Poster Abstract: Introducing an adaptive MAC layer to support Quality of Service in WSN

Matteo Ceriotti
FBK-IRST & University of Trento, Trento, Italy
Email: ceriotti@fbk.eu

Amy L. Murphy
FBK-IRST, Trento, Italy
Email: murphy@fbk.eu

I. Motivations

Typical wireless sensor networks (WSNs) are characterized by large geographic areas with many sensors that collect and deliver data to both mobile and stationary users. The convenience of battery power and wireless communication make it easy to add, substitute and remove nodes at any time in active deployments. Current solutions to support application development, however, provide only best-effort service, an unacceptable limitation for a wide range of applications with clear quality constraints, such as home and building security, medical monitoring, or disaster recovery. In these cases, the available resources must be carefully tuned to provide the maximum benefit to the system as a whole. This resource management can be achieved in various ways, e.g., the application programmer can develop custom solutions, or general solutions can be offered by the system as a service to the application. Clearly, the latter option provides a more generally applicable solution, hiding the complexity of resource control from the application developer.

To focus our work, we outline an earthquake recovery scenario that can greatly benefit from Quality of Service (QoS) support in a WSN. Consider that after the disaster, sensors are distributed throughout a city and the surrounding countryside, possibly connecting with sensors deployed prior to the disaster. The goal of the sensor network is to support coordination of the short term recovery efforts, with typical duration on the order of months. These efforts include building stability monitoring, transportation of supplies, and general coordination among aide workers.

In this scenario, the sensors provide data to aide workers to augment their decision making. The needs of these users is likely to vary greatly. For example, a worker assigned to a specific area may require data with high quality from a zone localized around him, to guide precisely his actions. Instead, a coordinator may require a lower quality, but over a larger zone, to coordinate the recovery effort as a whole. Furthermore, not all data consumers will be human, allowing for actuators that consume data and produce immediate effects on the environment. Finally, the guarantees provided by the service, e.g. the maximum delay experienced or the bandwidth available to communicate with other components in the system, can be exploited by the system users to better anticipate the effects of their own actions, e.g. the time required to notify a worker of a particularly dangerous circumstance.

Recent efforts have begun to explore QoS in WSNs [1], however they primarily support narrow metrics, e.g., minimal energy usage, which are strongly encapsulated inside the coordination protocol. In contrast, we offer a full solution, providing a flexible API for QoS specification as well as the supporting algorithms and protocols. We also target a very dynamic, heterogeneous scenario, in which devices have different capabilities, and concurrently running applications individually express QoS needs that must be simultaneously supported.

We approach this challenge through REINS, a Resource-aware Infrastructure for Networks of Sensors. The key elements of REINS are its application interface and supporting algorithms and protocols. The interface both accepts application-defined constraints, and provides feedback if they cannot be met. To support the interface, REINS is composed of a suite of algorithm and protocol layers, which, when vertically connected, result in an integrated solution to support application quality needs.

When considering the resources of a typical WSN, the communication channels can be seen as the core resource that enables all application functionality. As data flows along multi-hop paths from producers to consumers, the nodes must coordinate to appropriately share the wireless medium to meet the application requirements for data delivery. Because the MAC layer allocates the resources that govern all information exchange, it is a reasonable starting point for supporting application defined quality constraints on data flows.

II. Supporting QoS, Starting with the MAC

Some communication guarantees are already provided at the MAC layer by TDMA-like protocols, as each node is assigned a fixed length time slot inside a frame. During the allocated slot, the owner can communicate without interference. However, typical allocation mechanisms treat all nodes equally, ignoring their communication needs, and hence the application requirements on the data flows moving through the network. Our intent is to study the dimensions along which the TDMA-like channel allocation can be modified in order to support application needs, defining a decentralized algorithm that considers both application needs and the available wireless communication resources. Additionally, as the domain we target is dynamic, our approach must adjust to the injection of

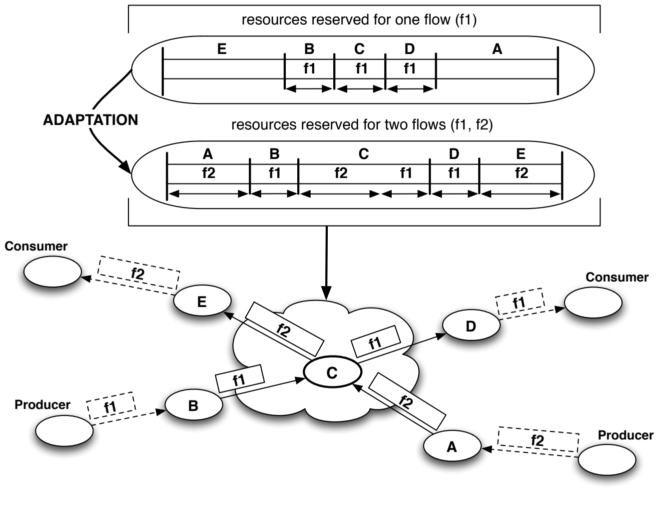

Figure 1: Adaptation example: the frame as perceived by C, is adapted to allocate resources for a new data flow ($f2$), preserving slots already assigned to another data flow ($f1$).

new requests, the completion of old requests, and the insertion and removal of new nodes.

We identify two main dimensions that affect the multi-hop communication properties, namely slot *duration* and the *position* of the slot within the frame. By adapting these two properties, we are able to both provide guarantees to applications and improve the overall system utilization over existing approaches.

The first property, namely slot duration, is typically predefined and common among all nodes. Such a solution provides each node with a constant amount of time to exchange information with its neighbors; however in our environment, nodes are likely have different duties and the various data flows moving through the system could vary greatly in terms of individual data size and production frequency. Indeed, the definition of single unique slot length influences the performance of the entire system. Although some protocols [2] allow nodes to contend for unused slots to increase the bandwidth of a node with data to transmit, the success of this contention is dependent on the current system load and therefore does not allow quality guarantees to be made to the application.

The second key dimension is the position of the assigned time slot. More precisely, this refers to the offset of the slot from the beginning of the frame and affects the latency experienced by data as it travels through the network. Consider that after a message is received, the node must wait until its assigned time slot to forward the data to the next hop. The longer a message is delayed at a node, the greater the latency of the communication from producers to consumers. The latency on the entire route can be decreased by properly aligning the slots of the nodes along the path, ensuring that the receiving nodes have assigned transmission slots immediately after the slots of the nodes sending data to them.

Our approach allows variation and control over both these dimensions, as shown by example in Figure 1. Specifically we remove the assumption of fixed slot duration, adjusting slot length according to application requirements at each node. For the second dimension, we both remove the random offset selection and allow the slot to start at any arbitrary offset in the frame. This requires knowledge of the data path and the data size, available from the routing and application layers. Our initial protocol uses key ideas from wake-up scattering [3] to exchange information (e.g., slot duration and location) among neighboring nodes and to spread the slots according to flow requirements.

To address the dynamic nature of the environment, we recognize that updates in the slot schedule can have significant overhead. For example, the adaptation of both duration and position at a single node can recursively require similar changes at its neighbors. This chain of updates stops when the local adaptation no longer influences the neighborhood, however, in the worse case, the update will propagate to the borders of the network to determine the feasibility of the proposed modification. To limit this overhead, we introduce the notion of *barriers*, across which adjustments cannot propagate. These barriers can be established on the basis of physical network properties (e.g., the confines of a building) or according to logical properties (e.g., a maximum number of hops). In any case, the goal is to limit the influence of an operation.

III. Future Work

We are currently working on the initial implementation and evaluation of a MAC protocol that adapts the position and dimension of the allocated time slots to meet required workload. The behavior of the protocol will be influenced by several parameters, including the boundaries of the allowed adaptation and the interval of variation of the time slot length. We must further evaluate how these parameters affect the ability of the distributed protocol to achieve the same system utilization and application guarantees of an offline optimization.

Finally, the MAC protocol outlined here is only one component of the REINS infrastructure where we will address QoS in WSNs at multiple levels. For example, the selection of the multi-hop route greatly influences the capacity and ability of the system to meet multiple simultaneous application requests. While we will consider existing routing proposals [4], we anticipate developing a new approach to consider both flow requirements and inter-operate with the MAC protocol. Such cross layer interaction is key to realize the full potential of the REINS framework to support QoS in WSNs.

References

[1] D. Chen and P. K. Varshney, “Qos support in wireless sensor networks: A survey,” in *ICWN ’04: Proceedings of the 2004 International Conference on Wireless Networks*, 2004, pp. 227–233.

[2] I. Rhee, A. Warrier, M. Aia, and J. Min, “Z-mac: a hybrid mac for wireless sensor networks,” in *SenSys ’05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, 2005, pp. 90–101.

[3] A. Giusti, A. L. Murphy, and G. P. Picco, “Decentralized scattering of wake-up times in wireless sensor networks,” in *EWSN ’07: Proceedings of 4th European conference on Wireless Sensor Networks*, 2007, pp. 245–260.

[4] K. Akkaya and M. F. Younis, “A survey on routing protocols for wireless sensor networks,” *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.