DOI: 10.1016/j.compag.2017.01.026

Sensing-cloud: Leveraging the Benefits for Agricultural Applications

Tamoghna Ojha*†, Sudip Misra*, Narendra Singh Raghuwanshi†
*Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur
Email: {tojha, smisra}@sit.iitkgp.ernet.in
†Department of Agricultural and Food Engineering, Indian Institute of Technology Kharagpur
Email: nsr@agfe.iitkgp.ernet.in

Abstract

The advent of the sensor-cloud framework empowers the traditional wireless sensor networks (WSNs) in terms of dynamic operation, management, storage, and security. In recent times, the sensor-cloud framework is applied to various real-world applications. In this paper, we highlight the benefits of using sensor-cloud framework for the efficient addressing of various agricultural problems. We address the specific challenges associated with designing a sensor-cloud system for agricultural applications. We also mathematically characterize the virtualization technique underlying the proposed sensor-cloud framework by considering the specific challenges. Furthermore, the energy optimization framework and duty scheduling to conserve energy in the sensor-cloud framework is presented. The existing works on sensor-cloud computing for agriculture does not specifically define the specific components associated with it. We categorize the distinct features of the proposed model and evaluated its applicability using various metrics. Simulation-based results show the justification for choosing the framework for agricultural applications.

Index Terms

Sensor-cloud, virtualization, system model of sensor-cloud, sensor-cloud for agricultural applications

I. INTRODUCTION

In precision agriculture, WSNs are used to address different problems (e.g. [1]–[4]). Some of the existing works in this domain with WSN applications are categorized into irrigation management [5]–[10], precision agriculture [11]–[15], farmland monitoring [16], [17], greenhouse gases monitoring [18]–[20], agricultural production process management [13], [21], optimization of plant growth [22], and security and intrusion detection in fields [23], [24]. However, these WSN-based applications primarily target serving single application only, on which WSN is deployed by the users only at their specific interest area. Consequently, only the users (generally the user organization) has access to the data, and, thereby, they are in sole charge of the maintenance of the network. Third party access to this information is generally not enabled in this framework. Alternatively, data sharing may happen between organizations with exchange of money.

In recent times, the sensor-cloud framework has become very popular in various application domains. Compared to traditional WSNs, sensor-cloud provides numerous advantages. The science behind cloud computing empowers the distributed WSNs for enhanced storage and information processing capability. The integrated framework also creates a virtualized platform of sensors, which facilitates efficient and real-time information sharing among multiple users. The virtualization technique also enables dynamic resource management, which, in turn, increases resource utilization. Due to the abstraction of computing resources and efficient access control techniques, the overall architecture also provides information security. All these features the sensor-cloud framework suitable for real-time decision support in multi-user, multi-application scenarios.

The initial works in sensor-cloud focused on defining the infrastructure and its components [25]. Over the recent years, the concept of sensor-cloud and its architecture has matured [26]. In one of the initial works, the concept of physical sensor and its services virtualization was proposed by Evensen *et al.* [27]. Later, Ibbotson *et al.* [28] presented a semantically rich service oriented architecture (SOA), which focuses on simplification of sensor service discovery. Recently, Misra *et al.* [29] presented a theoretical model of sensor-cloud, which mathematically formulates the underlying virtualization technique involved in this technology. The authors promoted the concept of Sensor-as-a-Service (Se-aaS) [29] and showed the benefits of the architecture in terms of cost effectiveness, lifetime of sensor nodes, and fault-tolerance. Madria *et al.* [30] present an architecture for sensor-cloud which define the different part of the protocol stack and interconnections with physical sensors as well as users. In this work, the authors envision the sensor-cloud protocol stack comprised of three vertical layers – *sensor-centric*, *middleware*, and the *client-centric*. The Agriculture Sensor-Cloud Infrastructure (ASCI) [31] shows how various agricultural services can be offered via the sensor-cloud platform. The ASCI framework also devises a layered architecture, which shows the integration of various deployed sensors with different services. Krintz *et al.* [32] proposes an open source, cloud-based agricultural analytics service named SmartFarm. This platform integrates various different technologies such as satellite imagery, weather predictions, and existing data sets with on-farm sensors. One major objective of this platform is to provide a cost-effective platform for data analytics ensuring data privacy.

A. *Motivation*

The sensor-cloud framework is strongly founded on the principle of virtualization. The cloud provides the facility for storage and retrieval of huge amount of sensed data. Using the virtualization concept, the service providers are empowered with the power of greater sensor utility, while maintaining information security. In this distributed framework, the sensor owner is responsible for the deployment of sensor nodes. The service provider takes care of the maintenance and deployment overhead for the deployed nodes. The end-users consume the sensed information through various services offered by the service providers. Thus, the users are relieved from the task of deployment, maintenance, redeployment, system up-gradation, and any such works.

In Figure 1, we depict the structural comparison of the architecture for WSN and sensor-cloud. In this figure, we depict that WSNs are envisioned to work with single user and single application. On the other hand, sensor-cloud provides a framework supporting multi-user and multi-applications. The end-users subscribe to these various services which are running as applications in the framework. Unlike the traditional WSNs, in sensor-cloud we can construct different access levels which ensures information safety for different levels of users.

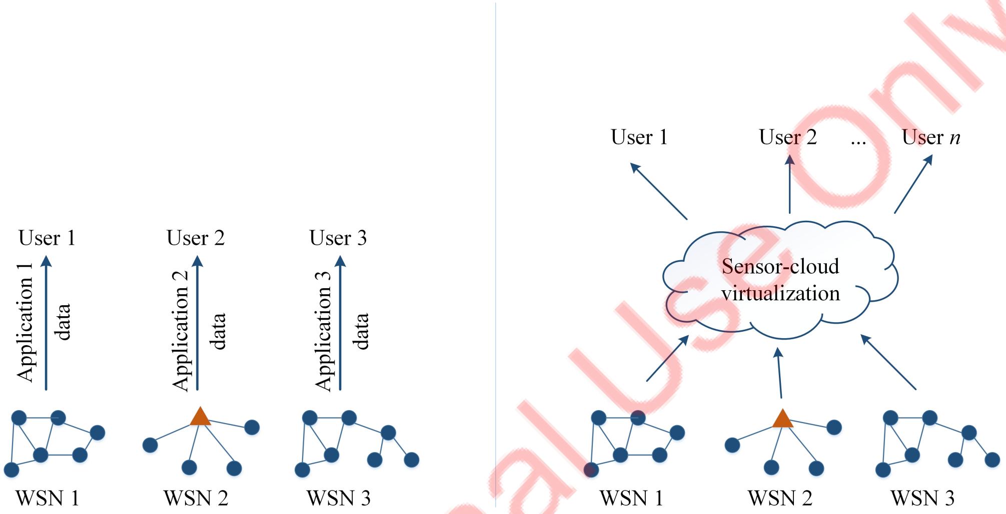


Fig. 1: Comparison of architectures: WSN vs sensor-cloud

In agricultural applications, this framework is very helpful due to its cost effectiveness and minimal maintenance requirements. The end-user, typically a farmer, has no burden of deploying and maintaining the field sensors unlike what would happen if they had used the conventional WSNs. From the service providers point-of-view as well, the sensor-cloud framework provides enhanced benefits. Unlike WSNs, in sensor-cloud, the service provider is able to utilize the deployed sensors for multiple applications and services. In turn, the service providing organization is able to provide the services to more number of people. The distributed framework also guarantees certain amount of fault-tolerance for the services. This is of great help specifically for agricultural applications, where sensor nodes face harsh climatic conditions leading to fault-proneness of the nodes. Thus, the sensor-cloud framework has the potential for leveraging benefits for both the end-users and service providers.

B. *Contributions*

In this paper, we present a sensor-cloud architecture for the agricultural applications. We present a mathematical model of sensor-cloud virtualization specifically targeting agricultural applications. The mathematical model includes details on different components involved in virtualization. Using case studies, we point out the specific benefits of this new framework over the existing WSN-based framework. Simulation-based results are presented for both these frameworks. In the following, we list the major contributions of our work.

- We present the physical node virtualization model for agricultural applications. Mathematically, we justify the advantages of the sensor-cloud framework over the traditional WSN-based framework.
- We formulate the sensor node utilization model targeting any agricultural application. The theoretical model presented in this paper focuses on building up a virtual sensor configuration, which enhances node utilization.
- We present a model for providing cost effective agricultural computing services to large number of farmers.
- The theoretical model depicted in this paper is suitable for a multi-organization, multi-user, and multi-application scenario. This is a significant paradigm-shift from the typical agricultural applications.

C. *Organization*

The paper is organized as follows. In Section II, we briefly discuss the existing literature in the area of sensor-cloud. The proposed system architecture is depicted in Section III highlighting the specific challenges and advantages of the sensor-cloud framework in the agricultural domain. The energy consumption optimization and virtualization models are presented in Sections IV and V, respectively. We also discuss, in Section VI, two potential applications, as case studies, which can be offered using the sensor-cloud framework. In Section VII, the simulation-based results are presented and discussed. Finally, the paper concludes in Section VIII, citing direction for future works.

II. RELATED WORKS

Over the recent years, the concept of sensor-cloud has drawn lot of attention. A detailed survey of the sensor-cloud framework, applications, and its pros and cons is presented by Alamri *et al.* [26]. One of the initial works by Evensen *et al.* [27] proposed SenseWrap as a middleware for virtualizing any type of physical sensors. They also presented ZeroConf [27], a protocol which lets the applications programmers interact with the sensors without the need of knowing their physical configuration. In this work, the authors mainly focus on sensor abstraction to facilitate the discovery of services and devices using a common interface. However, the virtualization framework presented in this work does not look into various important issues such as security, resource access, and dependency management. Ibbotson *et al.* [28] presented the challenges of various sensor networks connected by heterogeneous communication infrastructures. The authors present a semantically rich service-oriented architecture (SOA) which simplifies the sensor discovery, access control, sensor data consumption, and utilization.

Yuriyama *et al.* [25] presented a detailed description of sensor-cloud infrastructure. This infrastructure enables the on-field sensors in a cloud framework to facilitate the virtualization of resources. Through this infrastructure, the deployed sensors can be accessed and controlled from the end-users' side. Liu *et al.* [33] described a sensor-cloud architecture, which is based on an extended architecture of the CloudMiner [34]. Liu *et al.* [33] envisioned this architecture to be a platform offering various computational, analytical, and storage services. Liu *et al.* [33] addressed the use of virtualization of sensors in the CloudMiner architecture to capture and process sensor data for enabling new applications, and consequently, providing various services that can be availed based on that.

Dynamic duty scheduling for minimizing the energy consumption of the sensor nodes in a sensor-cloud environment is presented in [35]. The authors propose an algorithm to dynamically select a optimal duty interval for each WSN irrespective of the others. Misra *et al.* [36] studied the problem of ensuring Quality of Service (QoS) in a mobile sensor-cloud environment. The authors show that, instead of bandwidth shifting, bandwidth redistribution is required to maintain QoS in such environment. [37] showed how to allocate sensors while maintaining QoS for target tracking applications in a sensor-cloud environment. In an another work, Misra *et al.* [29] presented a theoretical model of sensor-cloud. The authors mathematically define virtualization of sensors in the architecture, and thereafter, the framework is analyzed in detail with respect to various performance evaluation metrics. For virtualization within the sensor-cloud framework, an optimal composition of a virtual sensor and an adaptive data caching method was proposed by Chatterjee *et al.* in [38] and [39], respectively. Chatterjee *et al.* [40] studied the issue of optimal data center selection. A pricing model for the sensor-cloud framework was presented in [41].

Madria *et al.* [30] present a sensor-cloud architecture and described the different parts of the protocol stack. In this work, the protocol stack is divided into three vertical layers – *sensor-centric*, *middleware*, and the *client-centric*. The bottom layer is *sensor-centric*, and as the name suggests, this layer directly interacts with the physical sensors and takes care of the registration, maintenance, and data collection. The *middleware* layer is responsible for handling incoming requests, creating virtual sensors, and computing billing information. Lastly, the *client-centric* layer provides the user interface (UI), manages sessions, checks membership, and manages user repositories. However, this work considers only static sensors to be present in the framework. The integration of WSNs with Mobile Cloud Computing (MCC) was studied by Zhu *et al.* [42], [43]. The authors proposed two different schemes to increase the sensor data reliability in cloud and to minimize the energy consumption of the deployed sensors. In the first scheme, the WSN gateway selectively sends sensory data which is more meaningful to the cloud, by considering the time and priority of requested data. The other scheme is for the sensors to minimize their energy consumption by optimizing the sleep-wake schedule.

Kim *et al.* [31] proposed the architecture of Agriculture Sensor-Cloud Infrastructure (ASCI), which shows how various agricultural services can be offered via the sensor-cloud platform. The ASCI framework devises a layered architecture, which shows the integration of various deployed sensors with different services. Distefano *et al.* [44] argue on the benefits of the device-centric approach of IoT over the data-centric one. In their proposed IoT-A reference architecture, the Sensing-and-Actuation-as-a-Service (SAaaS) approach is presented. In this architecture, various sensors, mobiles and personal devices are abstracted through a unified platform. A smart, cloud-controlled irrigation management system was proposed by Sales *et al.* [45]. The field implementation consists of a set of sensors and actuators deployed for monitoring plants' water requirements. On the other hand, the cloud-based framework hosts a weather forecast application, which helps in making optimal irrigation decision. Whenever the on-field sensor system determines the soil moisture of the field to be lower than the threshold value, the next 6 hour's weather forecast is checked, and then the decision on whether or not to irrigated is made. An open source, cloud-based agricultural analytics service named SmartFarm was proposed by Krintz *et al.* [32]. This platform integrates various different technologies such as satellite imagery, weather predictions, and existing data sets with on-farm sensors. One major objective of this platform is to provide a cost-effective platform for data analytics ensuring data privacy. Additionally, the authors propose an app named RootRApp, which analyzes the SmartFarm sensor data to find the cause of wine grape quality difference.

III. PROPOSED SYSTEM ARCHITECTURE

In the sensor-cloud framework, we consider n number of on-field sensor networks ($\mathcal{W}$), which are deployed at different locations. For example, the set of all the on-field WSNs is represented by $\mathcal{W} = \{W_1, W_2, \cdots, W_i, \cdots, W_n\}$. In any such WSN, say $W_i \in \mathcal{W}$, the on-field sensor nodes and the gateway are denoted by S_i and G_i, respectively. The set of all the gateway nodes is denoted by $\mathcal{G}$. Any sensor node $j \in S_i$ communicate with the cloud through the gateway node G_i. Consequently, the sensor-cloud renders efficient and smart decisions based on the sensed data. We consider any on-field WSN to be represented by a graph $G(S_i, E)$, where S_i and E represent the set of sensor nodes in any WSN W_i and the set of communication links between the nodes. For example, in this graph G, node i and j are connected, e.g. $\overline{i,j} \in E$, *iff* $d_{ij} \leq r$, where r is the transmission range of nodes i and j. Here, we assume that the on-field nodes have similar transmission range. The on-field nodes $j \in S_i$ follow the duty values ($\tau^*_{i,t}$) computed by the cloud, and communicate their sensed information to the gateway G_i at specific times. The dynamic duty values are computed by sensor-cloud framework using the dynamic duty selection scheme discussed in Section IV.

We depict the system architecture in Figure 2. In this figure, we show how these n number of on-field WSNs communicate with the cloud via their local gateways. In each of these WSNs, actual topologies are formed by the deployment of sensor nodes at that area. The local topology can also support multi-hop communication between the sensor nodes and the gateway. Needless to say, the end-users are unaware of the actual topology and connectivity. They are able to monitor the information of the nodes, which are pertinent to their subscribed services. On the other hand, the sensor owners can access status/maintenance related information about their deployed nodes in various locations. In Section V, we present the detailed discussion on the components of each of these various entities and the relations among them.

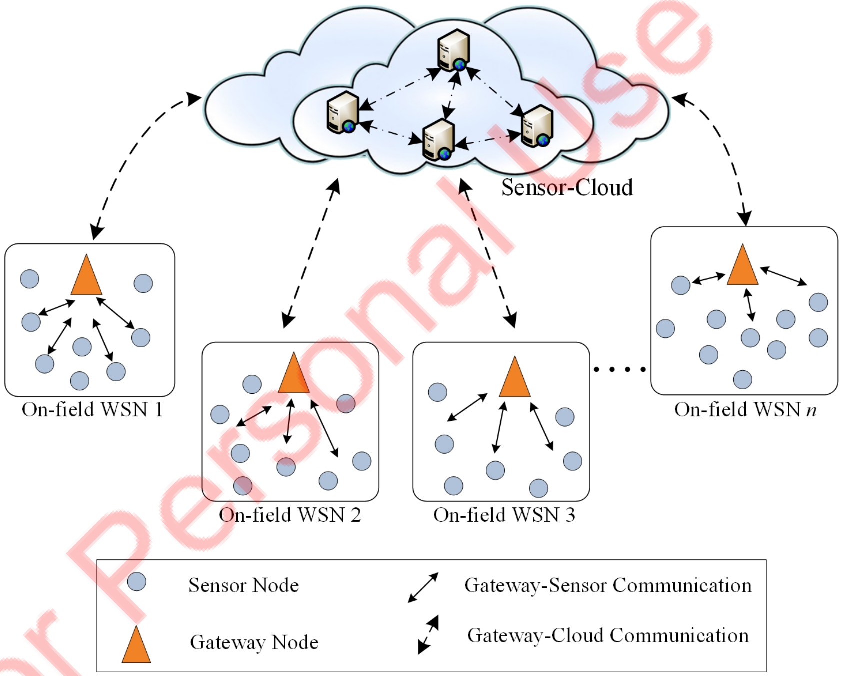


Fig. 2: The proposed system architecture

A. Challenges Specific to Agriculture

In this section, we list the challenges devising a sensor-cloud framework specific to agricultural applications. We also point out challenges specific to developing countries such as India.

i. *Low maintenance*: The agricultural applications are desirable to have low maintenance effort. This will essentially keep the system's maintenance cost very low. Low maintenance is desirable in these systems due to the fact that a large number of end-users of these systems are typically non-technical persons.
ii. *Scalability*: In agricultural applications, the monitoring area may cover a large area with multiple fields. Thus, it is desirable to have a scalable system with the change in target application area.
iii. *Cost-effective solutions required*: The overall system cost is required to be low for larger scope and outreach of the applications. The challenge is to reduce the system cost targeting Low and Middle Income Countries (LMICs).
iv. *Small and irregular land holding*: In developing countries such as India, the average land holding of a farmer is low, and the field shape is also irregular. Moreover, farmers may have non-contiguous fields allocated for them. Based on these, the field level heterogeneity is high for agricultural applications.

v. *Easy of operation*: The agricultural system and applications will be used by non-technical persons also. Thus, it is desirable to keep the operations and interactions with the system minimum.

vi. *Fault-tolerance*: Typically, the deployed systems in agricultural applications have to function in harsh environmental conditions, as they are deployed mostly at outdoor locations. Thus, tolerance against communication and other systemic failures are inevitable.

B. Advantages of Sensor-cloud Architecture Over WSN

In the following, we enlist the specific advantages of applying sensor-cloud architecture over the traditional WSN.

i. *Scalability*: Sensor-cloud architecture supports scalability of physical sensors with application/service demands. This architecture facilitates that the sensors can be accessed by multiple services and multiple users without any intervention.

ii. *Cost*: In sensor-cloud, the end-user is relieved from most of the cost incurring tasks such as sensor deployment, maintenance, and other overheads related to system up-gradation. However, he/she is only required to subscribe to services, and pay-as-per-use of the subscribed services. On the other hand, the service provider utilizes the deployed infrastructure by using it for multiple users and multiple services. Consequently, the services can be made cheaper for individual users while generating sustainable revenue for service providers.

iii. *Lifetime*: The advent of sensor-cloud provides optimal duty scheduling of the deployed sensor nodes. Due to this, the energy consumption of each sensor node can be reduced. Therefore, the overall network lifetime also increases significantly.

iv. *Reconfigurability*: Sensor-cloud provides multiple services to end-users. Based on the available services, the user can choose to switch applications, and virtually reconfigure the sensor nodes as per their demand.

v. *Fault-tolerance*: In WSNs, faults can occur randomly and can disrupt the network. However, in the sensor-cloud framework, the service provider provides fault-tolerance guarantees to end-users through service-level agreement (SLA). It is envisioned that sensor-cloud framework will bring in multiple service providers on a unique platform. Thus, the users will have option to choose and switch to different service providers with time.

vi. *Information Security*: Virtualization techniques enable the sensor-cloud to provide access control to users. It caters to the various layers of abstraction on the stored information and the physical sensors, thereby providing security to various levels of end-users.

vi. *Farmland Security*: Ensuring security of farmland and crop is an essential requirement for the agricultural applications. The sensor-cloud framework provides flexibility to add additional sensors or up-gradation of deployed system. Bapat *et al.* [46] presents a WSN-based system for detecting animal intruders and subsequently diverting them using deterring gadgets. Any existing sensor-cloud based deployment can be upgraded with installation such sensor nodes. Compared to the traditional WSNs, in the sensor-cloud architecture, the up-gradation procedure is simple and scalable. The service provider, thus, is able to provide additional services on security and intrusion detection after deployment of required sensors. Also, in case a field's data is contaminated, the abnormality in the recorded data can be detected by various existing anomaly detection techniques.

vii. *Quality of service (QoS)*: In a sensor-cloud, QoS guarantee is also defined through the use of SLAs. Thus, the end-users can choose and switch among different service providers.

viii. *Dynamic management*: The sensor-cloud framework facilitates dynamic resource sharing by applying the theory of resource abstraction [29], [36], [37]. Consequently, end-users are allowed to upgrade the service levels. Resources are dynamically allocated to services in synchronization with the change in the requirements.

IV. ENERGY OPTIMIZATION AND DUTY CYCLE COMPUTATION

In this section, we show and compare the energy consumption of typical on-field WSN and the sensor-cloud framework. We adopt the optimal duty scheduling approach proposed in our previous work [35]. Using this approach, the on-field sensor nodes, in a sensor-cloud framework, are able to optimize their energy consumption.

A. Energy Consumption of On-field WSN

In any on-field standalone WSN $W_i \in \mathcal{W}$ (i.e. without the cloud framework), the total energy consumption ($\mathcal{E}_{W_i}(t)$) at any time t caters to the following events – communication between the sensor node $j \in W_i$ and the gateway node $G_i \in W_i$ ($\mathcal{E}_{s,g}(t)$) and computation and information processing inside the node ($\mathcal{E}_p(t)$). Thus,

$$\mathcal{E}_{W_i}(t) = \mathcal{E}_{s,g}(t) + \mathcal{E}_p(t) \tag{1}$$

The overall energy consumption during communication between the sensor nodes ($\forall j \in W_i$) to the gateway $G_i \in W_i$ is defined as,

$$\mathcal{E}_{s,g}(t) = \sum_{j \in W_i} \aleph_j^{\tau}(t) e_{s,g}^{j} \tag{2}$$

where $\aleph_j^{\tau}(t)$ is the sensed information by node $j \in W_i$ at time t for the time-interval τ. The entity τ is set by the gateway node $G_i \in W_i$ and its value is determined only on local observations. Any node j's energy consumption for communication with the gateway node is denoted as $e_{s,g}^{j}$.

Similarly, the entity $\mathcal{E}_p(t)$ is computed as follows,

$$\mathcal{E}_p(t) = \sum_{j \in W_i} \aleph_j^{\tau}(t) e_p^j \tag{3}$$

where the energy consumption in node $j \in S_i$ for its own computation and information processing is e_p^j.

Therefore, we can compute the total energy consumption of an on-field WSN W_i as follows:

$$\mathcal{E}_{W_i}(t) = \sum_{j \in W_i} \aleph_j^{\tau}(t) \left(e_{s,g}^j + e_p^j \right) \tag{4}$$

Thus, in an on-field WSN W_i, the optimization problem for overall energy consumption is expressed as follows,

$$\text{Minimize} \tag{5}$$

$$\mathcal{E}_{W_i}(t) = \sum_{j \in W_i} \aleph_j^{\tau}(t) \left(e_{s,g}^j + e_p^j \right)$$

subject to

$$e_{s,g}^j|_{min} \leq e_{s,g}^j \tag{6}$$

$$e_p^j|_{min} \leq e_p^j \tag{7}$$

$$\tau \leq \Gamma \tag{8}$$

Thus, finally, the overall energy consumption for the whole on-field WSNs is,

$$\mathcal{E}(t) = \sum_{W_i \in \mathcal{W}} \mathcal{E}_{W_i}(t) \tag{9}$$

B. Optimization of Energy Consumption in the Sensor-cloud Framework

On the other hand, in the sensor-cloud architecture, the energy optimization problem is formulated with the help of various components of energy consumption within the network. The three major components of total energy consumption ($\mathcal{E}_{W_i}^*(t)$) are – information uploading from sensor to gateway ($e_{s,g}^j$), gateway to cloud ($e_{g,c}^u$), and in-cloud processing (e_c^p). Therefore, in summary, the optimization problem is formulated as,

$$\text{Minimize} \tag{10}$$

$$\mathcal{E}_{W_i}^*(t) = \sum_{j \in W_i} \aleph_j^{\tau^*}(t) \left(e_{s,g}^j + e_{g,c}^u \right) + \sum_{j \in W_i} \bar{\aleph}_j^{\tau^*}(t) e_c^p$$

subject to

$$e_{s,g}^j|_{min} \leq e_{s,g}^j \tag{11}$$

$$e_{g,c}^u|_{min} \leq e_{g,c}^u \tag{12}$$

$$e_c^p|_{min} \leq e_c^p \tag{13}$$

$$\bar{\aleph}_j^{\tau^*}(t) \subseteq \aleph_j^{\tau^*}(t) \tag{14}$$

$$\tau^* \leq \Gamma \tag{15}$$

where $\aleph_j^{\tau^*}(t)$ and $\bar{\aleph}_j^{\tau^*}(t)$ represents the information sensed by node j and the subset of the information at time t for an interval of τ^*. τ^* denote the value of optimal time-interval for any W_i at time t, and it can be expended as $\tau^* = \tau_{i,t}^*$.

We show the computation of overall energy consumption $\mathcal{E}^*(t)$ for all the on-field WSNs as follows,

$$\mathcal{E}^*(t) = \sum_{W_i \in \mathcal{W}} \mathcal{E}_{W_i}^*(t) \tag{16}$$

C. Optimal Duty Scheduling

The sensor-cloud framework collects the on-field information from the deployed sensor nodes at different intervals of time. It is noteworthy to mention that the time-interval of information collection is controlled by the sensor-cloud. The sensor-cloud framework can optimally decide on this parameter, and consequently, the energy consumption of the on-field WSNs can be minimized. In this regard, the following assumption is taken.

Assumption 1. *For any deployed WSN W_i, the sensor-cloud framework calculates the required information update time-interval ($\tau_{i,t}$) from the received information [42].*

The cloud-based framework computes the required time-interval $\tau_{i,t}$ for any on-field WSN W_i, as follows:

$$\tau_{i,t} = f(\aleph_t^*) \tag{17}$$

where $\aleph_t^*$ represents the set of information collected by the cloud from the deployed sensor nodes at time t. $\aleph_t^*$ is calculated using the following equation.

$$\aleph_t^* = \bigcup_{W_i \in \mathcal{W}} \aleph_i(t) \tag{18}$$

The optimal value of the time-interval in each step is computed with the help of time-interval determined at the t^{th} and $(t-1)^{th}$ instance. Mathematically,

$$\tau_{i,t}^* = \alpha_t \tau_{i,t} + \alpha_{t-1} \times \tau_{i,t-1}^* \qquad \forall \alpha_t, \alpha_{t-1} \in [0,1] \tag{19}$$

where $\tau_{i,t-1}^*$ is the value of optimal time-interval determined at the $(t-1)^{th}$ instance.

V. VIRTUALIZATION MODEL

In this section, we present the underlying virtualization model for the sensor-cloud framework targeting agricultural applications. We mathematically define the participating units of sensor virtualization. We define the units in two different types – *actors* and *entities*. The *actors* associated with the virtualization process are – the service providers, sensor owners, and the end-users. On the other hand, *entities* are the distinct components of virtualization related to sensors and services/applications. Thus, we define the different functions which represents the actual mapping between the different entities and actors.

A. Actors

We, in the following, mathematically define the components of different types of actors associated with the sensor-cloud framework.

Definition 1. *In sensor-cloud, multiple service providers can coexist in the same framework. We name this actor as Cloud Service Provider (CSP), defined as $CSP = \{CSP_1, CSP_2, \cdots, CSP_\kappa\}$, where κ is the number of service providers.*

In the following, we use the work 'service provider' and 'cloud service provider' interchangeably. CSPs are differentiated by their SLAs, which defines the critical service level parameters such as QoSs and fault-tolerance. Based on the region of deployment, which may be different sub-CSPs offering similar services.

Definition 2. *Any CSP (CSP_i) is defined as a tuple with the following parameters:*

$$CSP_i =< CSP_{id}, CSP_{type}, CSP_{zone}, CSP_\Phi, SLA_{type} > \tag{20}$$

where CSP_{id} is a unique identifier of the CSP and CSP_{type} indicates the type of the CSP – *national* or *regional*. CSP_{zone} holds the specific zone(s) the CSP is operating in. CSP_Φ denotes the set of services which are offered by this CSP and SLA_{type} is specific to the Service-level agreement between the CSP and the end-user.

Definition 3. *A sensor owner is defined as $\theta_i \in \Theta = \{\theta_1, \theta_2, \cdots, \theta_m\}$, where m is the number of owners registered at the sensor-cloud framework at any point of time. The associated parameters for a sensor owner are:*

$$\theta_i =< \theta_{id}, \theta_{type}, \theta_{CSP}, \theta_\Phi > \tag{21}$$

where θ_{id} and θ_{type} are the sensor owner's unique identifier and type, respectively. Similarly, θ_{CSP} and θ_Φ denote the CSPs and services associated with the sensor owner θ_i. Sometimes, a CSP organization may also deploy the sensor nodes, and thus, is recognized as a sensor owner.

Definition 4. *The end-user is denoted as $U_i \in U = \{U_1, U_2, \cdots, U_n\}$, where n is the number of users. End-user is a four-tuple actor defined as:*

$$U_i =< U_{id}, U_{type}, U_{CSP}, U_\Phi > \tag{22}$$

where U_{id} and U_{type} are the user's unique identifier and type, respectively. Similarly, the CSPs and services associated with the user U_i are denoted by U_{CSP} and U_Φ.

B. Entities

The different entities, which forms the essential components of the sensor-cloud framework are mathematically defined in the following. In addition to the details about these components, we also discuss the inter-relation between them.

Definition 5. *A physical sensor node $w_i \in W$ is defined as a tuple using the following parameters,*

$$w_i =< w_{id}, w_{type}, w_{state}, w_{loc}, w_\theta, w_{CSP} > \tag{23}$$

where w_{id}, w_{type}, w_{state} and w_{loc} denote the physical sensor's identifier, type, state, and deployment location respectively. w_θ denotes the owner of the sensor. The set of CSP which has access to this sensor are denoted by w_{CSP}.

Definition 6. *A virtual sensor $s_i \in \mathcal{S}$ is defined as a tuple comprising of the following parameters:*

$$s_i =< s_{id}, s_{type}, s_{state}, s_{loc}, s_{CSP}, s_\phi, s_{crop}, s_\mathcal{A}, s_\chi > \tag{24}$$

where s_{id}, s_{type} and s_{state} denote the virtual sensor's identifier, type, and state, respectively. The location of the deployed sensor is indicated by s_{loc}. The set of CSPs which has access to s_i is denoted by s_{CSP}, s_ϕ defines the services associated with s_i, and s_{crop} refers to the set of the crops associated with the sensor s_i. The influence area of a deployed sensor is denoted by $s_\mathcal{A}$. A sensor may also be used for deriving values for different types of sensors. In this case, s_χ denotes the set of the possible derived sensors.

The virtual sensor id s_{id} for any particular sensor s_i is defined by the CSP. Therefore, for a specific CSP, s_{id} is unique. Mathematically,

$$s_i.s_{id} \neq s_j.s_{id} \quad \forall s_i, s_j \in \mathcal{S}; CSP_k \in CSP \tag{25}$$

Definition 7. *A service $\phi \in \Phi$ is defined as:*

$$\phi =< \phi_{id}, \phi_{type}, \phi_\mathcal{S}, \phi_{cov}, SLA_\phi > \tag{26}$$

where ϕ_{id} and ϕ_{type} denote the service identifier and its type, respectively, and $\phi_\mathcal{S}$ represents the set of sensors for any particular CSP attached with this service. For example, the virtual sensor s_i for CSP_k is defined as,

$$s_i \in \{CSP_k.CSP_\Phi.\phi_\mathcal{S}\}; \qquad CSP_k \in CSP \tag{27}$$

Definition 8. *The coverage of a service is defined as the total influence area ($s_\mathcal{A}$) covered by the sensors associated with the service. Mathematically,*

$$\phi_{cov} = \bigcup_{s_i} s_i.s_\mathcal{A} \quad \forall s_i \in \{CSP_k.CSP_\Phi.\phi_\mathcal{S}\}; CSP_k \in CSP \tag{28}$$

Definition 9. *Service-level agreement (SLA) for any service $\phi \in \Phi$ is denoted by a tuple comprising of the following parameters:*

$$SLA_\phi =< SLA_{id}, QoS_{th}, fault_{th}, \mathcal{R}(\cdot) > \tag{29}$$

where $\mathcal{R}$ denotes the access levels granted to the end-user. Thus, using this metric, we can clearly define the actors which are given access to the service information at various levels.

C. Functions

In this process, the allowed sensor types are also checked while granting access to a certain service, and consequently, access to the attached sensors. Any service has a specific set of sensors which are compatible with the service type, and thus, may be attached with it. Furthermore, the access between the actors and the field sensors depends on the security levels i.e. access levels defined in the SLA.

Definition 10. *We define the relation between the set of sensor types ($\mathbb{T}$) and the service type (τ) using the Sensor-Service Compatibility (S2C) function $\Gamma(\cdot)$. This compatibility check ensures proper filtering of the information for rightful access to the actors.*

$$\Gamma(\tau_i) = \{\mathbb{T}_i | \mathbb{T}_i \in \mathbb{T}; \tau_i = \phi_i.\phi_{type}\} \tag{30}$$

Here τ_i the service type of i^{th} service.

Definition 11. *The ingress rules $\mathcal{R}(\cdot)$ which define the actors and the manner in which they gain access to the service related information. Therefore,*

$$\mathcal{R}(x_i) = \{s_i | s_i \in \mathcal{S}; s_i.s_\phi.SLA_\phi \in SLA\} \qquad x_i = \{CSP_i, \theta_i, U_i\} \tag{31}$$

where x_i denotes an actor, which may be CSP, sensor owner, or an end-user.

Definition 12. *The mapping between a physical sensor and a virtual sensor is defined by the function* $f_{p2v}(\cdot)$.

$$s_i = f_{p2v}(w_j) \qquad s_i \in \mathcal{S}, w_i \in W \tag{32}$$

A physical sensor can be attached to multiple virtual sensors of different CSPs. On the other hand, a virtual sensor can also be attached with multiple physical sensors.

Definition 13. *The selection function* $f_{sel}(\cdot)$ *is defines the relation between a service and the concerned sensor nodes. For any offered service, the concerned sensor nodes can be found using this function. Therefore,*

$$\phi_{\mathcal{S}} = f_{sel}(\phi_i) \qquad \forall \phi_i \in \Phi, s_i \in \mathcal{S} \tag{33}$$

The function returns the set of concerned sensor nodes $\phi_{\mathcal{S}}$ for service ϕ_i having a coverage of ϕ_{cov}. The concerned SLA is defined by SLA_{ϕ}.

VI. CASE STUDIES – TYPICAL APPLICATIONS

In this section, we discuss two typical agricultural applications designed using the sensor-cloud framework.

A. *Irrigation Management*

Figure 3 shows a typical application of irrigation management using the sensor-cloud framework. In this application, the on-field sensor network is deployed at the users' fields with application specific sensors. For example, water level sensors can be used only for crops which require standing water, and soil moisture sensors for other cases. However, sometimes, both these sensors can be present in the nodes helps the node to be reusable for different types of applications. It also empowers the node to measure whether the field is water-logged or not in case of heavy rain. In such situation, actuation of water removing pumps can be possible with the help of water-logging information. Optionally, the nodes can have soil temperature sensors, ambient temperature and humidity sensors as well.

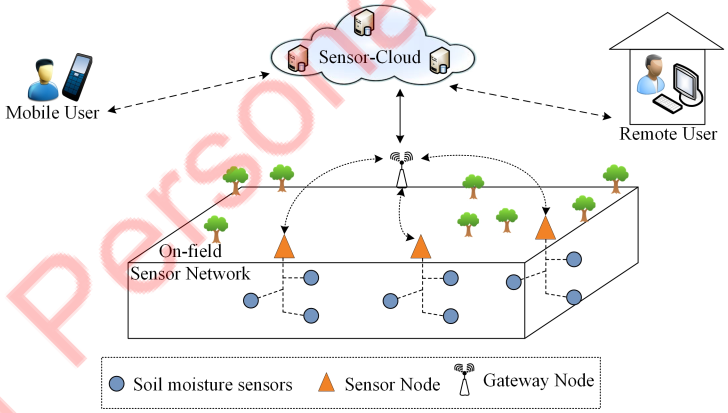


Fig. 3: Typical application scenario: irrigation management using sensor-cloud framework

In the sensor-cloud framework, these nodes are deployed by the service providers or sensor owners. The end-users subscribe to the irrigation management service and access the on-field information. Thereby, they are able to monitor the change in field conditions, and are able to control the irrigation schedule. In this framework, the micro-weather information is also taken into consideration while determining the proper irrigation schedule [45].

B. *Crop Disease Monitoring*

Crop disease monitoring is a potential service that can help the farmers in taking counter measures against the crop diseases. In Figure 4, we depict a example scenario of crop disease monitoring with on-field sensors and Unmanned Aerial Vehicles (UAVs). The service offers dynamic monitoring of the crop disease. The UAVs (drones) are deployed for on-demand use of pesticides at specific locations. Limited use of pesticides and fertilizers also enhances the crop quality, while keeping the farming cost lower. In the cloud-based framework, the local climate related information [47] such as ambient temperature, humidity, wind speed can also be included while taking decisions on crop disease. One recent work by Gonçalves *et al.* studied the influence of mobility in precision spray based systems with WSN. In Ref. [48], UAVs are used for video sensing in precision agriculture.

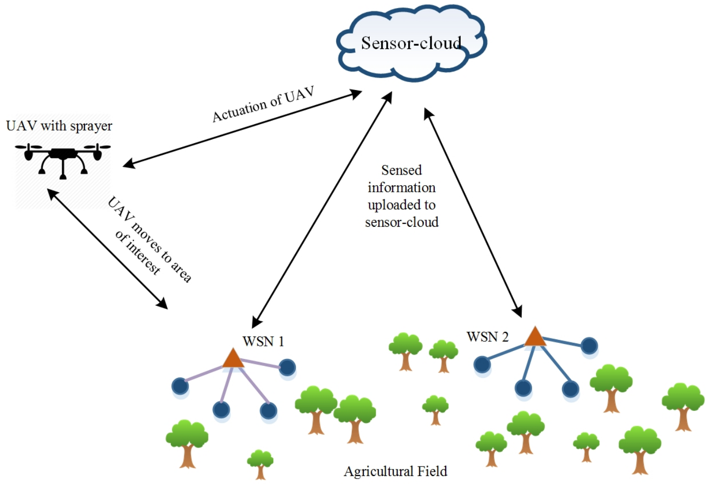


Fig. 4: Typical application scenario: crop disease monitoring using sensor-cloud framework

TABLE I: Simulation Parameters

Parameter	**Value**
Number of WSNs	10
Number of nodes in a WSN	50 - 100
Simulation Area	10 Km × 10 Km
Transmission Range of a sensor node (r)	100 m
Initial energy of a node	200 J
Threshold battery level (W_{th})	25 J

VII. PERFORMANCE EVALUATION

A. Simulation Settings

To evaluate our proposed scheme, we perform simulations using the NS-3 (http://www.nsnam.org/) simulator (version: 3.14). Table I enlists the simulation parameters. We use different performance metrics and evaluate the results with respect to the metrics.

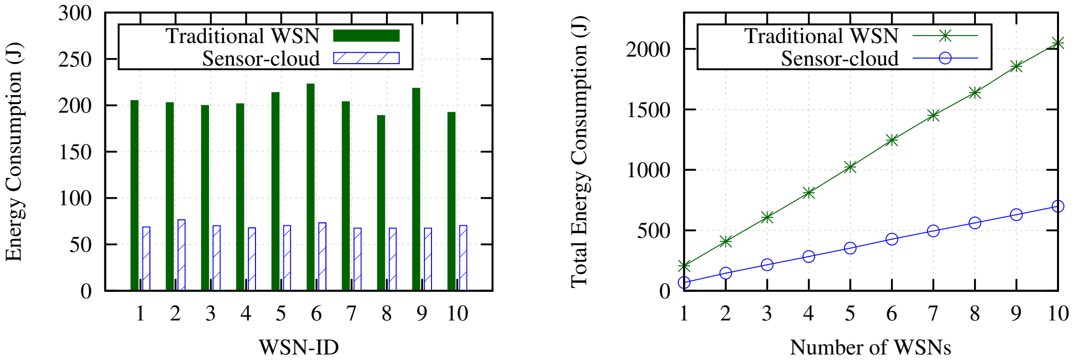
In the simulations, we simulated two different scenarios – traditional WSN (without the sensor-cloud framework) and the proposed sensor-cloud based framework. We considered a total of 10 different WSNs deployed in a field area spanning 10 Km × 10 Km. Each WSN consists of randomly 50-100 nodes. The nodes in any WSN send their data to cloud at ‘duty intervals’ defined by the cloud. Accordingly, we calculate the energy consumption of the on-field WSNs, and compute the remaining lifetime of the WSNs. In our experiments, we considered the following unit cost values: $c_{depl} = 1$, $c_s = 2$, $c_{repr} = 0.3$, and $c_{sc} = 1$. We also considered 20% of the nodes to be faulty.

We simulate such a network keeping in mind the irrigation management application presented in Section VI-A. In the simulation based experiment, we considered 10 different WSNs deployed over a area of 10 Km × 10 Km. Here, each WSN consists of 50–100 nodes, on an average. For simplicity, we keep the simulated nodes to be homogeneous. This type of configuration mimics an irrigation management application where the sensor nodes are deployed at different fields covering a vast area. These sensor nodes communicate with the cloud via their local gateway. Thereafter, we performed dynamic duty scheduling for these WSNs and accordingly compute their network lifetime. We show the utility of sensor-cloud over the WSNs, and calculate the cost for end-users for both sensor-cloud framework and traditional approach.

B. Evaluation Metrics

We studied the performance of the proposed system model with respect to the following metrics:

- *Energy consumption*: The energy consumption of the deployed sensor nodes is computed for both traditional WSN and the sensor-cloud based framework. We define these entities in Sections IV-A and IV-B.
- *Duty*: In this work, duty is defined as the effective *active time* of a sensor node for a given period of time. In our simulations, the duty value for sensor-cloud framework is selected optimally, as explained in Section IV.
- *Network lifetime*: We define the network lifetime as in the percentage of nodes which has not yet depleted their battery or their battery level is greater than that of a predefined threshold level. Thus, the metric is evaluated as the percentage of alive nodes over time.
- *Utility*: Utility is defined as the *benefit* or *gain* achieved in proposed framework ($\mathcal{U}_{cloud}$) over the traditional framework ($\mathcal{U}_{WSN}$). We present the individual utility gain ($\mathcal{U}_{indv}(W_i)$) as well as the average utility gain ($\mathcal{U}_{ovAvg}(W_i)$) with different



(a) Energy Consumption of the On-field WSNs (b) Total energy consumption of the on-field WSNs

Fig. 5: Energy Consumption

number of WSNs. The individual utility reflects the utility gain of a specific WSN (e.g., W_i) in sensor-cloud framework over the traditional approach. On the other hand, average utility reflects the *overall* utility gain with multiple WSNs (e.g., W_1 to W_i) in a sensor-cloud framework over the traditional approach. Mathematically,

$$\mathcal{U}_{indv}(W_i) = \frac{\mathcal{U}_{cloud} - \mathcal{U}_{WSN}}{\mathcal{U}_{WSN}} \times 100\% \qquad \forall W_i \in W$$

$$\mathcal{U}_{ovAvg}(W_i) = \frac{\sum_{W_1}^{W_i} \mathcal{U}_{cloud} - \sum_{W_1}^{W_i} \mathcal{U}_{WSN}}{\sum_{W_1}^{W_i} \mathcal{U}_{WSN}} \times 100\% \qquad \forall W_i \in W$$

- *Cost*: We define the overall cost for the end-users in sensor-cloud (C_{SC}) and traditional WSN (C_{WSN}) as follows:

$$C_{SC} = n \times c_{sc} \qquad \forall n \in \theta_i.\theta_{\Phi}.\phi_{\mathcal{S}}; \forall \theta_i \in \Theta$$

where c_{sc} is the unit price per sensor allocated in the subscribed service.

$$C_{WSN} = n_1 \times (c_s + c_{depl} + c_{repr}) + n_2 \times c_{depl}$$

where c_s is the cost of each sensor, c_{depl} and c_{repr} are the cost of sensor deployment and repair, respectively, n_1 is the number of sensor nodes originally deployed, and n_2 is the number of faulty nodes, which are eventually replaced.

C. Results and Analysis

In the following, we present the simulation-based results for the traditional WSN and the sensor-cloud framework. We compare and analyze the performance of both the framework with different performance metrics.

1) Energy Consumption: In this section, the results for the energy consumption for both traditional as well as sensor-cloud scenarios are presented. Energy consumption for the individual on-field WSNs is presented in Figure 5(a). On the other hand, Figure 5(b) shows the results for the different number of WSNs and the corresponding cumulative energy consumption. In these scenarios, the energy consumption of the field nodes is controlled by the duty value selected for them. In the sensor-cloud framework, as presented in Section IV, the duty value is selected optimally. The results presented in Figures 5(a) and 5(b) indicate the the energy-efficiency of the sensor-cloud framework.

2) Duty: Figure 6 shows the duty values allocated to the on-field WSNs at different instants of time. These values are computed by the cloud dynamically, and are consequently allocated to the deployed networks. It is noteworthy that the duty values allocated to a WSN is fully independent of the duty values allocated to the other WSNs. This dynamic duty allocation is particularly useful in optimal energy scheduling for the on-field networks. Accordingly, the energy consumption in the sensor-cloud framework reduced over that of the traditional WSN based framework.

3) Network Lifetime: We present the results for the network lifetime of on-field WSNs for both the traditional and sensor-cloud based frameworks in Figure 7. In this figure, the results for traditional approaches are marked as ‘T-WSN’ (which stands for the acronym of Traditional WSN), and the results for the proposed sensor-cloud based framework are denoted by ‘S-C’ (which stands for the acronym of Sensor-cloud). The figure depicts the results for all the deployed WSNs (e.g. 1-10). For example, the on-field WSN is tagged with the legend ‘WSN1’. The energy consumption of the on-field WSNs in the sensor-cloud based framework is less than that of the traditional WSN based framework due to optimal duty scheduling. Also, unlike the traditional WSN based framework, multi-hop routing is not required in the sensor-cloud based framework. Consequently, the lifetime of the nodes in the on-field WSNs in the traditional approach depletes rapidly compared to the proposed approach.

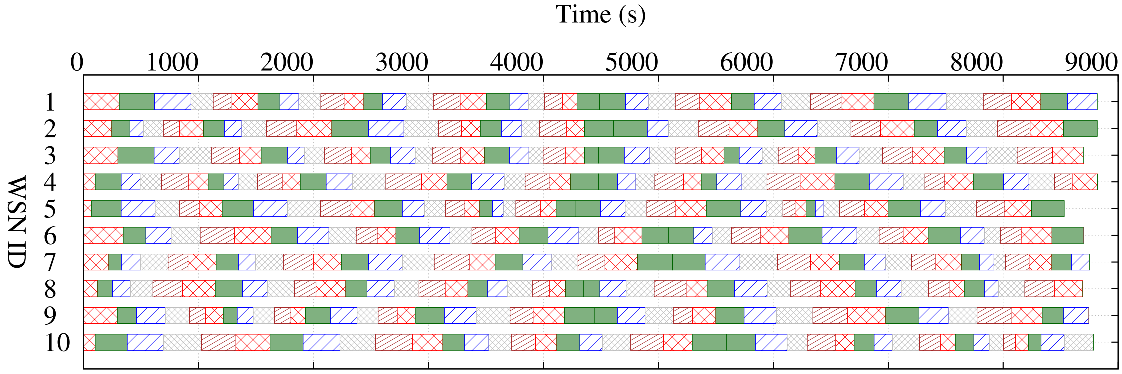


Fig. 6: Duty scheduling in the proposed scheme

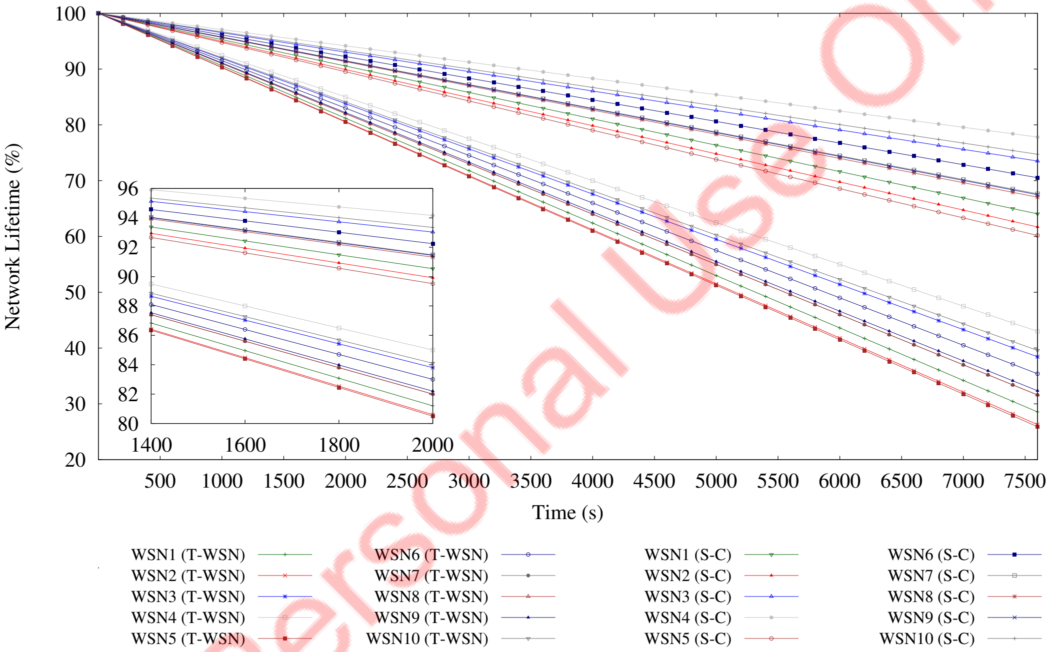


Fig. 7: Network lifetime for individual sensor networks with and without cloud

4) Utility: We present the utility of the proposed scheme over the traditional approach in Figure 8. We demonstrate the individual utility gain ($\mathcal{U}_{indv}(W_i)$) as well as the average utility gain ($\mathcal{U}_{ovAvg}(W_i)$) with different number of WSNs. The individual utility reflects the utility gain of that specific WSN (e.g., W_i) in the sensor-cloud framework over the traditional approach. On the other hand, the average utility reflects the *overall* utility gain with that many number of WSNs (e.g., W_1 to W_i) in sensor-cloud framework over the traditional approach. Therefore, when the number of WSN is 1, the individual and cumulative utilities are the same. However, sometimes the individual utility is greater than the cumulative utility, and the vice versa may also be possible. For example, for WSN id 7, the individual utility is greater than the cumulative utility, which refers to the situation that the other WSNs, with ids 1-6, have less utility than that of WSN id 7, and as result, the average utility is less than the individual utility gain. On the other hand, for WSN id 8, the opposite happens, and thus, the average utility gain is greater than the individual utility gain.

5) Cost: We analyze the cost incurred to any end-user in the sensor-cloud framework as well as in the traditional WSN based framework. In Figure 9(a), the cost incurred by the end-users in different WSNs is shown. The total cost with increasing number of on-field WSNs is plotted in Figure 9(b). The sensor-cloud remains cost-effective to the end-users, as they are only charged for their service access. The other expenditure incurred in the traditional framework, such as purchase, deployment and maintenance, are accounted for the service providers or the sensor owners.

From these figures, we also find that the total cost for each WSN remains nearly same for the sensor-cloud framework. However, for the WSN based traditional framework, various types of costs results in varying total cost for each WSNs. Also, from Figure 9(a), it is evident that for individual end-users, the incurred cost will be lower in sensor-cloud framework than

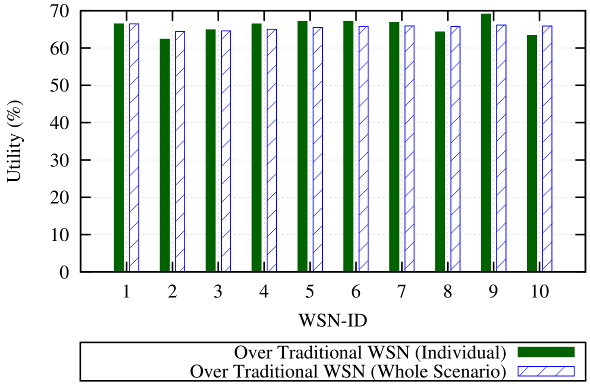
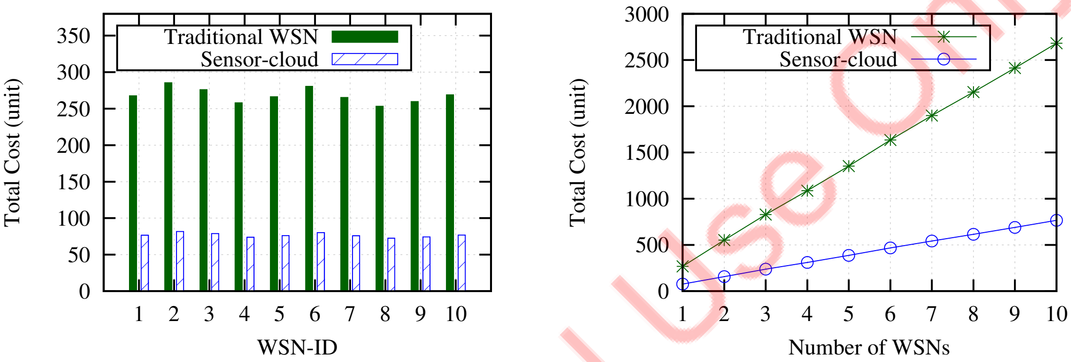


Fig. 8: Utility of the on-field WSNs



(a) Cost of the On-field WSNs

(b) Total cost of the on-field WSNs

Fig. 9: Cost incurred for end-users

that of the traditional WSNs.

VIII. Conclusion

In this paper, we analyzed the benefits of using a sensor-cloud framework for efficient management of various agricultural applications. We discussed the specific challenges associated with designing a sensor-cloud system for agricultural applications. We mathematically devised the virtualization technique underlying the proposed sensor-cloud framework by considering specific challenges. Consequently, the energy optimization framework for sensor-cloud is presented and the duty scheduling to conserve energy in this framework is discussed. The existing works for sensor-cloud computing for agriculture does not specifically define the specific components associated with it. We present case studies of different applications as an example of the framework. Finally, with simulation-based results, we show the justification for choosing the framework for agricultural applications.

In the future, we plan to extend the proposed framework for various other agricultural applications. Consequently, features related to mobility-aware dynamic service management are also required to be added in the framework.

Acknowledgement

This work is supported by Information Technology Research Academy (ITRA), Government of India under ITRA-Water grant ITRA/15(69)/WATER/M2M/01.

References

[1] T. Ojha, S. Misra, and N. S. Raghuwanshi, “Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges,” *Computers and Electronics in Agriculture*, vol. 118, pp. 66–84, October 2015.

[2] A. ur Rehman, A. Z. Abbasi, N. Islam, and Z. A. Shaikh, “A review of wireless sensors and networks’ applications in agriculture,” *Computer Standards & Interfaces*, vol. 36, no. 2, pp. 263–270, 2014.

[3] J. M. Barcelo-Ordinas, J. P. Chanet, K. M. Hou, and J. García-Vidal, “A survey of wireless sensor technologies applied to precision agriculture,” in *Precision agriculture’13*, J. Stafford, Ed. Wageningen Academic Publishers, 2013, pp. 801–808.

[4] J. Burrell, T. Brooke, and R. Beckwith, “Vineyard computing: sensor networks in agricultural production,” *IEEE Pervasive Computing*, vol. 3, no. 1, pp. 38–45, 2004.

[5] S. A. Nikolidakis, D. Kandris, D. D. Vergados, and C. Douligeris, “Energy efficient automated control of irrigation in agriculture by using wireless sensor networks,” *Computers and Electronics in Agriculture*, vol. 113, pp. 154–163, April 2015.

[6] E. Lichtenberg, J. Majsztrik, and M. Saavoss, “Grower demand for sensor-controlled irrigation,” *Water Resources Research*, vol. 51, 2015, [DOI: 10.1002/2014WR015807].

[7] A. Reche, S. Sendra, J. R. Díaz, and J. Lloret, “A smart M2M deployment to control the agriculture irrigation,” in *Proceedings of Ad-hoc Networks and Wireless*, Benidorm, Spain, 2015, pp. 139–151, lNCS 8629.

[8] D. J. Greenwood, K. Zhang, H. W. Hilton, and A. J. Thompson, “Opportunities for improving irrigation efficiency with quantitative models, soil water sensors and wireless technology,” *Journal of Agricultural Science*, vol. 148, pp. 1–16, 2010.

[9] J. Gutiérrez, J. F. Villa-Medina, A. Nieto-Garibay, and M. Ángel Porta-Gándara, “Automated irrigation system using a wireless sensor network and GPRS module,” *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 1, pp. 166–176, 2014.

[10] M. Moghaddam, D. Entekhabi, Y. Goykhman, K. Li, M. Liu, A. Mahajan, A. Nayyar, D. Shuman, and D. Teneketzis, “A wireless soil moisture smart sensor web using physics-based optimal control: Concept and initial demonstrations,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 3, no. 4, pp. 522–535, 2010.

[11] N. Chen, X. Zhang, and C. Wang, “Integrated open geospatial web service enabled cyber-physical information infrastructure for precision agriculture monitoring,” *Computers and Electronics in Agriculture*, vol. 111, pp. 78–91, 2015.

[12] J. A. López, A.-J. Garcia-Sanchez, F. Soto, A. Iborra, F. Garcia-Sanchez, and J. Garcia-Haro, “Design and validation of a wireless sensor network architecture for precision horticulture applications,” *Precision Agriculture*, vol. 12, no. 2, pp. 280–295, 2011.

[13] S. E. Díaz, J. C. Pérez, A. C. Mateos, M.-C. Marinescu, and B. B. Guerra, “A novel methodology for the monitoring of the agricultural production process based on wireless sensor networks,” *Computers and Electronics in Agriculture*, vol. 76, no. 2, pp. 252–265, 2011.

[14] D.-H. Park, B.-J. Kang, K.-R. Cho, C.-S. Shin, S.-E. Cho, J.-W. Park, and W.-M. Yang, “A study on greenhouse automatic control system based on wireless sensor network,” *Wireless Personal Communications*, vol. 56, no. 1, pp. 117–130, 2011.

[15] J. A. L. Riquelme, F. Soto, J. Suardíaz, P. Sánchez, A. Iborra, and J. A. Vera, “Wireless sensor networks for precision horticulture in southern spain,” *Computers and Electronics in Agriculture*, vol. 68, no. 1, pp. 25–35, 2009.

[16] P. Corke, T. Wark, R. Jurdak, W. Hu, P. Valencia, and D. Moore, “Environmental wireless sensor networks,” *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1903–1917, 2010.

[17] A. S. Voulodimos, C. Z. Patrikakis, A. B. Sideridis, V. A. Ntafis, and E. M. Xylouri, “A complete farm management system based on animal identification using RFID technology,” *Computers and Electronics in Agriculture*, vol. 70, no. 2, pp. 380–388, 2010.

[18] X. Mao, X. Miao, Y. He, X.-Y. Li, and Y. Liu, “CitySee: Urban CO2 monitoring with sensors,” in *Proceedings of IEEE INFOCOM*, Orlando, FL, USA, 2012, pp. 1611–1619.

[19] H. Yang, Y. Qin, G. Feng, and H. Ci, “Online monitoring of geological CO2 storage and leakage based on wireless sensor networks,” *IEEE Sensors Journal*, vol. 13, no. 2, pp. 556–562, 2013.

[20] A. Malaver, N. Motta, P. Corke, and F. Gonzalez, “Development and integration of a solar powered unmanned aerial vehicle and a wireless sensor network to monitor greenhouse gases,” *Sensors*, vol. 15, no. 2, pp. 4072–4096, 2015.

[21] X. Dong, M. C. Vuran, and S. Irmak, “Autonomous precision agriculture through integration of wireless underground sensor networks with center pivot irrigation systems,” *Ad Hoc Networks*, vol. 11, no. 7, pp. 1975–1987, 2013.

[22] J. Hwang, C. Shin, and H. Yoe, “A wireless sensor network-based ubiquitous paprika growth management system,” *Sensors*, vol. 10, pp. 11 566–11 589, 2010.

[23] S. K. Roy, A. Roy, S. Misra, and N. S. Raghuwanshi, “AID: A prototype for agricultural intrusion detection using wireless sensor network,” in *Proceedings of IEEE International Conference on Communications*, London, UK, 2015, pp. 7059–7064.

[24] A.-J. Garcia-Sanchez, F. Garcia-Sanchez, and J. Garcia-Haro, “Wireless sensor network deployment for integrating video-surveillance and data-monitoring in precision agriculture over distributed crops,” *Computers and Electronics in Agriculture*, vol. 75, no. 2, pp. 288–303, 2011.

[25] M. Yuriyama and T. Kushida, “Sensor-cloud Infrastructure – Physical Sensor Management with Virtualized Sensors on Cloud Computing,” in *Proceedings of IEEE International Conference on Network-Based Information Systems*, Takayama, Japan, 2010, pp. 1–8.

[26] A. Alamri, W. S. Ansari, M. M. Hassan, M. S. Hossain, A. Alelaiwi, and M. A. Hossain, “A survey on sensor-cloud: Architecture, applications, and approaches,” *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1–18, 2013.

[27] P. Evensen and H. Meling, “SenseWrap: A service oriented middleware with sensor virtualization and self-configuration,” in *Proceedings of International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, Melbourne, VIC, Australia, 2009, pp. 261–266.

[28] J. Ibbotson, C. Gibson, J. Wright, P. Waggett, P. Zerfos, B. Szymanski, and D. J. Thornley, “Sensors as a service oriented architecture: Middleware for sensor networks,” in *Proceedings of International Conference on Intelligent Environments*, Kuala Lumpur, 2010, pp. 209–214.

[29] S. Misra, S. Chatterjee, and M. S. Obaidat, “On theoretical modeling of sensor-cloud: A paradigm shift from wireless sensor network,” *IEEE Systems Journal*, pp. 1–10, 2014 [DOI: 10.1109/JSYST.2014.2362617].

[30] S. Madria, V. Kumar, and R. Dalvi, “Sensor Cloud: A cloud of virtual sensors,” *IEEE Software*, vol. 31, no. 2, pp. 70–77, 2014.

[31] K. Kim, S. Lee, H. Yoo, and D. Kim, “Agriculture sensor-cloud infrastructure and routing protocol in the physical sensor network layer,” *International Journal of Distributed Sensor Networks*, vol. 2014, pp. 1–13, 2014.

[32] C. Krintz, R. Wolski, N. Golubovic, B. L. V. Kulkarni, B. Sethuramasamyraja, B. Roberts, and B. Liu, “SmartFarm: Improving agriculture sustainability using modern information technology,” in *Proceeding of ACM SIGKDD Workshop on Data Science for Food, Energy and Water*, San Francisco, USA, August 2016.

[33] Y.-H. Liu, K.-L. Ong, and A. Goscinski, *Networked Digital Technologies: 4th International Conference, NDT 2012, Dubai, UAE, April 24-26, 2012, Proceedings, Part II.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, ch. Sensor-Cloud Computing: Novel Applications and Research Problems, pp. 475–486.

[34] A. Goscinski, I. Janciak, Y. Han, and P. Brezany, *Grid and Cloud Database Management*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, ch. The CloudMiner - Moving Data Mining into Computational Clouds, pp. 193–214.

[35] T. Ojha, S. Bera, S. Misra, and N. S. Raghuwanshi, “Dynamic duty scheduling for green sensor-cloud applications,” in *Proceedings of IEEE CloudCom*, Singapore, Dec 2014, pp. 841–846.

[36] S. Misra, S. Das, M. Khatua, and M. S. Obaidat, “QoS-guaranteed bandwidth shifting and redistribution in mobile cloud environment,” *IEEE Trans. on Cloud Computing*, vol. 2, no. 2, pp. 181–193, 2014.

[37] S. Misra, A. Singh, S. Chatterjee, and A. K. Mandal, “QoS-aware sensor allocation for target tracking in sensor-cloud,” *Ad Hoc Networks*, vol. 33, pp. 140–153, 2015.

[38] S. Chatterjee and S. Misra, “Optimal composition of a virtual sensor for efficient virtualization within sensor-cloud,” in *Proceedings of IEEE International Conference on Communication*, London, UK, 2015, pp. 448–453.

[39] S. Chatterjee and S. Misra, “Dynamic and adaptive data caching mechanism for virtualization within sensor-cloud,” in *Proceedings of IEEE ANTS*, New Delhi, India, 2014, pp. 1–6.

[40] S. Chatterjee, S. Misra, and S. U. Khan, “Optimal data center scheduling for Quality of Service management in sensor-cloud,” *IEEE Transactions on Cloud Computing*, 2015 [DOI:10.1109/TCC.2015.2487973].

[41] S. Chatterjee, R. Ladia, and S. Misra, “Dynamic optimal pricing for heterogeneous service-oriented architecture of sensor-cloud infrastructure,” *IEEE Transactions on Services Computing*, 2015 [DOI: 10.1109/TSC.2015.2453958].

[42] C. Zhu, H. Wang, X. Liu, L. Shu, L. T. Yang, and V. C. M. Leung, “A novel sensory data processing framework to integrate sensor networks with mobile cloud,” *IEEE Systems Journal*, pp. 1–12, January 2014 [DOI: 10.1109/JSYST.2014.2300535].

[43] C. Zhu, Z. Sheng, V. C. M. Leung, L. Shu, and L. T. Yang, "Towards offering more useful data reliably to mobile cloud from wireless sensor network," *IEEE Transactions on Emerging Topics in Computing*, vol. 3, no. 1, pp. 84–94, 2015.

[44] S. Distefano, G. Merlino, and A. Puliafito, "A utility paradigm for IoT: The sensing cloud," *Pervasive and Mobile Computing*, vol. 20, pp. 127–144, 2015.

[45] N. Sales, O. Remédios, and A. Arsenio, "Wireless sensor and actuator system for smart irrigation on the cloud," in *Proceedings of IEEE 2nd World Forum on Internet of Things (WF-IoT)*, Milan, Italy, 2015, pp. 693–698.

[46] V. Bapat, P. Kale, V. Shinde, N. Deshpande, and A. Shaligramd, "WSN application for crop protection to divert animal intrusions in the agricultural land," *Computers and Electronics in Agriculture*, vol. 133, pp. 88–96, February 2017.

[47] A. Matese, S. F. D. Gennaro, A. Zaldei, L. Genesio, and F. P. Vaccari, "A wireless sensor network for precision viticulture: The NAV system," *Computers and Electronics in Agriculture*, vol. 69, no. 1, pp. 51–58, 2009.

[48] C. C. Baseca, J. R. Díaz, and J. Lloret, "Communication Ad Hoc protocol for intelligent video sensing using AR drones," in *IEEE Mobile Ad-hoc and Sensor Networks (MSN)*, 2013, pp. 449–453.