume that energy is not an issue for actor nodes.

We study the performance of DEAP for two different value sets of $\alpha$ and $\beta$ as follows:

**DEAP-1:** $\alpha = 2$ and $\beta = 1.5$

In this scenario when there are packets buffered, a sensor doubles its active duration and when there is no load, the sensor would reduce its active duration by 1.5. Because of higher value $\alpha$ and lower $\beta$, this scheme is expected to result in lower delays and higher energy consumption as confirmed by simulation results.

**DEAP-2:** $\alpha = 1.5$ and $\beta = 2$

In this scenario when there are packets buffered, a sensor increases its active duration by 50% and when there is no load, the sensor would halve its active duration. Because of lower value $\alpha$ and higher $\beta$, this scheme is expected to consume less energy but results in higher delays.

In [23] we have presented the results of analysis and simulations on the choice of our protocol parameters: threshold, time frame length and active period, average latency per hop, and average delivery ratio at the offered load.

We consider two different network scenarios:

1. Wireless Sensor Networks (WSNs) consisting of different kinds of sensors. Sensor nodes of various (but similar) capabilities are assumed to be present and the each sensor can be sensing different phenomenon. In such a network, the initial energy of each sensor is randomly set to a value between 5 and 20 power units.

2. Wireless Sensor and Actor Networks (WSANs) consisting of sensor nodes of same capabilities along with some actor nodes. The initial energy level of a sensor node is set to 20 power units and that of actor node is set to 100 power units.

Figure 2 and 3 present the performance of DEAP in a sensor network for different loads in terms of delay experienced per hop, delivery ratio and energy consumed. As expected, DEAP-1 consumes more energy than DEAP-2, but results in lower delays. Also, it should be noted that both cases of DEAP result in lower energy consumption when compared to RAW [23].

Depending on the network application and data being reported, whether its delay sensitive traffic, different values can be set for $\alpha$ and $\beta$.

<table>
<thead>
<tr>
<th>Transmit</th>
<th>Receive</th>
<th>Idle</th>
<th>Sleep</th>
</tr>
</thead>
<tbody>
<tr>
<td>15mA</td>
<td>8mA</td>
<td>7mA</td>
<td>2µA</td>
</tr>
</tbody>
</table>

**Table 2. Typical current draw values of sensor nodes for simulation purpose**

Figure 2. Delay and Energy expended in a Sensor network with 150 sensor nodes, 5*5 network

Proceedings of the 2005 11th International Conference on Parallel and Distributed Systems (ICPADS’05)
0-7695-2281-5/05 © 2005 IEEE
DEAP is a distributed, randomized algorithm where nodes make local decisions on whether to sleep, or to be active based on the delay level experienced by packets. DEAP is scalable to the change in network size, node type, node density and topology. DEAP accommodates seamlessly such network changes, including the presence of actors in heterogeneous sensor networks. DEAP takes advantage of actor nodes, and uses their resources when possible, thus reducing the energy consumption of sensor nodes.

5. Conclusion

We presented Delay-Energy Aware Routing Protocol (DEAP) for heterogeneous sensor and actor networks. DEAP has two main components, a novel energy management scheme and a loose geographical routing protocol for heterogeneous sensor networks.

DEAP supports delay sensitive applications on heterogeneous sensor and actor networks by enabling flexible trade-offs between the packet delay and the energy consumption. DEAP is a distributed, randomized algorithm where nodes make local decisions on whether to sleep, or to be active based on the delay level experienced by packets. DEAP is scalable to the change in network size, node type, node density and topology. DEAP accommodates seamlessly such network changes, including the presence of actors in heterogeneous sensor networks. DEAP takes advantage of actor nodes, and uses their resources when possible, thus reducing the energy consumption of sensor nodes.

References


