Sustainable Computing: Informatics and Systems xxx (2012) xxx–xxx

Contents lists available at SciVerse ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

QuARES: A quality-aware renewable energy-driven sensing framework

Nga Dang*, Elaheh Bozorgzadeh, Nalini Venkatasubramanian

Computer Science Department, University of California, Irvine, USA

ARTICLE INFO

Article history:
Received 9 February 2012
Accepted 17 August 2012

Keywords:
Energy harvesting
Wireless sensor network

ABSTRACT

Renewable energy technologies have become a promising solution to reduce energy concerns that arise due to limited battery in wireless sensor networks. While this enables us to prolong the lifetime of a sensor network (perpetually), the realization of sustainable sensor platforms is challenging due to the unstable nature of environmental energy sources. In this paper, we propose an adaptive energy harvesting management framework, QuARES, which exploits an application's tolerance to quality degradation to adjust data collection quality based on energy harvesting conditions. The proposed framework consists of two phases: an offline phase which uses prediction of harvested energy to allocate energy budget for time slots; and an online phase to tackle fluctuations in the time-varying energy harvesting profile. We formulate the energy budget allocation problem as a linear programming problem and implemented heuristics to minimize error in data quality at runtime. Our techniques are implemented in a network simulator, QualNet. In comparison with other approaches (e.g., [8]), our system offers improved sustainability (low energy consumption, no node deaths) during operation with data quality improvement ranging from 30 to 70%. QuARES is currently being deployed in a campus-wide pervasive space at UCI called Responsphere [11].

1. Motivation

In energy harvesting systems, energy is derived from environmental sources such as sunlight, wind and heat. Renewable energy sources allow us to continuously harvest energy from the environment, providing a constant and perpetual source of energy to the systems they drive. Despite its low efficiency, renewable energy technology is a viable and promising solution for low power wireless sensor network systems (WSN). WSNs have found application in areas ranging from environmental monitoring, health monitoring, and smart building to military applications (see Akyildiz et al. [29] for a survey of WSN applications). We are particularly interested in data collection applications – here, sensors collect information about their surrounding environments; update this information at a server via a base station through single-hop or multi-hop adhoc networks and respond to frequent or sporadic monitoring requests from users. It is desirable that data collection process satisfies the quality (accuracy, timeliness) and reliability needs (no dead nodes) of the application at hand.

Scavenging energy from surrounding environments permits such smart sensor infrastructures to function indefinitely by eliminating the need for frequent battery changes; in the long term, this enables the deployment of sustainable and manageable infrastructures. Recent research prototypes have been developed that couple wireless sensor motes with solar cells enabling them harvest energy from surrounding environments for continuous operation [1–3]. Energy-harvesting capable wireless sensor networks have obvious advantages. They are self-sustainable, perpetually powered by replenishable energy sources from the surrounding environment. The systems hence become autonomous and life-time of the system is only limited by the robustness of the underlying hardware. Another benefit is the low maintenance cost associated with managing the infrastructure especially when the size of sensor network can span several orders of magnitude, ranging from hundreds to thousands or even millions of nodes [29]. In some cases, it is infeasible to change batteries on sensor networks deployed in inaccessible locations such as volcanoes or battlefields.

However, the ability to leverage renewable energy effectively depends on a variety of factors. Renewable energy sources exhibit both temporal and spatial variations. There could be low or no availability of replenished energy for extended periods of time, e.g., night time for solar energy. Environmental conditions dictate the availability of energy sources (e.g., harvested solar energy on a sunny day is much higher than it is on a cloudy or rainy day). Fig. 1a depicts the solar energy profile in Los Angeles during a week in September, 2011 [20]. Fig. 1b [31] illustrates widely different solar irradiance levels at locations within close proximity of each other. In addition to variation in harvesting capabilities, there is also variation in sensing needs (due to events and activities) and application requirements (accuracy of sensing required for the target application). Queries might arrive in bursts, events occur randomly and

* Corresponding author. Tel.: +1 9493945560.
E-mail addresses: ntdang@ics.uci.edu, ngad@uci.edu (N. Dang), eli@ics.uci.edu (E. Bozorgzadeh), nalini@ics.uci.edu (N. Venkatasubramanian).

2210-5379/$ – see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.suscom.2012.08.001

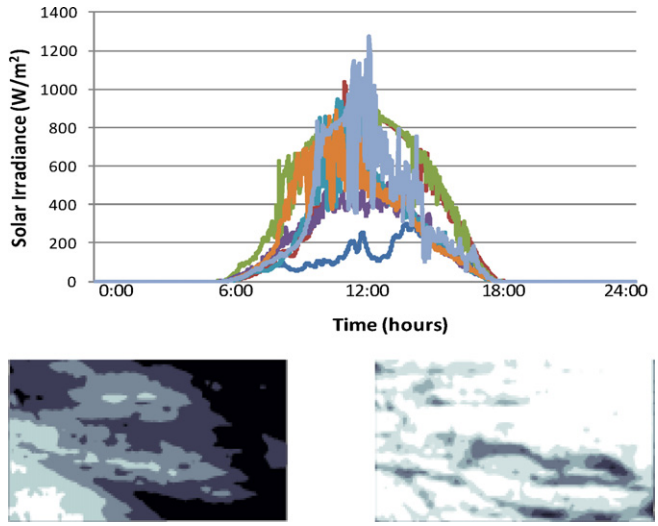

Fig. 1. (a) Temporal variation and (b) spatial variation in solar energy profile [20,31].

not evenly distributed in space and application requirements might change based on context.

This dynamicity of energy supplies and heterogeneity of application needs create new and exciting research problems in harvesting-capable wireless sensor networks. In particular, it creates a shift in research focus from energy-efficient to energy-neutral approaches, i.e., from optimizing energy consumption and prolonging battery life-time to optimally adapting systems to deal with unstable energy sources. Designing a sustainable wireless sensor network (with replenishable but fluctuating energy supply) while optimizing the data quality of applications using the sensed information is a formidable challenge. To tackle this challenge is the main target of our work.

In this paper, we propose QuARES, quality-aware renewable energy-driven sensing framework for data collection applications in energy harvesting wireless sensor networks. To the best of our knowledge, this work is the first attempt to jointly use both application data quality and harvesting ability to perform energy management for such systems. Our framework includes two phases: an offline slot-based energy budget allocation algorithm and an online adaptation strategy. The offline phase exploits predictability of the energy harvesting sources and data quality-energy tradeoff to allocate slot-based energy budgets in a harvesting period (e.g., hours in a day or days in a week). In the online phase, online adaptation policies are proposed to guarantee timely responses to queries in spite of the time-varying characteristics of harvested energy.

Energy-neutral design in itself is not a new concept; this approach has been proposed in literature to cope with the time-varying characteristics of the energy harvesting profile. Many of these efforts adopt a cross-layer design approach; i.e., considering energy status at battery layer and adapting system at other layers. At the node layer, Hsu et al. [4] adapt duty cycling of systems to the changes in renewable sources. Hasenfratz et al. [10,35,36] modify routing protocols at the MAC layer to exploit both temporal and spatial variations of renewable energy and maximize data delivery rate for sensors. Voigt et al. [7] adapt LEACH, a cluster-based routing protocol for sensor networks to take advantage of energy harvesting. At operating system layer, Moser et al. [6] and Liu et al. [5] propose task scheduling techniques for energy harvesting systems. At the application layer, Wang et al. [9] use an adaptive technique to turn on and off storage services based on different energy thresholds. Ravinagarajan et al. [12] adapt task utility of structural health monitoring applications to maximize accuracy of tasks while sustaining the system.

What is unique about our approach is the fact that we consider applications with tolerance to quality degradation and exploit data quality-energy tradeoffs in designing energy-neutral approaches for managing WSNs. Refs. [21,22,37] attempt to maximize or fairly assign data rate for data collection in wireless sensor networks; the data rate per se does not directly reflect data quality; changing data rates without being aware of data quality requirements might lead to unsatisfying results. Techniques for energy-efficient data collection [14,32,33] optimize energy consumption to prolong life-time of finite charge batteries and as a result, life-time of the whole sensor network. Han et al. [14] utilize an application's tolerance to quality degradation to minimize energy consumption by maintaining lowest possible data quality. This approach is designed for and suited to traditional continuously discharging battery systems. In contrast, our work, exploits an application's tolerance to quality degradation to adapt the system to the fluctuations of renewable energy sources while simultaneously optimizing data quality.

Specific contributions include:

(1) Exploitation of application tolerance to quality degradation to adapt the sensor data collection process under unstable energy harvesting conditions (Section 2).
(2) Design of the QuARES framework, an energy harvesting management framework with 2 phases (online and offline) that utilizes energy harvesting prediction and knowledge of data quality–energy cost to maintain system sustainability and optimize data quality (Section 4).
(3) Deployment of an energy-harvesting sensor network test-bed at UCI to carry out a measurement study for indoor energy harvesting wireless sensor networks (Section 5)
(4) Evaluation of QuARES in comparison to other state-of-the-art offline/online strategies (fixed-error-margin, minimum variance [8]) under different application and energy harvesting scenarios. QuARES is implemented in QualNet simulator (here, the battery model was modified to simulate energy harvesting). Results show that our framework can tolerate lower error margins (i.e., higher data accuracy) of 30–70%, ensure a response to all queries; additionally, sensors do not have to shut down to replenish (Section 6.2).
(5) Illustration of our simulator as a valuable tool for designers to explore the design space by applying QuARES for indoor energy harvesting sensor network scenarios (Section 6.3).

2. The QuARES approach

In this section, we will first describe our wireless sensor network system and then the observations leading to the design of the complete QuARES framework.

Our system consists of a wireless sensor network (WSN) deployed in an infrastructure for monitoring purposes. The components of our systems are a set of n sensor nodes $\{s_1, s_2, \ldots, s_n\}$ and a base station(s) B as shown in Fig. 2. Each sensor node has a processor with limited memory, an embedded sensor(s), an analog-to-digital converter and a radio circuitry. Sensor node s_i collects information about its surrounding environment by reading a value

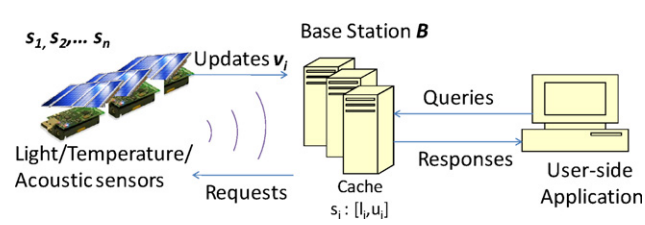

Fig. 2. Data collection in a wireless sensor network.

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

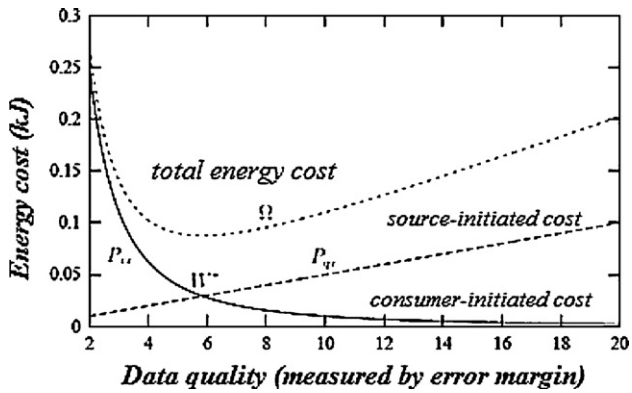

Fig. 3. Data quality–energy cost tradeoff.

v_i from its embedded sensor and periodically sends an update to the base station(s). v_i is a property of the environment, e.g., temperature, humidity or sound, that the application needs to collect to monitor the environment through our WSN. In this study, we assume that all sensor nodes are equipped with a harvesting circuitry. Harvested energy is accumulated in an energy buffer that supplies power for a sensor node's operation.

A base station B resides at a node with unlimited resources, e.g., power, storage, computation. It collects data from sensor nodes and stores them in a cache. The cache contains an approximation range $[l_i, u_i]$, a range based representation for each sensor node s_i. The base station B is connected to a monitoring application on the user side. The application periodically polls sensor nodes through the base station(s) for the monitored phenomenon. When necessary, the application can ask for sporadic information. In particular, the application sends a query Q_j to the base station each time it needs data from a sensor node or a set of sensor nodes. Each query Q_j contains data quality constraints. If the approximation range $[l_i, u_i]$ for sensor s_i in the cache satisfies these constraints, base station B returns an approximated value to the monitoring application. Otherwise, B sends an update request to retrieve latest value v_i from sensor s_i and replies query Q_j with exact value.

Our approach to designing efficient data collection in energy-harvesting sensor networks is based on 2 main observations of this data collection application and the harvesting systems.

First, we notice that there exists a tight coupling between the ability of the system to harvest energy and the consequent data quality, i.e., in systems with energy harvesting capabilities, sensor nodes only sense and communicate when there is sufficient harvested energy. Intuitively better harvesting leads to better data quality; poor harvesting conditions imply loss of accuracy. Fig. 3 [17] illustrates an example of how data quality (measured as error margin) needs can dictate energy cost. The figure shows the total energy cost which is the sum of 2 components, source-initiated cost and consumer-initiated cost. We later (in Section 3) will explain the concept of error margin, its relation to data quality and the cost components. The main observation is that in the useful range of data quality $[0, W^*]$, the smaller the error margin is, the higher data quality and the cost are. Such graph and observation can be generated for all range-based/approximated data collection applications [17].

Note that the total energy cost curve has a global minimum W^*. For traditional battery systems in which the battery charge keeps decreasing until replacement, this minimum total energy cost point is the optimal operating point because it consumes least energy, prolongs battery and system lifetime while maintaining reasonable data quality. This is the main idea behind energy efficient approaches for approximated data collection applications [14,32,33]. However, from the standpoint of an energy harvesting system, this is not the optimal operating point. Instead, the system should scale data quality (in this case, error margin) to match energy harvesting supply. Such scaling is possible since many applications can tolerate certain loss/reduce in data quality in favor of other features, such as longer system lifetime or system sustainability. In short, our goal is to take advantage of the data quality-energy tradeoff and to adapt data quality given energy harvesting constraints.

The second observation is that despite variation, renewable energy sources (e.g., solar energy) have repetitive patterns (day–night, season) and these patterns can be exploited to predict future energy harvesting capability. The goal of our work is not to design new energy harvesting prediction techniques; instead, we aim to use existing prediction algorithms for energy harvesting to optimally plan the energy budget data quality for the next harvesting period. In our proposed scheme, a harvesting period is divided into time slots of equal length. We propose techniques to allocate energy budgets for each slot that will maximize overall data quality while ensuring that nodes will not shutdown due to lack of energy. If high data quality is required for a slot with low energy harvesting capability, the energy supply will not meet the energy demand and the system might run out of battery. In this case, a node must shut down to replenish, suspending monitoring activities. Suspended nodes may cause critical events (often unpredictable) to be missed. On the other hand, if low data quality/low energy budget is assigned to a slot with predicted high energy harvesting, the harvested energy is not utilized fully and might be wasted due to energy overflow once energy accumulation exceeds buffer capacity. To solve this problem, we must leverage both data quality-energy tradeoff and energy harvesting prediction to optimally plan ahead.

Another challenge for slot-based budget allocation scheme arises due to fluctuations in the energy harvesting profile. Fine-grain variation in energy harvesting profile is unknown until real-time; they are not captured by prediction algorithms and cannot be dealt with during an offline phase of budget planning. This implies that we need an online phase to continuously monitor changes in energy harvesting profile and adapt the energy budget as well as data quality accordingly at runtime.

In summary, a high level definition of the problem is as follows: given predictions of energy harvesting and data quality requirements, the goal is to keep all nodes of the system operational while optimizing data quality. This can be done by optimally allocating energy budget in the offline phase and adapting data quality based on the changes in energy profile in the online phase.

We propose a framework called QuARES, quality-aware renewable energy-driven sensing framework to address this problem. QuARES is a cross-layer energy management framework consisting of 2 phases, an offline phase and an online phase. Fig. 4 depicts our framework and its phases.

Offline phase is executed during a harvesting period to plan energy budgets and data quality for the next harvesting period. In this phase, the base station runs a prediction algorithm to estimate energy harvesting availability in the next period. The predicted information and the knowledge of data accuracy-energy cost are inputs to an optimization algorithm using a linear programming formulation (step 1, Fig. 4). The optimization algorithm allocates energy budgets for each time slot in the next period to achieve optimal overall data quality. The results are then sent to sensor nodes (step 2) and each sensor node stores the energy budgets in its memory.

Online phase is executed during each harvesting period. At the beginning of each time slot, the energy budget calculated in the offline phase for this period is retrieved from memory and corresponding data quality is assigned (step 3). Sensor nodes and the base station exchange messages according to their protocol to

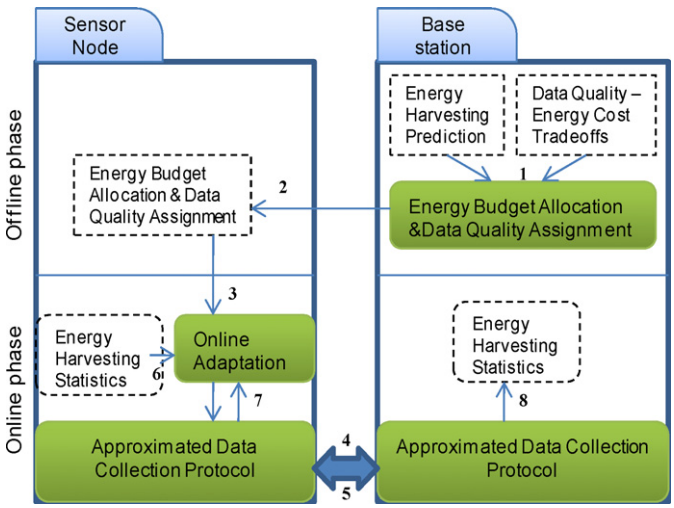

Fig. 4. QuARES framework.

maintain application's data quality (steps 4 and 5). Online adaptive heuristics monitor the energy buffer status and the actual harvested energy rate (step 6) to adjust the data quality accordingly (step 7). Adaptation allows the system to cope with variation in renewable energy sources and maintain system operation. In addition, energy harvesting statistics are sent to the base station (together with messages in step 4) to enhance the energy harvesting prediction for the next harvesting period (step 8).

3. Energy harvesting and data quality models

In this section, we define system parameters, energy harvesting and data quality models. This section is preliminary explanation for the algorithms used in each phase of our framework in the next section.

3.1. Energy harvesting prediction model

Renewable energy sources such as solar energy show predictable patterns that are exploited in several coarse-grain (or slot-based) harvested energy prediction algorithms [4,8,15,16,34]. These prediction algorithms use estimation techniques such as Exponential Weighted Moving Average [4] or Weather-Condition Moving Average [34]. Sharma et al. [30] leverage weather forecast to improve its prediction model. Harvested energy prediction then can be used for planning activities of nodes in a sensor network to keep them functional, yet still powered. In this paper, we assume an ideal prediction algorithm for slot-based energy harvesting estimation.

Fig. 5 shows our energy harvesting prediction model. We assume that each sensor has an energy buffer, with capacity C, to store harvested energy. Energy accumulated beyond capacity C will be discarded (energy overflow). Let E^0_{initial} be the available energy at the beginning of the harvesting period. Let $E_{\min}$ be the minimum energy to be reserved at the end of the harvesting period.

We denote the length of a harvesting period as T. A harvesting period is then divided into equal-length intervals or time slots. Let N be the number of slots in a harvesting period T. The values of T and N are defined by the energy harvesting prediction algorithms on the basis of renewable energy source, the nature of sensor input and query model. Our framework, however, is independent of these parameters. For each time slot i, the prediction algorithm provides E^i_{harvest} which is the amount of harvested energy in that slot. Using harvested energy prediction, our framework can plan energy budget for future slots E_{budget} in order to sustain system operation and maximize data quality based on quality-energy tradeoff presented in Section 3.2 below.

Some existing works exploit patterns in renewable energy profile to predict future harvested energy and to plan energy budget accordingly. Noh et al. [8] propose a minimum variance slot-based energy budget allocation for systems which prefer steady level of operation. Gorlatova et al. [28] propose several time-fair energy budget planning algorithms. These solutions do not consider data quality requirements and possible changes in such requirements. Hence it is not suitable for systems whose level of operation is dictated by application and user constraints that vary. Furthermore, since data quality is not the goal, it will not be optimized. Moser et al. [13] allocate energy budget for time slots to maximize quality of service for general systems. However, given the fluctuation of energy harvesting within each time slot, an offline budget allocation methodology cannot always guarantee system operation and certain data quality. Therefore, online phase with adaptation is important and necessary in an energy harvesting management framework.

Both the offline budget planning and online adaptation needs a formal way to express data quality needs and data quality-energy cost trade-off in order to allocate energy budget and leverage data quality. This model of data quality is essential to our work and will be present in the next section.

3.2. Data quality model

Our work focuses in data accuracy, the most important aspect of data quality. Data accuracy requirement can be expressed using

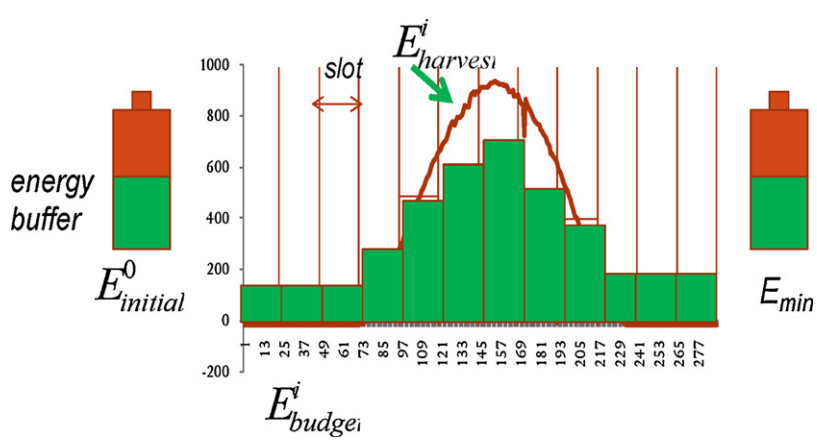

Fig. 5. Our energy harvesting prediction model.

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

error margin of an actual value v_i, e.g., $v_i \pm 10$ or $v_i \pm 10\%$. Such error tolerance in data quality requirement can be exploited while tuning the system. Error margins can be increased or decreased to meet both data accuracy constraints and system constraints, such as varying energy supply. Our energy harvesting management framework exploits this error tolerance to adapt the system to the availability of renewable energy sources.

We model data accuracy in terms of error margin. Error margin δ_i is the bounded difference between the sensing value v_i at the sensor node s_i and the approximated response r_j to a query Q_j, i.e., $\left|r_j - v_i\right| \Leftarrow \delta_i$. The approximation range $[l_i, u_i]$ in the base station's cache where l_i is the lower bound and u_i is the upper bound for sensing value v_i must satisfies $u_i - l_i = 2\delta_i$. This error margin is the constraint of the application and is not the measurement error or sensitivity of physical sensors. We provide the following definitions:

Definition 1. Consistent state refers to the state of cached entry on the base station B. If the approximation range $[l_i, u_i]$ satisfies $u_i - l_i = 2\delta_i$ and sensing value v_i satisfies $l_i \le v_i \le u_i$, the cache entry is in a consistent state.

Definition 2. Inconsistent state refers to the state of cached entry on the base station B in which v_i falls outside the approximation range $[l_i, u_i]$ or $u_i - l_i \neq 2\delta_i$.

When sensor node s_i reads a new value v_i, it checks if the cache is still consistent. If the cache is inconsistent, a new approximation range $[v_i - \delta_i, v_i + \delta_i]$ is sent to the base station to update the cache. This process is called source update. If the sensor node does not update the base station, e.g., running out of battery for sensing and communication, the cache entry is in inconsistent state. The number of source updates essentially reflects the physical characteristics of monitored phenomenon. We assume the sampling rate of sensors is sufficient to detect changes in the monitored environment and time to send update message is negligible.

When the base station B receives a new query Q_j, it first checks if the current cache entry satisfies query's data accuracy constraint A_j, i.e., $\delta_i \le A_j$. If it is satisfied, the base station immediately responses to the query with value $(u_i + l_i)/2$. Otherwise, the base station sends a request to s_i for current sensing value v_i. The sensor node replies with the updated approximation range $[v_i - \delta_i, v_i + \delta_i]$, the base station updates its cache entry and sends v_i to the application. This process is called consumer update. The number of consumer updates reflects the nature of query model both in term of query frequency and associated accuracy constraints.

The smaller the error margin is, the more source updates and less consumer updates are. On the other hand, the larger the error margin is, the less sources updates and more consumer updates are. The two extremes are $\delta = 0$ (only source updates) and $\delta = \infty$ (only consumer updates). The total energy cost consists of both energy cost for source updates and energy cost for consumer updates. Fig. 2 is one example. According to Olston et al. [17], the total energy cost for a given error margin can be expressed as

$$E(\delta) = \frac{\text{Cost}_{\text{source update}} \times K1}{\delta^2} + \text{Cost}_{\text{consumer update}} \times K2.$$

where $K1$ and $K2$ are parameter defined by application and query model. The value for δ such that the total energy cost is minimum is

$$\delta = (2 \times \text{Cost}_{\text{source update}}/\text{Cost}_{\text{consumer update}} \times K1/K2)^{1/3}$$

We call this error margin as $\delta_{\max}$. Beyond this point, both the error margin and energy consumption is higher, hence it is not worth considering error margin > $\delta_{\max}$. Therefore, our adaptation choose only error margin in the range $[0, \delta_{\max}]$.

Problem Formulation
Given input:
Harvesting period T, N time slots
Battery capacity $C \le 0$ and Initial battery $E^0_{initial} \ge 0$
Minimum battery remained after T: $E_{\min} \ge 0$
Energy harvesting prediction of each time slot $E^i_{harvest}$ $\quad 0 \le i < N$
Error margin and energy cost vectors: $(\delta_1, \ldots, \delta_K)$ and $(E^1_{\cos t}, \ldots, E^K_{\cos t})$
Objective: Minimize $\overline{\delta}$, the average actual error margin, in a harvesting period T

Fig. 6. Problem formulation.

Definition 3. Baseline average error margin denoted as $\overline{\delta}_{i,\text{base}}$ is the average error margin predicted in the offline phase, representing the predicted average data accuracy of data collected from sensor node s_i.

Definition 4. Actual average error margin denoted as $\overline{\delta}_i$ is the average error margin maintained by sensor s_i in a harvesting period T, representing the actual average data accuracy of data collected from sensor node s_i.

Our framework, QuARES, optimizes $\overline{\delta}_{i,\text{base}}$ in the offline phase and uses online adaptation to maintain $\overline{\delta}_i$ close to $\overline{\delta}_{i,\text{base}}$ during online phase.

A more formal characterization/formulation of the problem is given in Fig. 6 where our goal is to minimize the actual average error margin. In the next section, we describe our algorithms in each phase to achieve this goal.

4. The QuARES framework: algorithms

In this section, we will present our linear programming solution to the offline budget planning problem, considering data quality-energy tradeoff to optimize data quality while regulating energy usage to match energy harvesting supply. We then propose several online policies to cope with fluctuations in energy profile at runtime.

We assume to have a vector of error margin levels $(\delta_1, \ldots, \delta_K)$ and a corresponding vector energy consumption $(E^1_{\cos t}, \ldots, E^K_{\cos t})$ where $E^j_{\cos t}$ is the energy required to maintain error margin δ_j in a time slot. K is the number of discreet error margin levels.

4.1. Offline phase: energy budget allocation and data quality assignment

In the offline phase, we solve a linear optimization problem (see Fig. 7) to allocate energy budget and assign a corresponding baseline error margin for each time slot. The linear optimization problem for each sensor node is solved by a linear solver on the base station. System parameters (T, N, battery capacity C, initial battery E^0_{initial}, $E_{\min}$), harvested energy prediction (E^i_{harvest}), and error margin-energy cost $(\delta_j - E^j_{\text{cost}})$ in Fig. 6 are the inputs to this optimization problem.

Let E^i_{initial} denote the energy in the battery at the beginning of time slot i. Since the battery cannot store more than its capacity C, E^i_{initial} must be constrained by this upper bound (constraint 1, Fig. 7). Any excess energy is discarded (energy overflow). Let E^i_{budget} denote the energy budget for slot i which could only be drawn from the available energy in the battery at the beginning of the slot and the energy harvested during this slot (constraint 2). Vice versa, the energy in the battery at the beginning of a slot is limited by the amount of energy available in the previous slot subtracted by its energy consumption (constraint 3).

Offline phase at Base Station B

1. Run a linear solver to solve this linear programming for each sensor node s

Objective Minimize $\overline{\delta}_{base} = \dfrac{\sum_{i=0}^{i=N} \delta^i_{base}}{N}$

Subject to constraints:

Input and Constraints in Figure 3

$$E^i_{initial} <= C \quad (1)$$

$$E^i_{budget} \le E^i_{initial} + E^i_{harvest} \quad (2)$$

$$E^{i+1}_{initial} <= E^i_{initial} + E^i_{harvest} - E^i_{budget} \quad (3)$$

$$0 <= q[i,j] <= 1 \quad \forall 0 <= i < N, 0 <= j < K$$

$$\forall i < N : \sum_{j=0}^{K} q[i,j] = 1 \quad (4)$$

$$\forall i < N : \delta^i_{base} = \sum_{j=0}^{K} q[i, j]^* \delta_j \quad (5)$$

$$\sum_{j=0}^{K} q[i,j]^* E^j_{cost} \le E^i_{budget} \quad (6)$$

$$\forall i < N : \delta^i_{base} <= \delta^i_{max} \quad (7)$$

2. Send energy budget allocation and baseline data quality assignment to sensor *s*

Online Phase at Sensor node s

1. Receive energy budget allocation and baseline data quality from base station
2. Save in memory for run-time use in the next harvesting period

Fig. 7. Offline phase: energy budget allocation.

Policy 1: Inter-frame online adaptation (slot *i*)

1. *buffer* = current energy in the buffer
2. *offset* = $E^i_{initial}$ - *buffer*
2. **if**(*offset* > *epsilon*) **then** #adjust budget of future slots
3. $E^i_{budget} = E^i_{budget}$ - *offset*
4. *j* = find_quality_level(E^i_{budget})
5. $\delta = \delta_j$
6. update_server(s_i, $v - \delta$, $v + \delta$)

Policy 2: Intra-frame online adaptation (slot i)

1. *h* = current_harvesting_rate;
2. if (|*h* - h_{old}| < *epsilon*) then return
3. else h_{old} = *h*
4. *buffer* = current energy in the buffer
5. *reserve* = buffer energy reserved for future sub-slots
6. *l* = length of a sub_slot
7. *supply* = *h***l* + (*buffer-reserve*) #energy supply for this sub-slot
8. **if** (*supply* < E^i_{budget}/*number_of_subslots*)
9. **then** δ = find_quality_level (*supply***number of subslots*)
10. **else** $\delta = \delta^i_{base}$
11. **if** δ changes **then**
12. update server(s_i, $v - \delta$, $v + \delta$)

Fig. 8. Online phase: dynamic adaptation policies.

The baseline error margin for each slot is assigned based on the allocated energy budget. Let $q[i, j] = 1$ if error margin δ_j is assigned to slot *i*, $q[i, j] = 0$ otherwise. Each slot *i* could only be assigned one base line error margin (constraint 4) and this error margin is computed in constraint 5. In addition, its energy budget must be sufficient to maintain this error margin, i.e., at least E^j_{cost} (constraint 6). Let δ^i_{max} be the maximum tolerated error margin of the application in slot *i* and also the upper bound for the assigned base line error margin of slot *i* (constraint 7). Many applications could have time-based quality constraints such as monitoring closely during day-time than night-time or vice versa. Under this constraint, the QuARES framework makes sure the data accuracy is never degraded beyond application needs.

The objective of this optimization problem is to maximize the baseline average error margin, $\overline{\delta}_{base}$. Next, we describe our online phase 2 of our framework, consisting of dynamic adaptation policies whose goal is to maintain the baseline error margin and guarantee continuous system operation.

4.2. Online phase: dynamic adaptation policies

During online phase, data collection protocol runs on both sensor and base station to keep the cache in consistent state and respond to monitoring queries. However, there could be fluctuations in energy harvesting profile and actual harvesting rate could be lower than the predicted average rate. The system thus needs online adaptation policies to tackle energy supply fluctuations, maintain system operation and data accuracy constraints.

Our online adaption running on sensor node is a heuristic which keeps track of current harvesting rate and battery status to adjust the error margin, guarantee system operation and consistency of cached entry on base station. We develop 2 dynamic adaptation policies: inter-frame adaptation and intra-frame adaption (see Fig. 8).

(1) Inter-frame adaption is triggered at the beginning of each time slot. The harvested energy often does not come at a constant rate; in particular, when harvested energy is abundant and the energy buffer is almost full, energy overflow happens. The harvested energy thus could be less than what is predicted and the system needs to adapt its energy budget plan and base line error margin. The inter-frame policy keeps track of this energy discrepancy and distributes the energy offset among current and future slots' energy budgets. Our inter-frame adaptation algorithm is summarized in Policy 1, Fig. 8.

(2) Intra-frame adaptation on the other hand is triggered more often every sub-slot within a time slot to quickly adapt to fluctuations of the renewable energy source. The length of a sub-slot depends on the energy source, which is the interval of time that the harvested energy rate remains fairly stable, e.g., 1–5 min. Every sub-slot, intra-frame policy will check if the current harvesting rate is significantly less than the predicted average harvesting rate and adjust error margin in the current sub-slot accordingly. In the next sub-slots, if the harvesting rate increases above the expected average rate, the policy restores the baseline error margin. Our inter-frame adaptation algorithm is summarized in Policy 2, Fig. 8.

4.3. Data collection protocol

Base station and server both run a data collection protocol to communicate with each other (see Fig. 9). When receiving updates from sensor nodes, base station will refresh its cached entry with the new approximation range (line 2). If this is the response to a previous query, a response message is sent back to the monitoring application (lines 3 and 4).

When base station receives new query (line 5), it first checks if the current approximation range satisfies the query constraints. If the current approximation is sufficient, base station responses immediately (lines 6 and 7). Otherwise, it sends a request message to sensor node and wait for reply from sensor node (line 8).

Protocol for Base Station B

1. **if** receive update_server(s_i, u_i, l_i) **then**
2. update database with new range [l_i, u_i]
3. **if** this is a consumer_initiated_update and query Q_j waiting
4. **then** responseQuery((l_i+u_i)/2)
5. **if** receive query Q_j(s_j, C_j) **then** #external queries
6. **if** $C_j > (u_j - l_j)/2$ #Q_j requires lower quality of data
7. **then** responseQuery((l_j+u_j)/2)
8. else querySensor(s_j)

Protocol for Sensor node (*s*, time *t*)

9. **if** (current_time *t* at beginning of a new slot *i*) **then**
10. $\delta = \delta^i_{base}$; update E^i_{budget}
11. inter_frame_adaptation()
12. **if** (current_time *t* at beginning of a new sub-slot) **then**
13. intra_frame_adaptation()
14. v = readSensorValue(t);
15. **if** ($v \in [l_i, u_i]$ **or** receive querySensor(s_i)) **then**
16. $l_i = v - \delta$
17. $u_i = v + \delta$
18. update_server(s_i, u_i, l_i)

Fig. 9. Data collection protocol.

On sensor node side, at the beginning of a new harvesting slot, the allocated budget and base line error margin saved in memory is used to configure the data collection protocol (line 10). Lines 11–13 are calls to online adaptation policies described in the previous section.

The sensor node keeps reading sensing value and checks it against the cached range on the base station (line 14). If the new value falls out of this range or if the sensor node receives an update request (line 15), a new approximation range is computed (lines 16 and 17) and is sent to the base station (line 18).

5. Indoor energy harvesting: a measurement study

In this section, we explore the opportunity of harvesting energy indoor. We show that these energy sources can sustain low power system such as wireless sensor networks in a sensorized infrastructure.

There are various ways of harvesting energy indoor from different energy sources. The most accessible are light in offices and hallways which can be readily harvested by solar panels. Hande et al. [23] devise a system to harvest light from fluorescent light in hospital hallways to support routing of patient data using Micaz motes. Tan and Panda [25] designed a hybrid energy harvesting system consisting of both indoor ambient light and thermal energy harvesting circuits. Refs. [26,27] present a prototype and analysis of hybrid solar and RFID harvesting sensor mote. Gorlatova et al. [28] carries out indoor energy harvesting measurement over 16 months at different settings. This work also describes several design space dimensions and proposes several algorithms to achieve time-fair resource allocation at several working points of the design space. Other energy sources such as kinetic, vibration, magnetic can also be harvested. inDOOR Energy Harvester is a project at NYU by Zollinger [24] that builds an add-on for hinged doors in order to convert kinetic energy from opening and closing a door to electricity for other grid uses.

We carry out measurement of indoor energy harvesting in several settings in such infrastructure. In Section 6, experiments with measurement data from dual energy harvesting sources and various types of sensors are executed in QuARES system simulation. The result shows that QuARES is able to exploit indoor energy harvesting availability to manage system efficiently while helping designer to explore the design space and choose optimal solar size.

We are deploying a test bed of energy harvesting wireless sensor network in Responsphere [11]. Fig. 10 shows our Responsphere infrastructure and prototype of our energy harvesting platform, including a Crossbow sensor (temperature, light and acoustic sound), two solar panels and 2 AA batteries.

The first step to explore opportunity of indoor energy harvesting for wireless sensor network is to measure energy harvesting availability in different setting and collect data about sensor and sensor mote energy consumption. We want to answer three essential questions:

Question 1: How much energy available to harvest in an indoor scenario?

Question 2: Is it possible to use this energy to support indoor sensor network? And to what extent?

Question 3: If it is possible, what is optimal data quality, battery size and optimal solar panel size?

With a focus on light sources, we identify two promising sources of indoor energy harvesting: light bulbs inside offices, along hallways and solar light through windows around the building. They are often available for extended period of time in all buildings. Table 1 is the summary of our indoor settings for measurement.

We carried out an extensive light measurement study at University of California, Irvine for different types of light bulb. We measure the harvesting power output of a solar panel in perpendicular to the light source at the distance of 10 cm (solar-made, 9.6 cm × 6.4 cm). Table 2 shows the results of our measurement. We experiment with different possible types of bulb used for lighting in offices and at home. We observe that the higher the energy a light bulb consumes, the higher the harvesting power it provides as the irradiance increases. However harvesting power is not proportionally to light output which is measured in Lux which indicates how human eyes perceive the light but not the energy the light carries. For example, soft white incandescent bulb has slightly lower Lux than compact fluorescent bulbs but significant higher harvesting power output. Among these types of bulbs, soft white compact fluorescent light bulb has a nice balance between its energy consumption and energy harvesting power it can provide.

Solar panel set up on window panels is another source of energy for indoor applications. Table 3 shows our comparison of 2 indoor window set-ups and 2 outdoor locations measured at noon. Result shows that filtered light carries less energy by one order of magnitude, from 4 to 10 times less compared to outdoor sunny locations but can be comparable to outdoor shadowy locations.

Longer-term measurement of these window energy harvesting set-ups reveals not only temporal but also spatial characteristics of light harvesting. Fig. 11 shows the power output of the solar

Table 1
Summary of indoor measurement settings.

Location tag	Location description	Quantity	Measurement duration
L-1	Different indoor light bulbs in lab without window	6	30 min to 1 h each type of bulb
L-2	On window panel, facing South-West direction	1	3 days
L-3	On window panel, facing South-East direction	1	3 days

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

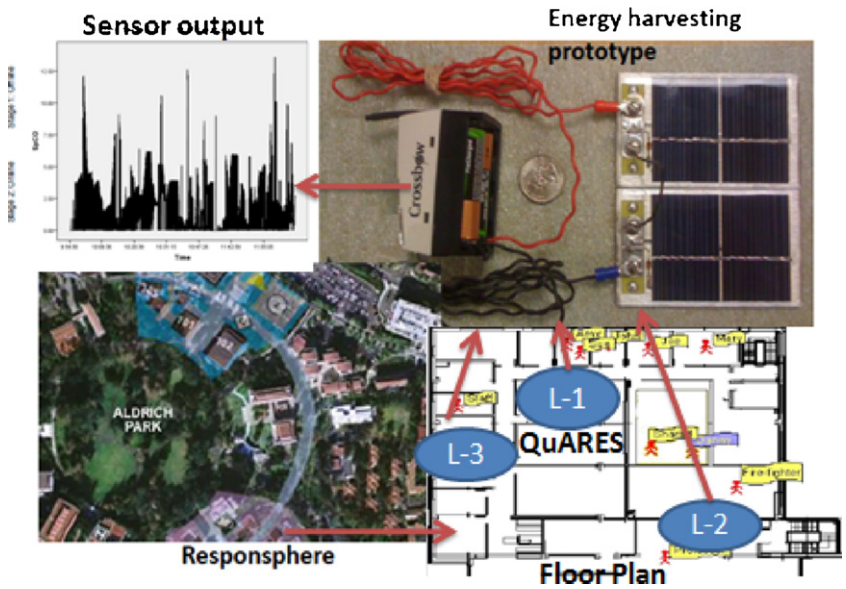

Fig. 10. System prototype.

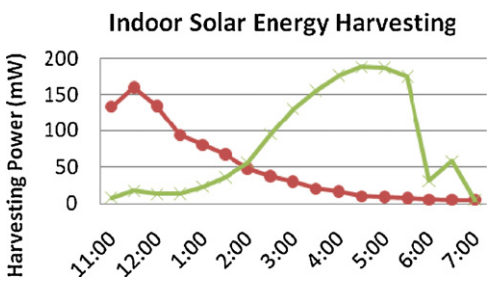

Fig. 11. Indoor energy harvesting – window setup.

panels installed on window panels at locations L-2 and L-3 on a same day. The average energy harvesting power in this period at L-2 is 18 mW while at L-3 is 28 mW. Both energy harvesting profiles have one peak point but each peak point occurs at different time. One peak point (L-2) is at 11 am while another (L-3) is at 5 pm, shifting by an amount of time depends on the spatial location of the solar panels with respect to building and the sun's direction. The energy harvesting power reaches its peak at a given location when the solar panel is directly proportional to the angle of the sun light and under no shadow condition. Being aware of these spatial and temporal variations is very important in building models, planning deployment and schedule activities for nodes in an energy harvesting wireless sensor network.

Table 4
Sensor power consumption.

Temperature	9.35 mW
Acceleration	10 mW
Humidity	3–15 mW
Heart rate/ECG	19.8 mW
Low-power image	80 mW
Acoustic	540 mW
Magnetic	1.5 W
Motion detection	4 W
Low-power GPS	5 W

Our study has answered question 1. To answer question 2, we need to know how much energy sensors require to sustain. There are various types of sensors being used in the building; we classify them into three groups:

- low-power sensors (temperature, acceleration, humidity, heart rate/ECG sensors);
- medium-power sensors (low-power image, acoustic and magnetic sensor);
- high-power sensors (motion detection, low-power camera, GPS, vision or fluid indicator).

Table 4 summarizes our study of different types of sensors in the market and its corresponding typical power consumption. This

Table 2
Indoor energy harvesting – light bulbs.

Type of bulb	Energy used (W)	Light output (Lux)	Harvesting power (mW)
Daylight compact fluorescent	14	800	28.32
Bright white compact fluorescent	14	800	29.75
Round back light	Max. 55		31.09
Soft white compact fluorescent	14	900	35.15
Halogen light	55	825	66.48
Soft white incandescent bulb	57	780	148.97

Table 3
Indoor energy harvesting – window panels.

Location	L2	L3	Outdoor (under shadow)	Outdoor (sunny location)
Voltage (V)	16.24	7.17	15.2–15.7	17.2–17.7
Current (mA)	20.7	2.6	9–11	59.4–65.2
Availability	10–12 h	10–12 h	10–12 h	10–12 h

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

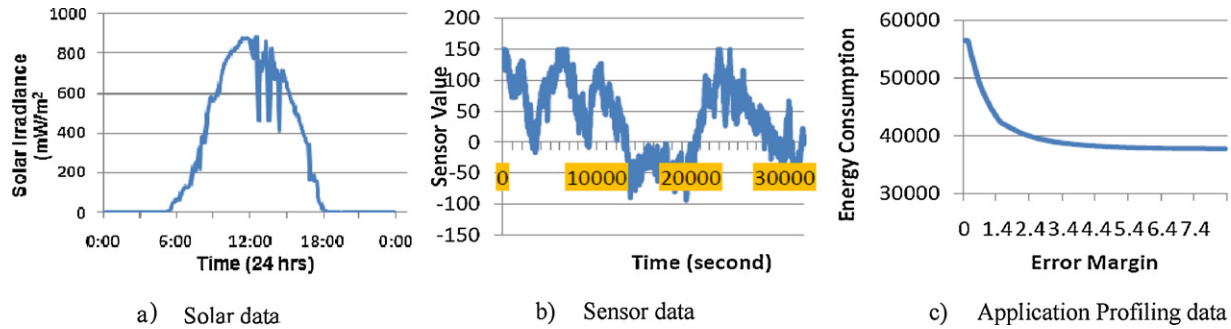

Fig. 12. Inputs to our simulator.

study gives us an understanding the range of power demand each class of sensors requires and roughly estimates whether our solar harvesting are good potential energy sources for these sensor systems. At the first glance, by matching power consumption of sensor and power generated by our indoor solar panel, it could support low power sensors.

However, this is power consumption of sensor alone and has not taken into consideration power consumption of sensor node/board whose processing unit, memory and radio component also draw significant portion of overall power consumption. In addition, energy management plays an important role in planning, adapting and thus sustaining the system. In Section 6.3, we will apply the real data we present in this section to evaluate the feasibility of indoor energy harvesting for sensor network and evaluate QuARES performance.

In short, the study in this section explores indoor energy harvesting and the initial measurement results shows potentials of indoor energy harvesting sensor network. We will use QuARES to evaluate this opportunity in the next section to fully answer questions 2 and 3.

6. Evaluation

In this section, we first explain the experimental setup to evaluate the effectiveness of our proposed framework QuARES and then we compare the results with other existing policies in terms of error margin (data accuracy), responsiveness to queries, and energy consumption. Based on measurement study from Section 5, we will also evaluate QuARES by applying this framework to indoor sensing scenarios (Section 6.3).

6.1. Experimental setup

We initially implemented the approximated data collection application and QuARES framework in QualNet network simulator [18]. The simulator is configured to simulate a sensor network of Mica motes with ZigBee standard specification. Power consumption of sensor node is set accordingly to Mica-2 [19].

Sensor data are generated randomly from the range [−150, 150]. The sampling rate is 100 Hz, each sampling either increases or decreases the previous value by an amount randomly chosen in the range [0.5, 1.5]. Fig. 12b gives an example of randomly generated sensor data for simulated time of 6 h. Periodic queries arrive every 100 ms. Sporadic queries are modeled by Poisson distribution with mean interval = 100 ms. Each query is associated with an error tolerance of mean = 20 and deviation = 1.

Energy harvesting profile is retrieved from National Renewable Energy Lab website [20]. Solar profiling for a day is shown in Fig. 12a. The data is average solar irradiance (mW/m^2) at a specific location every 5 min. The irradiance is converted to harvested energy by linear conversion considering solar panel size 9.6 cm × 6.4 cm, solar cell efficiency 10% and harvesting efficiency 80%. We modify QualNet battery model to charge battery every 1 min. We assume a perfect solar energy prediction algorithm which gives accurate slot-based prediction to our offline phase. We choose $T = 1$ day and $N = 48$ slots.

We profile error margin vs. energy cost by simulating application for different error margins in the range $[0, \delta_{\max}]$ (see Fig. 12c). Since it is impossible to profile every value in this range, we choose to profile discreet values every interval $r = 0.1$. For each error margin δ_{profile}, we fix the baseline error margin for all slots ($\delta^i_{\text{base}} = \delta_{\text{profile}}$) and set energy buffer always full. We run the simulation for a simulated time of one day and divide the total energy consumption by the number of time slots N to obtain average energy cost per slot at error margin δ_{profile}. For the given query model and sensor input, we find that $\delta_{\max} \approx 8.0$.

6.2. Experimental results

To evaluate the effectiveness of our proposed QuARES framework, we have implemented several offline or online policies for energy management during data collection. We evaluate the policies in terms of their data accuracy (error margin), system sustainability, and energy consumption. The implemented policies are as follows:

- FIX_ERROR ($\delta = 8.0$) and FIX_ERROR ($\delta = 0.5$): FIX_ERROR is an offline policy and has no online adaptation. A fixed baseline error margin is assigned to all time slots.
- GREEDY_ADAPT: greedy adaptation is a completely online protocol without offline energy budget assignment. It sets an error margin at the beginning of each time slot according to the amount of available energy in the buffer.
- MIN_VAR: adopted from [8], it allocates energy budget for slots in the offline phase with minimum variance. Its goal is to maintain steady operation for the system. It does not have online adaptation.
- QuARES: as presented in this paper, our quality-aware framework minimizes error margin both in offline and online phases.

Table 5 shows results of QuARES in comparison with other approaches. FIX_ERROR ($\delta = 8.0$) has a very high error margin and consumes minimum energy. Due to high error margin, source updates is very low compared to consumer updates; this scheme is thus very close to push-based data collection. The system responds to all queries at the trade-off of higher error margin. This is the extreme case where energy saving is a dominant requirement compared to data accuracy and is suitable for traditional discharging battery systems [14]. The second column FIX_ERROR ($\delta = 0.5$) is another fixed error rate policy which attempts to maintain a lower error margin $\delta = 0.5$ by exploiting energy harvesting. Due to very low error margin, source updates is very high compared to consumer updates; this scheme thus is very close to pull-based data collection. However, it fails since the sensor node cannot maintain

Table 5
Comparison of 5 different approaches.

	FIX_ERROR ($\delta = 8.0$)	FIX_ERROR ($\delta = 0.5$)	GREEDY_ ADAPT	MIN_VAR	QuARES offline	QuARES offline + online
Average error margin	8.00	0.50	0.348	0.388	0.156	0.159
Total energy consumption (J)	1813	2686	2656	2641	2641	2641
Shut down time for harvesting (min)	0	45	21	7	4	0
Failed responses to queries	0	570	420	196	64	0

this low error margin when the harvested energy is low. The battery is exhausted and system needs to shut down for 45 min to replenish energy. This leads to a very high number of failed responses to queries. FIX_ERROR approach in general cannot work in dynamic energy harvesting systems.

The third and the fourth column show results of GREEDY_ADAPT and MIN_VAR [8]. Both have comparable average error margin and energy consumption. However, both do not consider fluctuation of energy harvesting in its greedy online adaptation or offline budget allocation. Therefore, in both cases, sensor node runs out of battery and shuts down for harvesting, leading to failed responses to queries. It is thus necessary to have online adaptation to handle fluctuation of renewable energy. In columns 5 and 6 of Table 5, we study the significance of each phase in the QuARES framework: offline budget allocation and online adaptation. We compare our QuARES offline and the whole QuARES with both online and offline phases. As seen from Table 5, without online adaptation, QuARES (offline) must shut down the sensor node for 4 min and thus failed to response to 64 queries. The system often shuts down during the sunrise when both battery reserves and harvested energy rate is low.

Among all approaches, QuARES with both online and offline phases keeps the system alive for the whole harvesting period, responds to all queries, maintains low error margin from 30 to 70%, and enables a uniform energy usage across nodes. The results show that while our proposed offline phase in QuARES maximize data accuracy, the online adaption phase is required for successful query responsiveness.

6.2.1. Varying application constraints

We evaluated our framework for different application constraint profiles.

- Application constraint profile 1 (AC1) maintains low error margin during night time ($\delta < 1.5$) from 5 pm to 7 am.
- Application constraint profile 2 (AC2) maintains low error margin during night time ($\delta < 1.5$) from 7 pm to 5 am.

QuARES (Table 6) satisfies all application constraints at the trade-off of suboptimal energy budget allocation compared to no constraint case, higher error margin and less energy utilization. Interestingly, the QuARES offline phase also gives immediate feedback to system designers if the given data accuracy constraints are infeasible in the next harvesting period and needed to be adjusted. MIN_VAR, on the other hand distributes energy budget among slot with minimum variance regardless of application constraints, thus do not give any feedback to designers when application constraints are infeasible.

Furthermore, the budget is not in accordance with energy demand in different slots, MIN_VAR spends more energy in slots where it should save energy for higher-demand slots in the future and soon runs out of battery and fails to response to queries.

Table 6
Varying application constraints.

	App. constr. 1		App. constr. 2	
	MIN_VAR	QuARES	MIN_VAR	QuARES
Average error margin	3.09	0.891	1.067	0.783
Energy cons. (J)	2481	2509	2481	2501
Shut down time (min)	32	0	9	0
Failed responses to queries	521	0	227	0

6.2.2. Impact of battery capacity

We simulate the application with different battery capacities. Fig. 13a shows average error margin during day time and night time under different battery capacities on a summer day. As seen from the figure, battery capacity has negligible effect on error margin during the day as the energy supply is abundant. On the other hand, battery capacity has significant impact on error margin during the night. There is no energy harvested during this period and the capacity of battery limits energy saving to maintain data accuracy at night. In addition, we compare results of QuARES on summer days with results on winter days (Fig. 13b). Results show that to obtain a same error margin, the battery capacity required on winter days is larger than the battery capacity on summer days as winter night time is longer than summer night time. For example, to achieve average error margin of 1.0 at night, system would require battery capacity $C = 950{,}000$ (mJ) on a summer day but would require higher $C = 1{,}400{,}000$ (mJ) on a winter day.

6.3. Applying QuARES for indoor sensing

In this section, we evaluate QuARES in a realistic setting. In this setting, we assume Micaz motes, each with a sensor (we experiment with all different sensor types presented in Section 5). Sensor motes are powered by indoor energy harvesting. We first show an estimation of duty cycle such system can sustain given indoor energy harvesting. We then show how QuARES can improve the result, give the answers to important questions we proposed in Section 5 and also present many parameter options for system designers.

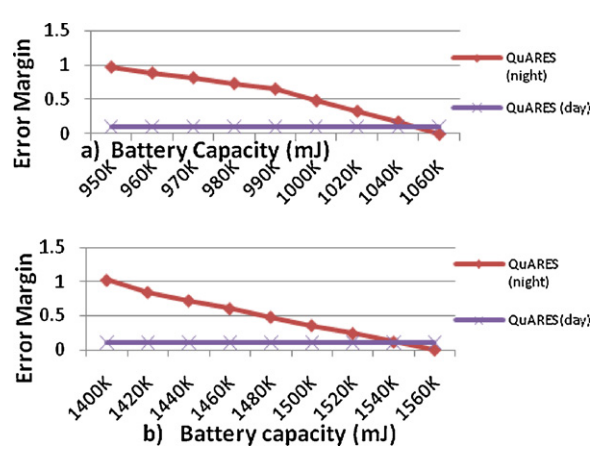

Fig. 13. Impact of battery capacity on application data accuracy (a) on summer days and (b) on winter days.

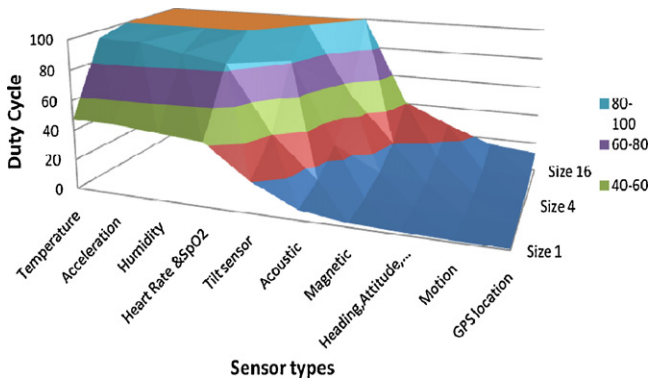

Fig. 14. System operation time – an estimation.

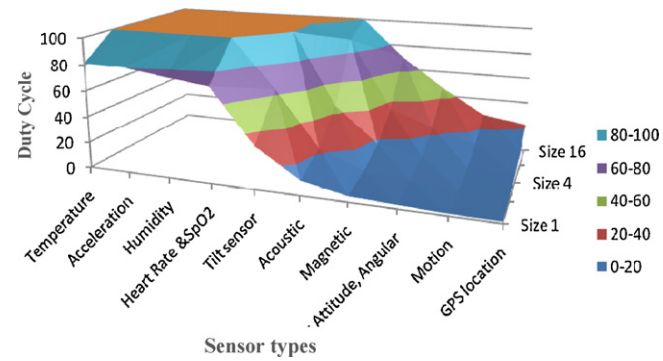

Fig. 15. System operation time by QuARES.

We do an estimation of system operation time or duty cycle using this raw computation:

$$\text{System operation time} = \frac{\text{maximum energy consumption}}{\text{average energy harvesting}}$$

and apply this for systems with Micaz sensor board connected to different type of sensors. Micaz board itself has maximum energy consumption of 95 mW, typical power consumption of different types of sensors was shown in Table 4. Maximum energy consumption of the system is the sum of embedded sensor's power consumption and sensor board's maximum power consumption.

In the base result and simulation result that we show later, we assume hybrid energy harvesting sources from one window solar panel and one soft white compact fluorescent bulb. We choose a simple model of the bulb that it is available for 10 h per day from 8 am to 6 pm. The average energy harvesting of this heterogeneous renewable energy source is 42.64 mW. Fig. 14 shows the estimation of duty cycle during which the system is active for each type of sensor and board.

This estimation is the base for our comparison later. From the figure we observe that low-power sensors such as temperature, humidity, heart rate needs at least 2 given solar panels to sustain 100% operation while medium power sensor such as acoustic, magnetic sensor might need more than 16 of such solar panels to sustain 80–100% operation. However, a solar panel of size 16 times of our solar panel could be too intrusive in an indoor environment such as offices or school and it can complete cover the light sources like lamps and small windows. For higher power sensor such as motion or GPS location, indoor energy harvesting is currently unable to support. From this base estimation, we might think that it is only feasible to use indoor energy harvesting for low power sensor network.

In fact, we run QuARES with measurement input of heterogeneous energy harvesting sources consisting of one window panel and one light bulb set-ups. We use a simplified model for light bulb energy harvesting in which we assume that the light is available at constant rate (as measured) for 10 h per day from 8 am to 6 pm. The result of measuring system operation time for systems with different embedded sensors is shown in Fig. 15. Overall there is significant improvement in system operation time for all sensor types. With only 1 solar panel, the energy harvesting system is able to sustain 70–80% operation time for low power sensors, improving over the base estimation by 30%. Acoustic sensor needs 9 solar panels to support 100% operation time which is about half of the size predicted by the base estimation. QuARES is able to do such improvement by exploiting tradeoff between data quality and energy consumption of sensor nodes. The summary of data quality vs. battery capacity for different types of sensor (low-power and medium-power sensors) is given in Fig. 16. Using simulation result from QuARES, the designer can explore the design space and decide:

- What is the solar panel size given the duty cycle requirement?
- What is the battery size required for a given data quality constraint?

The first question here is answered by tracing the graph of system operation time vs. solar panel size (e.g., Fig. 14) and the second

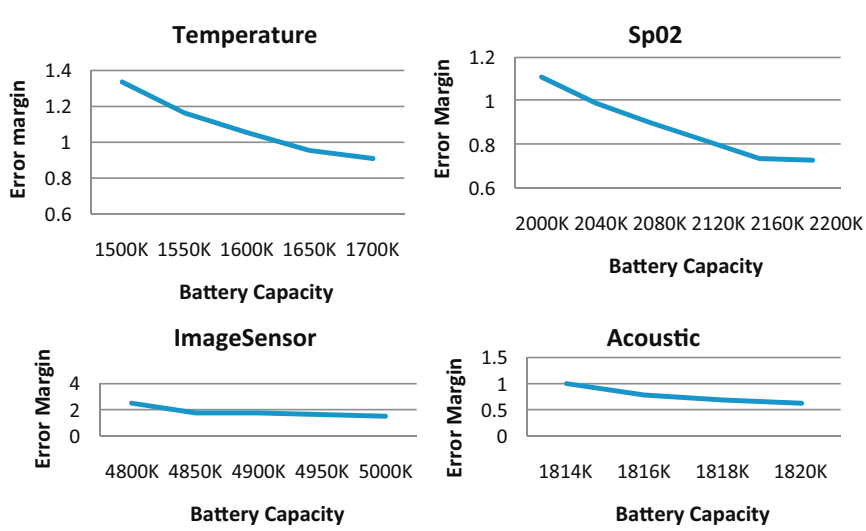

Fig. 16. Data quality for different types of sensor.

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

question can be answered by looking at the graph of data quality vs. battery capacity (e.g., Fig. 16).

To summarize this case study of indoor energy harvesting for wireless sensor network, we have shown an extensive study of energy harvesting power for different types of indoor light bulb as well as window solar panels. Our simulation result using synthetic data shows that QuARES can improve system operation (duty cycle) by exploiting trade-off between data quality and energy consumption. QuARES also helps designer to explore the design space to answer questions such as solar panels size and battery capacity needed according to requirement of application's data quality. To the best of our knowledge, our work is the first to explore systems with different types of sensor and different types of indoor lighting bulb for harvesting.

7. Conclusion

In conclusion, this paper proposes a complete autonomous energy management framework in energy harvesting WSNs whose goal is to optimize data accuracy for approximated data collection applications and sustain system operation. The offline phase explores energy harvesting prediction information to allocate energy budget among time slots in a harvesting period and maximize overall data accuracy. The online adaptation phase maintains the predicted data accuracy while coping with harvested energy fluctuation. Our framework is evaluated extensively in comparison with other approaches and considering different weather conditions, battery capacities and application constraints. Finally, we carry out a measurement study of indoor energy harvesting system and evaluate QuARES using measurement data. QuARES shows that it can improve system operation time for indoor energy harvesting system while helping designer to explore design space such as choosing optimal solar panel size and battery capacity for their systems.

References

[1] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan, M. Srivastava, Heliomote: enabling long-lived sensor networks through solar energy harvesting, in: SenSys, 2005.
[2] X. Jiang, J. Polastre, D. Culler, Perpetual environmentally powered sensor networks, in: IPSN, 2005.
[3] F. Simjee, P.H. Chou, Everlast: long life, supercapacitor operated wireless sensor node, in: ISLPED, 2006.
[4] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, V. Raghunathan, Adaptive duty cycling for energy harvesting systems, in: ISLPED, 2006.
[5] S. Liu, Q. Wu, Q. Qiu, An adaptive scheduling and voltage/frequency selection algorithm for real-time energy harvesting systems, in: DAC, 2009.
[6] C. Moser, D. Brunelli, L. Thiele, L. Benini, Lazy scheduling for energy harvesting sensor nodes, in: DIPES, 2006.
[7] T. Voigt, A. Dunkels, J. Alonso, H. Ritter, J. Schiller, Solar-aware clustering in wireless sensor networks, in: ISCC, 2004.
[8] D.K. Noh, L. Wang, Y. Yang, H.K. Le, T. Abdelzaher, Minimum variance energy allocation for a solar-powered sensor system, in: DCOSS, 2009.
[9] L. Wang, D.K. Noh, Y. Yang, H.K. Le, T.F. Abdelzaher, M. Ward, AdaptSens: an adaptive data collection and storage service for solar-powered sensor networks, in: RTSS, 2009.
[10] D. Hasenfratz, A. Meier, C. Moser, J.J. Chen, L. Thiele, Analysis, comparison, and optimization of routing protocols for energy harvesting wireless sensor networks, in: SUTC, 2010.
[11] Responsphere infrastructure test bed [Online]. http://www.responsphere.org/index.php
[12] A. Ravinagarajan, D. Dondi, T.S. Rosing, DVFS based task scheduling in a harvesting WSN for structural health monitoring, in: DATE, 2010.
[13] C. Moser, J.J. Chen, L. Thiele, Power management in energy harvesting embedded systems with discrete service levels, in: ISLPED, 2009.
[14] Q. Han, S. Mehrotra, N. Venkatasubramanian, Energy efficient data collection in distributed sensor environments, in: ICDCS, 2003.
[15] C. Bergonzini, B. Lee, J.R. Piorno, T.S. Rosing, Management of solar harvested energy in actuation-based and event-triggered systems, in: Energy Harvesting Workshop, 2009.
[16] M. Ali, B. Al-Hashimi, J. Recas, D. Atienza, Evaluation and design exploration of solar harvested-energy prediction algorithm, in: DATE, 2010.
[17] C. Olston, B.T. Loo, J. Widom, Adaptive precision setting for cached approximate values, in: ACM SIGMOD, 2001.
[18] QualNet network simulator [Online]. http://www.scalable-networks.com/products/QualNet/
[19] Crossbow Technology INC. Mica-2 Data Sheet [Online] Crossbow. http://www.xbow.com
[20] National Renewable Energy Lab [Online]. http://www.nrel.gov/midc
[21] B. Zhang, R. Simon, H. Aydin, Maximal utility rate allocation for energy harvesting wireless sensor networks, in: Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM'11), Miami Beach, FL, October, 2011.
[22] K. Fan, Z. Zheng, P. Sinha, Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks, in: SenSys, 2008.
[23] A. Hande, T. Polk, W. Walker, D. Bhatia, Indoor solar energy harvesting for sensor network router nodes, Journal of Microprocessors and Microsystems – Special Issue on Sensor Systems (December) (2006).
[24] R. Zollinger, inDOOR Energy Harvester [Online]. http://itp.nyu.edu/sigs/sustainable/indoor-energy-harvesting
[25] Y.K. Tan, S.K. Panda, Energy harvesting from hybrid indoor ambient light and thermal energy sources for enhanced performance of wireless sensor nodes, IEEE Transaction on Industrial Electronics 58 (September (9)) (2011) 4424–4435.
[26] Gummeson J., Clark S.S., K. Fu, D. Ganesan, On the limits of effective hybrid micro-energy harvesting on mobile CRFID sensors, in: MobiSys'10, 2010.
[27] M. Gorlatova, Z. Noorbhaiwala, A. Skolnik, J. Sarik, M. Zapas†, M. Szczodrak, J. Chen, L. Carloni, P. Kinget, I. Kymissis, D. Rubenstein, G. Zussman, Prototyping Energy harvesting active networked tags: phase II mica mote-based devices, in: Mobicom, 2010.
[28] M. Gorlatova, A. Wallwater, G. Zussman, Networking low-power energy harvesting devices:measurements and algorithms, in: InfoComm, 2011.
[29] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Journal of Computer Networks 38 (December) (2001) 393–422.
[30] N. Sharma, J. Gummeson, D. Irwin, P. Shenoy, Cloudy computing: leveraging weather forecasts in energy harvesting sensor systems, in: Secon, 2010.
[31] A. Kansal, J. Hsu, S. Zahedi, M.B. Srivastava, Power management in energy harvesting sensor networks, ACM Transactions on Embedded Computing Systems 6 (4) (2006).
[32] U. Cetintemel, A. Flinders, Y. Sun, Power-efficient data dissemination in wireless sensor network, in: MobiDE'03, 2003.
[33] T. He, S. Krishnamurthy, J.A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, B. Krogh, Energy-efficient surveillance system using wireless sensor network, in: MobiSYS'04, 2004.
[34] J.R. Piorno, C. Bergonzini, D. Atienza, T.S. Rosing, Prediction and management in energy harvested wireless sensor nodes, in: Wireless VITAE, 2009.
[35] Mikkel K. Jakobsen, Jan Madsen, Michael R. Hansen, T.U. Denmark, DEHAR: a distributed energy harvesting aware routing algorithm for ad-hoc multi-hop wireless sensor networks, in: WOWMOM, 2010.
[36] Emanuele Lattanzi, Edoardo Regini, Andrea Acquaviva, Alessandro Bogliolo, Energetic sustainability of routing algorithms for energy-harvesting wireless sensor networks, Journal of Computer Communications 30 (14–15) (2007) 2976–2986.
[37] V. Pryyma, L. Boloni, D. Turgut, Uniform sensing protocol for autonomous rechargeable sensor networks, in: MSWiM, 2008.

Nga Dang is currently a Ph.D student in the Department of Computer Science at the University of California, Irvine. Her research interests include energy harvesting management for embedded system and wireless sensor networks, software/hardware co-design and hardware functional verification. Nga Dang received the B.S degree in Computer Engineering from National University of Singapore, Singapore in 2007 and the M.S degree in Computer Science from University of California, Irvine in 2011.

Elaheh Bozorgzadeh received the B.S. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 1998, the M.S. degree in computer engineering from Northwestern University, Evanston, IL, in 2000, and the Ph.D. degree in computer science from the University of California, Los Angeles, in 2003. She is currently an Associate Professor in the Department of Computer Science at the University of California, Irvine. Her research interests include design automation for adaptive embedded systems, reconfigurable computing, and green computing. Dr. Bozorgzadeh is recipient of NSF CAREER award and the Best Paper award in IEEE FPL 2006.

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001

Nalini Venkatasubramanian is currently a Professor in the School of Information and Computer Science at the University of California Irvine. She has had significant research and industry experience in the areas of distributed systems, adaptive middleware, pervasive and mobile computing, distributed multimedia and formal methods and has published extensively in these areas. As a key member of the Center for Emergency Response Technologies at UC Irvine, Nalini's recent research has focused on enabling resilient and scalable observation and analysis of situational information from multimodal input sources; dynamic adaptation of the underlying systems to enable information flow under massive failures and the dissemination of rich notifications to members of the public at large. Many of her research contributions have been incorporated into software artifacts which are now in use at various first responder partner sites. She is the recipient of the prestigious NSF Career Award, an Undergraduate Teaching Excellence Award from the University of California, Irvine in 2002 and multiple best paper awards. Prof. Venkatasubramanian has served in numerous programs and organizing committees of conferences on middleware, distributed systems and multimedia and on the editorial boards of journals. She received a M.S and Ph.D in Computer Science from the University of Illinois in Urbana-Champaign. Her research is supported both by government and industrial sources such as NSF, DHS, ONR, DARPA, Novell, Hewlett-Packard and Nokia. Prior to arriving at UC Irvine, Nalini was a Research Staff Member at the Hewlett-Packard Laboratories in Palo Alto, California.

Please cite this article in press as: N. Dang, et al., QuARES: A quality-aware renewable energy-driven sensing framework, Sustain. Comput.: Inform. Syst. (2012), http://dx.doi.org/10.1016/j.suscom.2012.08.001