Improving Utilization and Sustainability of Low-power Wireless Sensors through Decentralized Role Allocation in a Multi-agent System

Anonymous Author(s)
Submission Id: «EasyChair submission id»

Abstract

Wireless sensors are low-power devices that are increasingly relying on sustainable energy harvesting over expensive batteries for their operation. This enables lower cost of installation and higher deployment flexibility, which benefits applications in many domains, such as building automation and environmental sensing. However, the constrained and non-deterministic availability of ambient energy in such devices means that the system designer has to estimate their energy availability when assigning them measurement roles in a system at design time. Such estimates are often overly conservative, which leads to tighter than necessary coupling and hence reduces functional flexibility at run time. Energy-harvesting sensors are therefore often under-utilized while systems rely to a greater extent on less sustainable battery-powered sensors due to higher predictability of their energy availability. We propose an approach where low-power wireless sensors, functioning as agents in a multi-agent system, use their local knowledge to autonomously evaluate their functional capabilities to serve the needs of the system in the best possible manner at run time. The knowledge about the system organization that is required for this purpose is disseminated in the network using an energy-efficient gossiping protocol.

Our evaluation shows that our approach not only simplifies the system design, but also leads to higher efficacy in fulfilling measurement roles, and improves the utilization of the energy-harvesting sensors while relying on battery-powered sensors as the last recourse, thereby leading to more sustainable measurement while prolonging battery life.

Keywords

Wireless Sensor and Actor Networks, Multi-Agent Systems

ACM Reference Format:
Anonymous Author(s). 2024. Improving Utilization and Sustainability of Low-power Wireless Sensors through Decentralized Role Allocation in a Multi-agent System. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 9 pages.

1 Introduction

Granular measurement of environmental conditions like air temperature, humidity, and pollutants in living and working spaces can help maintain comfort, save energy, and address safety and health concerns. In the face of an ever-increasing need for more widespread monitoring and control of environments like buildings, the industry is looking towards wireless sensor and actor networks (WSANs) since the traditionally used wired sensors are not only expensive to install, but are inflexible to modification when the deployment context changes (e.g., through changes in the usage pattern or the physical layout of a building). To achieve the desired flexibility (or even feasibility) of physical deployment without being tethered by power supply wiring, such wireless devices have to be powered either by a battery or employ some form of energy-harvesting (e.g., by utilizing energy from ambient light).

In most cases, the wireless sensors are both energy-constrained and also have little CPU and memory in comparison to fully-powered automation devices. A system designer who wishes to incorporate energy-harvesting sensors needs to carefully consider whether the expected energy available to the sensor can meet the measurement requirements (e.g., in terms of precision or measurement frequency) of the control and monitoring programs hosted by the actors.

To deal with uncertainties about the energy availability in energy-harvesting sensors, system designers usually incorporate battery-powered sensors as a backup [7, 17]. But, here the designers have to consider the environmental impact of single-use batteries and the costs of regularly replacing them and therefore their utilization has to be planned in such a way that the battery life can be prolonged. Such planning is specifically complicated because it requires careful consideration of the hardware and network characteristics that are expected at run time [19].

Consequently, sensors are assigned specific measurement roles at design time. This forgoes the flexibility that would be relevant to adapt the system at run in the face of changing energy availability or evolving application requirements.

In summary, although WSANs offer higher flexibility of deployment, the difficulty of predicting the ability of sensor nodes to fulfill measurement requirements at run time is a hindrance to their more widespread adoption (see also [11]).

This challenge could be resolved if energy-constrained wireless sensors could be deployed in a way that their capabilities are matched to application requirements at run time. Specifically, this takes the form of a matchmaking problem, where sensors should *autonomously* evaluate their energy situation and functional characteristics (e.g., what quantities can they measure, how much energy is required, how precise they are, etc.) and match these parameters—at run time—to the application requirements of the actors. In the presence of multiple sensors, it takes the form of an optimal matching problem across these sensors, where each sensor takes one or more measurement roles in the application and should dynamically adapt their role adoption if the run-time context changes either with respect to the requirements of the application or energy availability. Effectively, the required functional roles should be distributed

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand.

across the available sensors so that these fulfill the overall needs of the control or monitoring program, together.

We argue that multi-agent systems (MAS), which address the need for decentralized decision-making and social behavior, are well-suited for application-level coordination in distributed systems such as WSANs. Specifically in relation to the challenge we address, the abstraction of organizations in multi-agent-oriented programming serves to capture knowledge about the system [2, 9, 13], which the programs running on the sensor and actor nodes (i.e., the agents) can use at run time to achieve adaptive behavior.

In this work, we show that resource-constrained sensors can obtain and use an MAS organization specification to deliberate about their capabilities at run time and adopt suitable roles, in an energy- and resource-efficient manner. Our agents consequently can re-deliberate and adapt in the face of changes in the system which are reflected in the organization description. We show that these deliberations can be based on a simple market model where sensors as *sellers* evaluate offers from actors who wish to *buy* measurements.

Our main contributions in this paper are:

(1) We put forward a proposal to use MAS organizations in a manner that is amiable for use in an resource-constrained devices such as in WSANs.
(2) We propose a novel method whereby actor agents publish roles and potentially adapt them at run time.
(3) On top of these individual decisions, we show the possibility of employing a simple market model for WSANs which helps sensors and actors synchronize to together achieve the system functions.
(4) We evaluated our approach using an infrastructure consisting of software simulators, deployment with real hardware devices in a low-energy wireless network, and testing in a real-life environment. This serves to reinforce the feasibility and instill confidence in applying our approach to MAS for WSANs in theory as well as in practice.

Our results show that the agents in our system dynamically adapt to changes when available energy conditions and system-level goals are changed, and that the utilization efficiency of energy-harvesting sensors, the efficacy in fulfilling measurement roles, and the battery life of battery-powered sensors all exceeded the performance of an alternative approach that is representative of the current usage of similar systems in practice.

2 Towards a *Coordination Plane* in WSANs

With the increasing use of WSANs in various industrial domains, there are active research and standardization efforts toward enabling the creation of large, robust, and energy-efficient wireless networks [1, 35]. We are particularly interested in examining how systems with WSANs are engineered and what features offered by them support *adaptation to changes*. Run-time changes in a WSAN affect two aspects: 1. the topology and network roles of the wireless nodes, and 2. the application functions that the nodes execute [1]—in other words, their *application roles*. Energy constraints is a cross-cutting challenge that affects both these aspects. While network-level adaptation in energy-constrained wireless networks continues to be widely researched, adaptive behavior at the application level is still under-explored [1, 23, 25].

Changes in goals to be achieved by the automation system which occur at run time require the actor nodes to adapt their control programs. This, in turn, determines the kind of sensor measurements required from sensors that are part of a deployed WSAN. Conversely, the partially non-deterministic nature of sensors powered through energy harvesting requires dynamic realignment or reconfiguration of application functions in both sensor and actor nodes as the energy availability changes. Application-level coordination, which is required for adaptive behavior (in sensor and actor nodes) under the above-mentioned dynamic circumstances, still needs to be fully addressed by current standards and research [1, 10, 14].

Common application protocols used in WSANs in industry include Thread/Matter, ZigBee (both based on the IEEE 802.15.4 standard), Wirepas, Wireless HART, LoRa, and Bluetooth Low Energy (BLE) V5.0 [8, 20, 35]. To reduce the energy consumption of WSAN nodes, some of these protocols provide mechanisms for actors to subscribe to sensor measurements and thereby use them only on demand. In particular, standards like ZigBee and BLE V5.0 Mesh [4, 8, 28] provide for a context-based publish-subscribe mechanism where the system designer defines *groups* and *topics* that contextualize the bindings between sensors and actors. However, the decision about which sensors should serve which actors has to be made at design time. This is a challenging task because the designer is required to examine the energy characteristics of the individual sensors and consider the potential unreliability in energy available to the energy-harvesting sensors. This challenge has been pointed out by several authors [1, 21, 25] as being prevalent across protocols and standards in WSANs. To address the challenge, in [1] the authors suggest the need for a *coordination plane* in WSANs—a term which resonates well with the problem we are addressing.

MAS as a system-architectural paradigm has been successfully applied in the past for domains such as collaborative robotics to establish system-wide coordination in the face of uncertainty in the environment and changing requirements [6, 18]. Regarding application in the WSAN domain, [26] demonstrates how wireless sensor nodes can use energy-efficient communication during distributed reinforcement learning for control of a lighting system in a building. In [15], the authors show how WSAN that are based on MAS results in efficient fusion of sensor data. Taboun et al. [30] have modeled WSAN nodes as agents in a MAS which dynamically evaluate their available energy and decide on operational states. While these articles demonstrate and reinforce how the decentralized autonomy of MAS could support WSANs, the roles of the agents that are considered in these cases are not subject to change. It is hence assumed, and required, that the nodes in the target WSANs know, from a design-time specification, what application roles they need to fulfill; at run time, the agents only evaluate if they can meet these particular functional requirements, but they do not attempt to utilize their knowledge about own capabilities or local energy availability to assume different roles.

We argue that, in real-life systems, the measurement roles required to be fulfilled by sensors in a WSAN are subject to the tasks that the actor agents are allocated *at run time*. Therefore, upfront designation of roles leads to inefficient allocation of the sensors, which we show in our evaluation. At the same time, the allocation of roles to sensors at design time and hence without knowledge of their run-time energy availability causes an overly conservative loading of battery-powered sensors; energy-harvesting sensors, which are the more sustainable and more cost-effective option, receive a workload that is too low on average, which we also show in our evaluation. Instead of such design-time role allocation, our approach considers that the sensors themselves, equipped with local knowledge about their functional abilities and with current information about their energy availability, are in the best position to decide which role they can and should commit to. A counterargument to this allocation of responsibility to the individual sensors is that the required reasoning might not be desirable due to its potential high energy demand. However, it has recently been demonstrated that the use of a Belief-Desire-Intent (BDI) reasoner on an embedded device does not incur significant energy consumption for its reasoning cycles [31, 33]. As part of our study, we were able to confirm the findings of these articles on a real-life hardware implementation of WSAN nodes – our measurements showed no significant difference as compared to a purely reactive program[1].

Autonomous decision-making requires knowledge about the system in which an agent operates. In multi-agent oriented programming an abstraction that is suitable to capture the system requirements and design is that of MAS Organization [2, 34] – in particular, this has been shown for engineering systems [24]. Several approaches to model MAS organizations exists [6, 9, 13] which contains the notion of *groups* and *roles* used also in the design of technical systems. However, organization descriptions of real-life systems are large and their use by agents running on resource-constrained devices has not been explored.

We now describe our approach where we enabled system knowledge in the form of organization specification to be disseminated to the resource-constrained WSAN nodes, which then use it to achieve an adaptive run time.

3 Approach

In this section, we describe our novel approach wherein sensor agents on resource-constrained nodes in a MAS-based WSAN adopt functional roles at run time to collaboratively achieve system goals. In the face of limited or non-deterministic energy availability in WSANs, our approach achieves application-level coordination through decentralized deliberation on functional roles while considering the local energy availability of a sensor. With this approach, we achieve a demonstrably more favorable allocation of tasks to sensor nodes: It leads to an increase in the utilization of energy-harvesting nodes while decreasing the utilization of battery-powered nodes. At the same time, the overall efficacy of role fulfillment is increased as well. We discuss the evaluation of our approach in detail in Section 4.

In traditional system design, the roles for the actors and sensors are decided upfront, with the role of the actor depending on the role of the sensor—for instance, the role to "*maintain room temperature*" for an actor depends on the role to "*measure temperature*" for a sensor. We distinguish these two roles using the terms *actor role* and *measurement role*.

While actors may assume that measurement roles that they require will always be filled by sensors in an *unconstrained* system, the enactment of a measurement role *cannot* be taken for granted when energy-constrained sensors are involved, which is the setting we consider.

Our approach involves the actors in a WSAN (e.g., a heating controller) publishing measurement roles at run time. These role profiles specify the measurements the actors require to successfully execute their programs (e.g., maintaining a comfortable temperature in a room). For publishing these requirements, we reuse an approach that shows that such structural and functional organizational knowledge does not need to be manually coded, but can be automatically derived from engineering system descriptions [24]. To disseminate this organizational knowledge to the sensor agents on different WSAN nodes, we use a gossiping protocol, where our system is able to accommodate heterogeneous nodes—including those *without* own persistent memory—through a novel mechanism where organizational knowledge is used to manage the dissemination process itself; we describe how organizational knowledge is automatically derived and disseminated in our system in Section 3.1. Based on the role specifications in the received organizational information, our sensor agents then evaluate individually if they are capable of fulfilling a role by considering the requested *type* of measurement, measurement *frequency*, measurement *precision*, etc., where we re-use a standard vocabulary. During this matching, the nodes furthermore take into account their locally estimated energy supply and current energy situation and, if applicable, consider assuming a selected role; this process is described in more detail in Section 3.2. Finally, our approach incorporates a market mechanism, where WSAN actors announce rewards for the fulfillment of individual roles. In addition to the roles' functional requirements, which the sensor nodes use to determine whether they *can* fulfill a role, these rewards are used by the sensor nodes to determine whether they *should* adopt a role. For energy-harvesting nodes, this means that the node can optimize its utilization of (predicted) available energy, while for battery-powered nodes it supports the node in allocating its (finite) power supply to the highest-reward role; we detail this mechanism in Section 3.3.

3.1 Disseminating Organizational Knowledge

As a first step to enable the decentralized decision-making about fulfilling measurement tasks, we require a method for the actors in a WSAN to make their requirements known to sensing agents. To accomplish this, we make use of the concept of *roles* in MAS organization specifications as this corresponds well to the intent that roles model obligations of an agent [9, 13]. Concretely, we use MOISE+ [16] to model the MAS organization and represent it with RDF, using an OWL ontology for MOISE [36]. While we make use of MOISE+ in our implementation, we stress that our approach is equally applicable to any other MAS organizational models that include an abstraction that is similar to *roles*. Work by [24] has shown that it is possible to link functional and non-functional requirements of a system to the design description, and that the resulting interlinked knowledge can be used to automatically synthesize a MAS organization specification in RDF.

To enable run-time decision-making by the sensor agents about whether to assume a role (instead of relying on design-time bindings), the synthesized knowledge about the organization—including the measurement roles dynamically created by the actors—needs

[1]Implementations using both variants are available in the supplementary material

to be disseminated to the sensors in our WSAN. However, real-life installations like building automation systems are usually large, which induces (a) long and hence energy-demanding wireless pathways between nodes to communicate the organization description, and (b) very large organization descriptions. For instance, a description of the 6-floor office building and its technical system that we consider as part of the evaluation of our approach requires about 1MB of storage space in memory (as RDF in the N3 format); however, typical microcontrollers in the WSANs we consider have less than hundred kilobytes of available RAM.

Realizing that only a small part of the entire description is relevant to any single node, and that we can use the organization description itself to find out which fragment of the description is relevant for which node, we propose an extension of a gossiping protocol to only disseminate those fragments of the description to a node which are indeed relevant to the context and goals of that node. A *fragment*, in the context of organization description, is a unit of information that contains a description of a group (with or without child groups), or of one or more roles. Concretely, we build on *autonomous gossiping* (AG) [5], a gossiping protocol where nodes make autonomous decisions about dissemination, which is highly suitable for our use case. In the WSANs we consider, the nodes are energy- and memory-constrained devices whose availability and interaction opportunities are uncertain (because of varying energy input conditions) and which therefore need to be highly selective with respect to (energy-costly) wireless communication. In AG, each node *autonomously* makes the decision to *retain*, *migrate*, or *replicate* a data unit it holds in memory. Though AG targets mobile ad-hoc networks, we can reuse it with nodes that sporadically come into contact because they wake up (due to energy availability) to exchange organization-related information.

In AG and similar gossiping protocols, efficient dissemination depends on the nodes having the *right knowledge* to discern how to handle a data unit. We satisfy this requirement by using the knowledge contained in the MAS organization description *itself* for this purpose. The first piece of information that is relevant for our AG-based approach is the *group* affiliation of a node, which is known upfront in the system design and therefore contained in the organization specification and known to each node. When a node in our system wakes up, it advertises its profile according to the AG protocol, but it furthermore includes the identifiers of the groups to which it belongs. This enables a neighboring node to evaluate if a fragment of information that it currently possesses is relevant to the advertising node. In addition, a node that receives a fragment which is not of interest to it needs to decide how to treat it. In our approach, we propose that if (a) a fragment concerns a neighbor node which is incapable of retaining it, e.g., due to limited memory or energy (which is known via the advertised profile), and (b) the current hosting node can afford the memory and energy to retain it, then it opts for retention; else, in favor of conserving the energy required for its communication, it discards the fragment. In this decision-making it gives higher priority to a neighbor belonging to the same group. Such strategies can be customized, for example, depending on the application or characteristics of the network.

Our approach to use a MAS-organization-informed AG protocol in the WSAN shows that the organization description itself can serve as an in-band information source to help the agents obtain and share knowledge that they require for autonomous decision-making. In the following, we discuss how this decision-making is implemented in our approach.

3.2 Dynamic Role-creation at Run Time

Fulfilling a control or coordination role requires the actor agents to perceive the state of the underlying physical process, and for this they require the relevant sensors to measure and communicate the required data. Based on our study of the specification of measurement roles required by control strategies in the domains of building automation, factory and process automation, mobile robotics, and environmental sensing, we identified the parameters required for role definition. In our approach, the specification of these required measurements that is contained in the disseminated role profile. prominently includes a *functional specification* (FS). The FS specifies which physical quantity needs to be measured, and also gives quality attributes of these measurements (precision, frequency, etc). To motivate the need to achieve interoperability amongst heterogenous WSAN devices, we used the SSN/SOSA ontology to specify FS (see also [12]), but our approach can also be used with other ontologies. Control algorithms often specify tolerance in the acceptable quality of measurements. For instance, a heating control logic may state that the temperature measurement updates ought to be ideally every 5 min, but at least every 15 min—in our approach, the actor agent can use such flexibility to decide at run time what update rate to specify in the measurement role (e.g., if no sensing agent is willing to take up a role, then the actor might decide to lower the required update rate). Finally, the actor's control strategy may require measurements to be delivered by a specific number or type of sensors. For instance, an actor might require data from *at least three* sensors to cope with errors, or a control strategy for a critical application may explicitly require *battery-powered* sensors only. Summarizing, in our approach, an actor agent creates a role profile by including the specification for the measurement data along with additional information that states the characteristics of the engagement it expects from the sensors.

Formally, we define a measurement role as

$$R_m = \langle FS, d, N_{min}, N_{max}, r \rangle \quad (1)$$

with a functional specification (FS), a duration the role is required for (d), the minimum and maximum number of agents that are required in the role (N_{min} and N_{max}), and the reward offered for playing the role (r; this is explained in Section 3.3).

To appropriately inform the sensor agents about the requirements specified in an FS, actors and sensors are required to use a common vocabulary to maintain interoperability between actors and sensors. In our approach, we re-use an already well-known ontology—SSN/SOSA[2]—which is used to specify characteristics of sensors in industrial applications. In terms of SSN/SOSA, the core part of the FS requires minimally the *quantity kind* that needs to be measured (e.g., temperature, humidity, light-level, etc.), denoted by q. Optionally, it may include the frequency of measurements, frequency of updates, precision, etc. Amongst these parameters, the measurement and update frequencies (f_m and f_u) are particularly

[2]https://www.w3.org/TR/vocab-ssn/

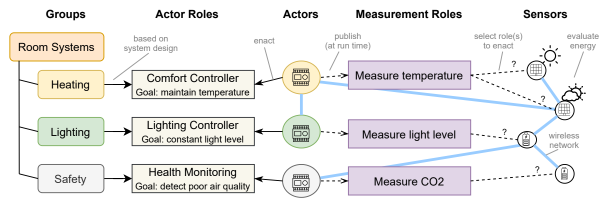

Figure 1: Given their energy situation, how do the sensors evaluate which roles are suitable for them?

important for calculating the measurement energy cost, and hence implemented in our system:

$$FS = \langle q, f_m, f_u \rangle \tag{2}$$

Summarizing, in our system, a measurement role description R_m with the required elements from Equation 1 is created (or updated) by an actor and then added to the list of available roles in the actor's organization group of the organization specification. Through our organization-informed AG protocol, this information is then disseminated to the other agents in the group.

3.3 A Market of Actors and Sensors

Having obtained the required information about roles available in its organization group, a sensor needs to decide which of them to enact. It makes this decision based on the role description of a measurement role R_m (see Equation 1), which enables the agent to evaluate if it can satisfy the requirements and how costly this is for them in terms of energy.

In a first step, the sensor agent checks if its sensor hardware provides for measurement of the quantity kind q. If so, it computes the energy required to enact the role, E_m. E_m is calculated by the sensing agents based on q, f_m, and f_u:

$$E_m = estimateEnergy(q, f_m, f_u) \tag{3}$$

$estimateEnergy$ hence considers the sensor's local knowledge to calculate the amount of energy it estimates that fulfillment of R_m would cost it in terms of sensing, processing, and network communication. Our sensors next compare the required energy E_m to their available energy, E_a. E_a might consider the current battery charge or current energy yield of the sensor's energy harvesting system, or the agent might predict the amount of energy it expects to have available in the future—the details of how E_a is computed are hence up to the sensor. After calculating E_a, *energy-harvesting sensors* in our system computes the *cost* of enacting a role by comparing it to the energy required for measurement, E_m:

$$C(R_m) = \frac{E_m}{E_a} \tag{4}$$

For *battery-powered sensors* in our system, their E_a is high at the start (as the battery has full charge) but is *non-renewable* and therefore the above equation will result in the sensor at first perceiving a too low cost for enacting any role, and consequently expend its charge even when not required (i.e., when other sensors with a renewable energy source could have fulfilled the role). To ensure that the battery-powered sensors enact measurement roles only as a last resort, they are configured to consider a role only when the reward exceeds a certain threshold.

Complementary to the sensing agents computing the cost of *enacting* a role, the actor agents offer a *reward r* for role fulfillment. We hence adopt a simple market paradigm which has demonstrated benefits in architecturally distributed systems [29, 32, 37]. Applying this market metaphor to our WSAN, we consider sensors as analogous to *sellers* of goods (i.e., measurement data) which are produced using energy that is often scarce and non-deterministic, while actors are the *buyers* who benefit for their own purpose by acquiring the goods. Since this model is agnostic of the domain, communication protocol, and network topology, we also see it as a general approach [32] for the allocation of distributed resources whose availability is not deterministic; this supports that, in the event of scarcity, the most needed requirements are served first. [3] shows that this functions work well for distributed manufacturing scenarios with benevolent resource producers (sellers)—i.e., where these sellers neither speculate nor rescind from commitments for higher gains.

Since the actor agents are not aware of the actual cost of enacting a role (we consciously maintain a separation of concerns here), the offered reward is normalized. A reward value of 1.0 expresses that the actor is willing to pay the sensor's cost price. In our model, the actor agents hence do not benefit by obtaining measurements at below par price, and therefore, they begin with a reward value of 1.0. In the event that a harvesting-based sensing agent possesses the requisite energy budget, it subsequently computes its benefit for that designated role using the following simple formula:

$$B(R_m, E_a, E_m, r) = \begin{cases} r - C(R_m), & \text{if } E_a \geq E_m \\ 0, & \text{otherwise} \end{cases}$$

Battery-powered sensors adopt a role when the reward exceeds a (pre)determined threshold. An optimal determination of this threshold was not investigated within the scope of this work; we propose that this should be computed dynamically during run time.

Regarding market dynamics, the actor agents gradually increase their rewards r for a role if that role remains unfulfilled. Dynamicity is supported through the dissemination mechanism discussed in Section 3.1 to make role enactment, modifications to roles (including to rewards), and new roles known across the WSAN.

In our approach the sensing agents do not exit a role mid-way to adopt another role that has a higher benefit. By remaining committed to a role, we not only avoid the energy overhead of switching roles but also see higher overall efficacy in fulfillment of roles.

4 Evaluation

The goal of our evaluation is to compare our approach, which we refer to as *Dynamic RolE Allocation Method* or **DREAM**, to an existing and in-use standard in the industry to show how each fares in terms of utilization of the sensors in fulfillment of measurement requirements in the system.

4.1 Deployment

For the comparison, we chose a deployment based on the *BLE V5.0 Mesh* standard, which is widely used both in building and factory automation [4, 8]. Sensors and actors in a BLE V5.0 Mesh are provisioned to be part of *groups*. Sensors can publish measurements to *topics* in their groups, and correspondingly, actors subscribe to one or more topics. In other words, the measurement roles of the sensors are static and decided at design time. This BLE-based deployment is furthermore representative of a whole class of WSAN protocols where actors are pre-engineered with the knowledge about a set of sensors that they may use at run time (described in Section 2), and we hence claim that our results can be generalized for such protocols. We use the term *static role allocation* or **SRA** to refer to this approach.

4.1.1 Evaluation Assumptions To ensure that our evaluation can be generalized across domains, configurations, and use cases, we consider the following dimensions:

(1) Input Energy Profile: Based on a study of energy available to typical indoor PV-powered sensors [27], we identified three broad kinds of energy profiles (EP): EP1 peaks at approximately mid-day, EP2 peaks before noon (around 10:00), and EP3 peaks in the afternoon (around 14:00). These profiles represent heterogeneous energy conditions, for example caused by placement of the PV array with respect to light source.
(2) Measurement Role Kind: We studied standard control algorithms for heating, lighting, and environment monitoring in building automation systems—for commercial and industrial buildings, but also generally for any automation application [22]. Based on this study, we identify three broad kinds of measurement requirements of control programs: MP1 is on-demand, low-frequency, fixed duration, and tolerant to missed updates (e.g., a comfort-oriented heating system); MP2 is on-demand, high-frequency, short duration, intolerant to missed updates (e.g., a lighting system); and MP3 is low-frequency, undefined duration, tolerant to missed updates (e.g., a continuous air-quality measurement system).
(3) Network Topology: While our overall approach is agnostic to the network topology, the performance of the dissemination protocol is affected by topology. In our evaluation, and informed by typical layouts that occur in building automation systems [28], we consider a mesh network topology, where individual meshes might be connected through a star topology at the system level.
(4) Device Capability: We selected an industrial-grade hardware platform from Nordic Semiconductors[3] which is representative of the state of the art hardware technology for devices in WSAN.
(5) Application Domain: We selected building automation as our application domain as it presents a good diversity of control strategies used by actors and faces changes in requirements and design during the life of the system. We claim that our results generalize to domains with similar characteristics: low-power sensor nodes with fixed positions and where measurement specifications as well as available energy may change at run time. While we hypothesize that some of our results would also generalize to mobile settings, our experiments do not include this.

[3]https://www.nordicsemi.com/Products/Development-hardware/Nordic-Thingy-53

Based on these criteria, we carefully chose a test scenario which reflects a real-life use case. The test was first conducted in a simulation environment and then deployed in a large shared-space office room. The hardware deployment of our approach was based on the Thingy53 development platform from Nordic Semiconductors on which the agent programs were deployed using the Embedded BDI SDK [4]. The devices used the OpenThread [5] as the network protocol stack with CoAP as the application-layer protocol. Each node was both a CoAP client and server. The nodes used link-local multicast on the Thread network to send out the *advertise* message for implementation of the AG protocol. During initialization, the organization description was seeded in a fully-powered device in the network. For further details about our implementation, we point the interested reader to the documentation in the supplementary material of this paper. Furthermore, the software simulation environment, the code for implementation on embedded devices, the organization description synthesized from engineering information, and the results of our experiments are all available as supplementary material for interested readers to examine and reuse.

4.1.2 Evaluation Setup Our automation system was deployed in an office room that requires control of heating and lighting systems along with monitoring of air quality and humidity. To accomplish this, the room is equipped with five wireless sensing nodes, each of which can measure temperature, humidity, light level, and CO2. Three of the sensors are energy-harvesting (EH) and powered by PV cells that are coupled with a small energy buffer (a capacitor), whereas two of the sensors are battery-powered (BP). Due to their placement in the room, each of the three energy-harvesting sensors are subject to a different input energy profile (EP1/EP2/EP3)

The actor agent for the heating system, when activated, requires temperature and humidity measurement every 5 minutes for a duration of 1 hour, which corresponds to measurement profile MP1; the lighting system requires light level measurements every 1 minute for a duration of 1 hour (i.e., MP2); and the health safety monitoring agents require humidity and air-quality measurement every 1 hour throughout the day, corresponding to MP3.

The tests on the real hardware deployment in the office were run for a period of 1 week during which we also confirmed that the simulation model matched the real-life deployment. We then used the simulation environment to flexibly vary the input energy conditions, for instance to subject our system to seasonal variations that affect the amount of light incident on the PV-powered sensors, helping us to verify that our approach does not miss any corner cases.

4.2 Results

We considered the following metrics in our evaluation:

(1) **Utilization of EH Sensors**: We define this as the ratio of energy used for measurement to the amount of energy available from the environment in an EH sensor over n sampling periods: $\eta = \frac{\Sigma_{i=1}^{n} E_{m_n}}{\Sigma_{i=1}^{n} E_{a_n}}$. Ideally, η should be close to 1 if there are sufficient roles to be fulfilled.

[4]https://embedded-bdi.github.io/
[5]https://openthread.io/

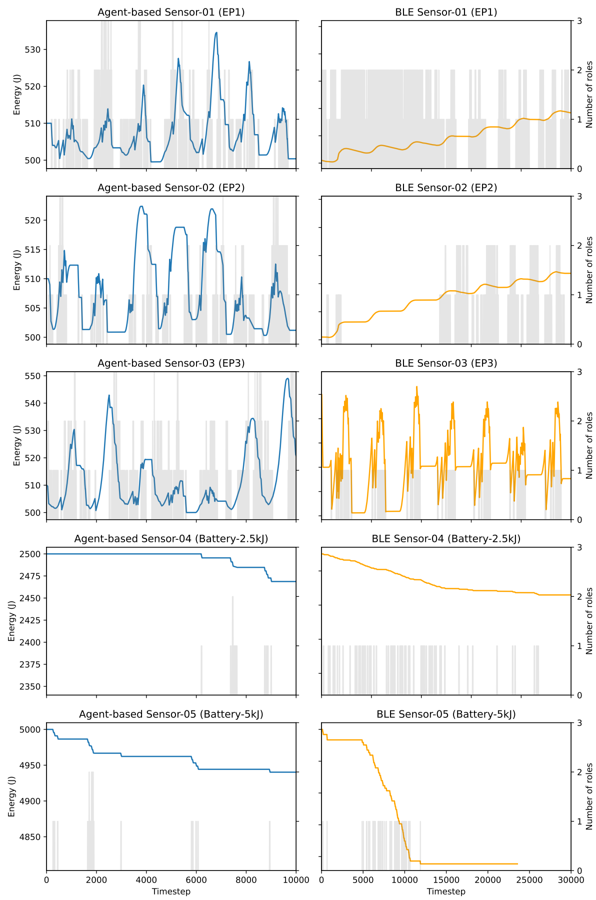

Figure 2: Sensor utilization across each of our five sensors. Energy in the buffer of each of five sensors (colored lines; left axis) and number of roles adopted by each sensor (grey bars; right axis). Left Column: DREAM; Right Column: SRA.

(2) **Role Fulfillment Efficacy**: For a role published by an actor agent, we expect at least one sensor to deliver the required measurements. We define average efficacy $\epsilon_{avg} = \frac{\Sigma_{i=1}^{k} \frac{N_{u_k}}{N_{e_k}}}{k}$ with the number of expected updates N_{e_k} and the number of received updates N_{u_k} by an actor in the kth instance when a role was published.

(3) **Battery Consumption**: For battery-powered sensors, we measure the discharge rate $\gamma_{bat} = \frac{\Delta Q}{T_d}$ where ΔQ is the energy discharged over the period T_d of the experiment.

(4) **Impact on Hardware Platform**: We verify if the implementation of the agent program on resource-constrained hardware is feasible from the viewpoint of memory footprint (in flash M_f and RAM M_{ram}), time required for computation during each wake cycle T_{wake}, and consequently, the energy consumed E_{wake} during this period (this includes energy required for sensing and communication of measurement data, i.e., E_m).

	DREAM	SRA
Sensor 1 (EP1)	100%	37%
Sensor 2 (EP2)	100%	31%
Sensor 3 (EP3)	95%	100%
Sensor 4 (BAT 2.5kJ)	1% (25J)	2% (50J)
Sensor 5 (BAT 5.0kJ)	1% (50J)	4% (200J)

Table 1: Our approach (DREAM) showed higher (almost 100%) utilization of the harvested energy while using the battery-powered sensors sparingly.

	Approach					
	DREAM			SRA		
Scenario	**EH**	**BP**	**Total**	**EH**	**BP**	**Total**
MP1	84%	0%	84%	51%	34%	85%
MP2	88%	12%	100%	64%	27%	91%
MP3	93%	7%	100%	30%	70%	100%

Table 2: Comparing the efficacy of role fulfillment achieved by our approach in comparison to the SRA-based deployment.

4.2.1 Utilization of EH Sensors Figure 2 shows the state of the energy storage buffers of the five sensors while comparing our approach to a SRA-based deployment.

At a high level, the graphs show that with our approach, the energy-harvesting sensor were productively using the energy while being engaged in roles published by the actors. The battery-powered sensors were only involved for roles which were published when none of the energy-harvesting sensors had energy available for the new role(s), which is the behavior that we had desired to achieve with the proposed approach.

Compared to our approach, in the SRA-based deployment, sensors 1, and 2, were heavily under-utilized. Consequently, the battery-powered sensors were being used unnecessarily (therefore, draining them faster).

Table 1 summarizes the performance our approach in comparison to a deployment using the SRA-based solution. The results show that the energy-harvested sensors utilized (η) on average was 98% of the available energy to enact roles as compared to about 60% for the deployment with BLE devices. These differences are expected because in our approach the sensors autonomously decide on the role to enact with the goal of maximising their benefit B, which hence means that they strive to maximise their utilization. In the alternative approach, the sensors' roles are statically assigned at design time, which prevents the system as a whole from exploiting run-time information (e.g., about current energy availability). In addition, the actor agents dynamically adjust the rewards, and potentially, the functional specification F in case a role remains unfulfilled. Such collaborative effort results in the high utilization of the energy available in energy-harvesting sensors.

4.2.2 Efficacy The efficacy of fulfilling role requirements (ϵ_{avg}) was on an average 95% with our approach whereas in the case of BLE based deployment it was 92% (See Table 2). Though this difference does not appear to be significant, we point out that the in case

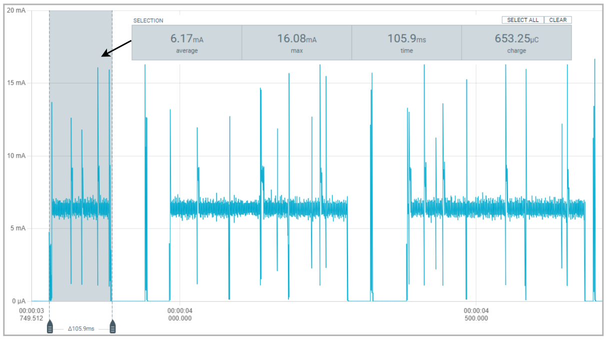

Figure 3: The part highlighted in gray is where the sensing agent is updating its organization fragment

of changes in system design, the SRA-based solution needs to be manually re-engineered, whereas our approach relies on knowledge available at run time to determine the best possible matchmaking between the sensors and roles published by the actors. More importantly, the efficacy achieved by the SRA-based deployment was at the cost of the battery-powered sensors (as described further in the following sub-section) – together with the results above indicating higher utilization of the energy-harvested sensors, this demonstrates the difficulty in achieving an optimal matchmaking solely based on design-time knowledge. Therefore, our approach demonstrates an autonomously adaptive behavior which achieves efficacy which is, in our evaluation, higher than the traditional solution and does not require manual engineering.

4.2.3 Battery Life Since our approach results in higher utilization of the energy-harvesting sensors, its impact is seen in the lowered demand on battery-powered sensors. Graphs in Figure 2 shows that the battery-powered sensors using our approach are enacting roles only when the the energy-harvesting sensors are low on available energy. This happens because an actor (which has published the role) increases the offered reward if a role remains unfulfilled and the battery-powered sensors are parameterized with a threshold for the offered reward only above which it considers a role for enactment. Raising this threshold has impact on the efficacy because the battery-powered sensors wait longer before electing to enact a role. We highlight in section 4.3 that peer-to-peer collaboration amongst the sensors can avoid this and thereby improve the efficacy even further.

With our approach energy demand on the batteries (γ_{bat}) was almost a third of what the SRA-based approach requires. For the test scenario (i.e. the office room), extrapolating from our evaluation results, the proposed approach achieves an expected lifetime of 3 years for both battery types, whereas for the BLE-based solution, it is 2 years (the calculation of battery life also includes its inherent degradation and therefore is not proportional to the energy consumed by the load).

4.2.4 Impact on Hardware Platform Figure 3 shows the trace of energy consumption for the embedded hardware device implementing our approach with the overhead incurred by the gossipping protocol highlighted. On the whole, compared to the SRA-based implmentation, our approach consumes about 10% more energy during each wake cycle when it listens for gossip messages. This additional expenditure of energy reaps benefit because by being *actively* aware of the changes in the organization, the sensing agent is able demonstrate an adaptive behavior, which in turn results in higher utilization of the sensors. However, we recognize that a more efficient gossiping protocol targetting the use case could result in lower energy overhead and consider this for future work.

The difference in memory requirement of our embedded BDI-based program in comparison to a standard implementation of BLE device was not significant: Our approach required $M_f = 528kB$ of Flash and allocated $M_{ram} = 395kB$ of SRAM in comparison to the BLE-based implementation where the corresponding values were 673 kB and 411 kB respectively (all reported values are with compiler optimizations enabled).

Therefore our approach is suitable for implementation on resource-constrained embedded hardware platforms.

4.3 Summary and Discussion

The evaluation of our approach has shown that autonomous decision-making (and problem-solving behavior) of agents in MAS can be applied advantageously to WSANs. During our study, we realized that this opens up several other interesting avenues for future exploration. We highlight some of these below.

Reputation Management : Agents could potentially be unreliable for e.g., a sensor elects to enact a role but then does not update at the requested frequency or quality. In such cases, the organization model can provide for reputation as quantifiable metric to be associated with the agents.

Collaborative Role Enactment : Sensors could form coalitions to collectively optimize the use of their available energy. For this purpose, the market model could be augmented with a method analogous to production planning and optimization.

Applying Multi-agent reinforcement learning : Sensing agents can, over time, gather and share knowledge about evolution of their energy availability. Such information can enable the sensing agent to strategically adopt roles to optimize its overall reward.

5 Conclusions

We have seen that fulfilling the measurement requirements by low-power sensors in WSANs while maximizing the utilization of harvested energy and minimizing the use of battery-powered sensors is challenging because it demands manual re-engineering as requirements and ambient energy conditions change. Our approach has shown that dynamic and decentralized adaptation of roles at run time can be achieved by modeling the sensors and actors as autonomous agents in a MAS to which we disseminate system knowledge in the form of organization description. The agents use this knowledge at run time to employ a simple market model to dynamically allocate roles. Our results which were based on real-life deployment and extensive simulations showed significant improvement in utilization of harvested energy while reducing the reliance on battery-powered sensors. We foresee that the principal impact of applying MAS to this and similar problems in engineering is that it helps the adoption of cutting-edge research in industry.

References

[1] Ian F Akyildiz and Ismail H Kasimoglu. 2004. Wireless sensor and actor networks: research challenges. *Ad hoc networks* 2, 4 (2004), 351–367.
[2] Estefania Argente, Vicente Julian, and Vicente Botti. 2006. Multi-agent system development based on organizations. *Electronic Notes in Theoretical Computer Science* 150, 3 (2006), 55–71.
[3] Albert D Baker. 1996. Metaphor or reality: A case study where agents bid with actual costs to schedule a factory. *Market-based control: A paradigm for distributed resource allocation* (1996), 184–223.
[4] Seyed Mahdi Darroudi and Carles Gomez. 2017. Bluetooth low energy mesh networks: A survey. *Sensors* 17, 7 (2017), 1467.
[5] Anwitaman Datta, Silvia Quarteroni, and Karl Aberer. 2004. Autonomous gossiping: A self-organizing epidemic algorithm for selective information dissemination in wireless mobile ad-hoc networks. In *International Conference on Semantics for the Networked World*. Springer, 126–143.
[6] Virginia Dignum and Frank Dignum. 2001. Modelling agent societies: Coordination frameworks and institutions. In *Progress in Artificial Intelligence: Knowledge Extraction, Multi-agent Systems, Logic Programming, and Constraint Solving 10th Portuguese Conference on Artificial Intelligence, EPIA 2001 Porto, Portugal, December 17–20, 2001 Proceedings 10*. Springer, 191–204.
[7] Stefan Draskovic and Lothar Thiele. 2021. Optimal power management for energy harvesting systems with a backup power source. In *2021 10th Mediterranean Conference on Embedded Computing (MECO)*. IEEE, 1–9.
[8] Leonardo Eras, Federico Domínguez, and Caril Martinez. 2022. Viability characterization of a proof-of-concept Bluetooth mesh smart building application. *International Journal of Distributed Sensor Networks* 18, 5 (2022), 15501329221097819.
[9] Jacques Ferber, Fabien Michel, and José Báez. 2005. AGRE: Integrating environments with organizations. In *Environments for Multi-Agent Systems: First International Workshop, E4MAS 2004, New York, NY, July 19, 2004, Revised Selected Papers 1*. Springer, 48–56.
[10] Christian Frank and Kay Römer. 2005. Algorithms for generic role assignment in wireless sensor networks. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*. 230–242.
[11] Vehbi C Gungor and Gerhard P Hancke. 2009. Industrial wireless sensor networks: Challenges, design principles, and technical approaches. *IEEE Transactions on industrial electronics* 56, 10 (2009), 4258–4265.
[12] Armin Haller, Krzysztof Janowicz, Simon JD Cox, Maxime Lefrançois, Kerry Taylor, Danh Le Phuoc, Joshua Lieberman, Raúl García-Castro, Rob Atkinson, and Claus Stadler. 2019. The modular SSN ontology: A joint W3C and OGC standard specifying the semantics of sensors, observations, sampling, and actuation. *Semantic Web* 10, 1 (2019), 9–32.
[13] Mahdi Hannoun, Olivier Boissier, Jaime S Sichman, and Claudette Sayettat. 2000. MOISE: An organizational model for multi-agent systems. In *Ibero-American Conference on Artificial Intelligence*. Springer, 156–165.
[14] Navid Heydarishahreza, Saeed Ebadollahi, Reza Vahidnia, and F John Dian. 2020. Wireless sensor networks fundamentals: a review. In *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 0001–0007.
[15] Khin Haymar Saw Hla, YoungSik Choi, and Jong Sou Park. 2008. The multi agent system solutions for wireless sensor network applications. In *Agent and Multi-Agent Systems: Technologies and Applications: Second KES International Symposium, KES-AMSTA 2008, Incheon, Korea, March 26-28, 2008. Proceedings 2*. Springer, 454–463.
[16] Jomi Fred Hübner, Jaime Simao Sichman, and Olivier Boissier. 2002. MOISE+ towards a structural, functional, and deontic model for MAS organization. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*. 501–502.
[17] Neal Jackson, Joshua Adkins, and Prabal Dutta. 2019. Capacity over capacitance for reliable energy harvesting sensors. In *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*. 193–204.
[18] Nick R Jennings. 1993. Commitments and conventions: The foundation of coordination in multi-agent systems. *The knowledge engineering review* 8, 3 (1993), 223–250.
[19] Philipp H Kindt, Daniel Yunge, Robert Diemer, and Samarjit Chakraborty. 2020. Energy modeling for the bluetooth low energy protocol. *ACM Transactions on Embedded Computing Systems (TECS)* 19, 2 (2020), 1–32.
[20] Xiaomin Li, Di Li, Jiafu Wan, Athanasios V Vasilakos, Chin-Feng Lai, and Shiyong Wang. 2017. A review of industrial wireless networks in the context of Industry 4.0. *Wireless networks* 23 (2017), 23–41.
[21] Tommaso Melodia, Dario Pompili, Vehbi C Gungor, and Ian F Akyildiz. 2007. Communication and coordination in wireless sensor and actor networks. *IEEE Transactions on mobile computing* 6, 10 (2007), 1116–1129.
[22] Ross Montgomery and Robert McDowall. 2008. *Fundamentals of HVAC control systems*. Elsevier.
[23] Virginia Pilloni, Huansheng Ning, and Luigi Atzori. 2021. Task allocation among connected devices: Requirements, approaches, and challenges. *IEEE Internet of Things Journal* 9, 2 (2021), 1009–1023.
[24] Ganesh Ramanathan. 2023. Synthesizing Multi-agent System Organization from Engineering Descriptions. In *Engineering Multi-Agent Systems*, Andrei Ciortea, Mehdi Dastani, and Jieting Luo (Eds.). Springer International Publishing, Cham.
[25] Hamidreza Salarian, Kwan-Wu Chin, and Fazel Naghdy. 2012. Coordination in wireless sensor–actuator networks: A survey. *J. Parallel and Distrib. Comput.* 72, 7 (2012), 856–867.
[26] Jaspal S Sandhu, Alice M Agogino, Adrian K Agogino, et al. 2004. Wireless sensor networks for commercial lighting control: Decision making with multi-agent systems. In *AAAI workshop on sensor networks*, Vol. 10. Citeseer, 131–140.
[27] Lukas Sigrist, Andres Gomez, and Lothar Thiele. 2019. Dataset: Tracing indoor solar harvesting. In *Proceedings of the 2nd Workshop on Data Acquisition to Analysis*. 47–50.
[28] Jolly Soparia and Nirav Bhatt. 2014. A survey on comparative study of wireless sensor network topologies. *International Journal of Computer Applications* 87, 1 (2014).
[29] Ken Steiglitz, Michael L Honig, and Leonard M Cohen. 1996. A computational market model based on individual action. *Market-Based Control* (1996), 1–27.
[30] Mohammed S Taboun and Robert W Brennan. 2017. An embedded multi-agent systems based industrial wireless sensor network. *Sensors* 17, 9 (2017), 2112.
[31] Danai Vachtsevanou, Jannik William, Matuzalém M dos Santos, Maiquel De Brito, Jomi Fred Hübner, Simon Mayer, and Andres Gomez. 2023. Embedding Autonomous Agents into Low-Power Wireless Sensor Networks. In *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 375–387.
[32] Michael P Wellman. 1996. Market-oriented programming: Some early lessons. *Market-based control: a paradigm for distributed resource allocation* (1996), 74–95.
[33] Jannik William, Matuzalém Muller dos Santos, Maiquel de Brito, Jomi Fred Hübner, Danai Vachtsevanou, and Andres Gomez. 2022. Increasing the Intelligence of low-power Sensors with Autonomous Agents. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*. 994–999.
[34] Michael Wooldridge, Nicholas R Jennings, and David Kinny. 2000. The Gaia methodology for agent-oriented analysis and design. *Autonomous Agents and multi-agent systems* 3 (2000), 285–312.
[35] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. 2008. Wireless sensor network survey. *Computer networks* 52, 12 (2008), 2292–2330.
[36] Alexandra-Madalina Zarafin, Antoine Zimmermann, and Olivier Boissier. 2012. Integrating Semantic Web Technologies and Multi-Agent Systems: A Semantic Description of Multi-Agent Organizations.. In *AT*. 296–297.
[37] Andrew T Zimmerman, Jerome P Lynch, and Frank T Ferrese. 2013. Market-based resource allocation for distributed data processing in wireless sensor networks. *ACM Transactions on Embedded Computing Systems (TECS)* 12, 3 (2013), 1–28.