ISOF: Information Scheduling and Optimization Framework for Improving the Performance of Agriculture Systems Aided by Industry 4.0

Gunasekaran Manogaran, *Member, IEEE*, Ching-Hsien Hsu, *Senior Member, IEEE*, Bharat S. Rawal, *Senior Member, IEEE*, BalaAnand Muthu, Constandinos X. Mavromoustakis, *Senior Member, IEEE*, and George Mastorakis, *Member, IEEE*

***Abstract*—Industry 4.0 is a promising evolution in the field of smart farming by improving the productivity and reducing human intervention to modernize agriculture. This smart paradigm incorporates different levels of the automation from cropping to production yield through sophisticated techniques. Different intelligent computing techniques and communication technologies are augmented with the industry paradigm for improving the efficiency of agriculture systems. This letter introduces information scheduling and optimization framework (ISOF) for optimizing the communication and information layer process in industry 4.0 architecture. Information scheduling and classification of agriculture information are optimized through this framework for reducing process latency and stagnancy. The control flexibility of a smart farm is determined using the latency and stagnancy at the end of yields. The classification part segregates information based on processing and completion time to reduce backlogs through offloading process. The advantage of this framework is that it inherits the advantages of Internet of Things (IoT) and edge computing (EC) technologies with interoperable feature to aid information processing, information classification, offloading, and periodic updates. The performance of the proposed framework is tested in a corn farm and some common metrics, such as delayed information, processing time, audit data, and information distribution, are analyzed for proving the reliability of the framework.**

***Index Terms*—Agriculture systems, industry 4.0, information classification, process offloading, smart farming.**

Manuscript received May 5, 2020; revised August 28, 2020 and November 23, 2020; accepted November 30, 2020. Date of publication December 17, 2020; date of current version February 19, 2021. This work was partially supported by the National Natural Science Foundation of China under Grant 61872084. *(Corresponding author: Ching-Hsien Hsu.)*

Gunasekaran Manogaran is with the Department of Computer and Information Science, University of California, Davis, Sacramento, CA 95833 USA, and also with the College of Information and Electrical Engineering, Asia University, Taichung 41354, Taiwan (e-mail: gmanogaran@ieee.org).

Ching-Hsien Hsu is with the School of Mathematics and Big Data, Foshan University, Foshan 528000, China, and with the Department of Computer Science and Information Engineering, Asia University, Taichung 41354, Taiwan, and also with the Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 406040, Taiwan (e-mail: robertchh@asia.edu.tw).

Bharat S. Rawal is with the Cybersecurity and Cyber Engineering Program, Gannon University, Erie, PA 16541 USA (e-mail: rawalksh001@gannon.edu).

BalaAnand Muthu is with the Department of Computer Science and Engineering, V.R.S. College of Engineering and Technology, Villupuram 605602, India (e-mail: balaanand@ieee.org).

Constandinos X. Mavromoustakis is with the Department of Computer Science, Mobile Systems Laboratory, University of Nicosia Research Foundation, University of Nicosia, Nicosia 1700, Cyprus (e-mail: mavromoustakis.c@unic.ac.cy).

George Mastorakis is with the Department of Management Science and Technology, Hellenic Mediterranean University, 72100 Heraklion, Greece (e-mail: gmastorakis@hmu.gr).

Digital Object Identifier 10.1109/JIOT.2020.3045479

I. Introduction

INDUSTRY 4.0 is a modern phase of the industrial revolution that hires interoperable technology and smart machines for automation, unified communication, learning, human interaction, and real-time data processing. Industry 4.0 is a sophisticated version of Industrial Internet of Things (IIoT) that integrates the physical real-world elements, smart objects, human, digital communication and wireless technologies for granting smart production, and management systems for organizations [1]. This paradigm acts as an interconnection of multiple levels of business scaling production and supply chain management in different aspects. The vision of industry 4.0 is to integrate manufacturers, products, people, and audits in a linear manner with less complexity, exploiting technological benefits of automation. Developing self-learning, intelligent and interactive systems are considered as a substitute for human intelligence and delay less processing in these environments. More specifically, this paradigm exploits the "intelligence" of the machines controlled through appropriate decision systems for communication and productive operations. Aided by the advantage of smart machines, and intelligence and communication technologies, industry 4.0 grants fully integrated and automated production feasibility with human–machine interaction. This integration transcends as the fundamental building block of the industrial automation replacing supplier, organization, and customer relationships [2], [3].

Smart agricultural and farming systems are generated in the recent decade to improve the reliability of crop production management in smart cities. The application of smart farming reduces human intervention and preserves round-the-clock work. The intelligent communication and sensing capabilities of the farming systems assist in pervasive monitoring and management of crops from planting to yielding period [4]. The evolution of industry 4.0, Internet of Things (IoT), and other communication models has led to the atomization of agricultural farms in a collaborative and intelligent manner. The work schedule of human farmers is replaced with sensors, communicating devices, control units, and smart machines. The process of planting, sowing, reaping, crop trimming, irrigation, weed cropping, and cultivation in accordance with the climatic and green house effects is determined by the smart features of the devices deployed. The smart farm environment is a collection of agricultural lands with communication technology and information processing systems to aid a ubiquitous operation environment [5], [7]. Janc *et al.* [6] explained that information and communication technologies (ICTs) act as a connecting point between farms and humans to improve the precision of agricultural practices. The reliability of ICT depends on the rationalized decision making, and qualitative and quantitative analysis of farm information to achieve better farming experience. Augmenting livestock management to cultivation, ICT is employed in all the aspects of the smart farming environment.

The smart farming environment is internally packed with software and hardware agents that coordinate the overall functions of humans. The software systems are responsible for decision making by gathering information from the environment and classifying it according to the needs. Planning, distribution, computation, optimization, and management solutions are provided by the software agents using different artificial intelligent techniques and other communication platforms such as clouds [7]. As the connected farming systems aid interoperability of heterogeneous devices, the information sharing and processing are the basic requirements for controlling the operations of the farm. For example, the need for irrigation is determined on the basis of the information shared by the temperature and humidity sensors deployed in the farm. The information transmitted from the sensors and other devices from the farm is converted into useful data for processing [8]. The connected system delivers output through actuators to control the functions of the farming machines. By exploiting wired and wireless communication technologies, information is gathered from and shared to the processing systems. The processing systems operate on the input data for delivering optimal actuation commands in coherence to the climatic and crop conditions. Therefore, information processing is the major concern in improving the precision of smart farming [9], [10]. The contributions of this letter are summarized as follows.

1) Designing an information scheduling and optimization framework (ISOF) for improving the performance of smart farm processing flexibility.
2) Designing learning-dependent processing for reducing stagnancy in information handling and improving the distribution.
3) Designing an attribute classified offloading to confine processing faults with an augmenting audit information classification.
4) Performing self and outward comparative analysis to measure the reliability of ISOF using different metrics.

Ayaz *et al.* [11] discussed the smart agriculture system based on IoT paradigm along with the applications and devices used. The authors described the communication standards, operations in the farm, services, and devices that are required for the smart farming. The overall discussion provides an introduction to IoT applications in improving the productivity of the farms. In [12], climate-smart agriculture (CSA) is discussed. The process of prioritizing farms on the basis of investment scale and related conditions for emphasizing them is briefly discussed in this reference. In the works presented in [13]–[15], service processing in smart farm and precision agriculture is elaborated to improve the reliability of the smart environment. Colezea *et al.* [13] proposed an integrated Web-based communication environment to leverage the productivity of the farms. With the help of a cloud network (CN), flexibility and adaptability of the smart farming systems are improved. In the work presented in [14], the service management for smart agriculture systems is discussed. Data gathering, processing, and statistics management are provided as services by integrating cloud platform. Energy-aware sensor-based precision agriculture is proposed in [15] for improving the functions of the sensing devices deployed in the farming region. In this method, Kalman filter is used to reduce the energy expenses of the sensing devices to improve the lifetime and quality of information sensing.

Similar to the work presented in [11], Ahmed *et al.* [16] introduced IoT for improving the performance of precision agriculture. Different from the proposal in [11], edge and fog computing paradigms are associated in this IoT-based farming method for improving the cross-layer efficiency along with latency control and increasing the coverage area of monitoring.

A hybrid computing framework is designed in [17] for improving the performance of smart manufacturing devices employing edge computing (EC) paradigms. This computing framework consists of four-layers for scheduling and task processing. This employs artificial intelligence (AI) techniques for greedy and threshold-based scheduling of available resources to achieve high degree of user satisfaction and energy efficiency.

The works presented in [18] and [19] exploit machine learning techniques for prediction and classification of events in agriculture applications. Synthetic minority oversampling technique (SMOTE) [18] is designed for predicting frost by validating the meteorological data from different locations. The information is selected, classified, and then trained for concluding the prediction using regression-based learning. Liu *et al.* [19] assimilated the convolutional neural network (CNN) and channel-spatial attention for detecting and classifying pests in the farming lands. Based on the region proposal network (RPN) and image maps, the location of the pests is identified using the trained CNN.

Another proposal of IoT along with edge and fog computing paradigms is presented in [20]. Different from the proposals in [11] and [16], the proposal introduced in [20] incorporates deep reinforcement learning and cloud computing for smart decision making. The irrigation requirements of the farm are periodically decided using the learning process. Aiken *et al.* [21] presented a big-data analytical model for analyzing the performance of different farms. The performance of the farms exploiting machine learning with respect to the deterministic and stochastic computation models is analyzed and discussed in this reference.

The application of machine learning is extended to predict solar radiation in the [22] by estimating suitable empirical models. This proposal concludes that hybrid evolutionary algorithm and neural network processing ensures better data validation and results.

A crop yield forecasting model is proposed by Filippi *et al.* [23] for improving the precision of decision making in smart farming. This model is based on machine learning approach that intakes multiple data sets from different forms in a layered manner.

Irrigation recommendation designed using machine learning is introduced in [24]. Crop monitoring, controlling, and irrigation requirement prediction are jointly devised in this proposal for leveraging the performance of smart farms. Data analysis and prediction model is found to improve the classification and accuracy of the data sets in correlation with the real-time scenario. Han *et al.* [25] employed unmanned aerial vehicles (UAVs) and machine learning for modeling field investigation. In this investigation, the above-ground biomass (AGB) of the maize is monitored to validate the crop growth. Using a conventional feature elimination model, support vector machine and random forest, the crop information is assessed for processing.

The information processing methods presented in [14], [15], and [17] lack stagnancy mitigation as the new information replaces the existing audits. This results in improper classification of field information due to the growing information density. On the other hand, the forecasting model in [23] is subjected to local optima problem due to fewer validations. The method presented in [18] and [19] results in high complexity due to sequential processing and training data accumulation. Considering the impact of the adverse effects over information processing, the proposed framework is designed to maximize the rate of information to reduce stagnancy.

II. Information Scheduling and Optimization Framework for Improving Process Flexibility

ISOF is designed to improve the process flexibility and swiftness in the agriculture-based industry 4.0 architecture. It operates between the information and communication layers for improving the performance of the architecture. This framework inherits the benefits of IoT and EC paradigms for addressing the issues in delayed processing and response. Therefore, the framework is modeled to achieve three kinds of operations listed as follows.

1) Learning-driven processing for reducing information stagnancy.
2) Attribute-based offloading for improving the swiftness on information processing and thereby reducing faults.
3) Classification of the processed information for improving audit rate in the organizational level of industry 4.0.

The first operation is solely achieved by rationalizing the behavior of the sowing machines in an intelligent manner. The second is facilitated by incorporating IoT and edge paradigms and the third operation is the collaborative result of the previous processes. The general architecture of the proposed framework in for a smart farm is illustrated in Fig. 1.

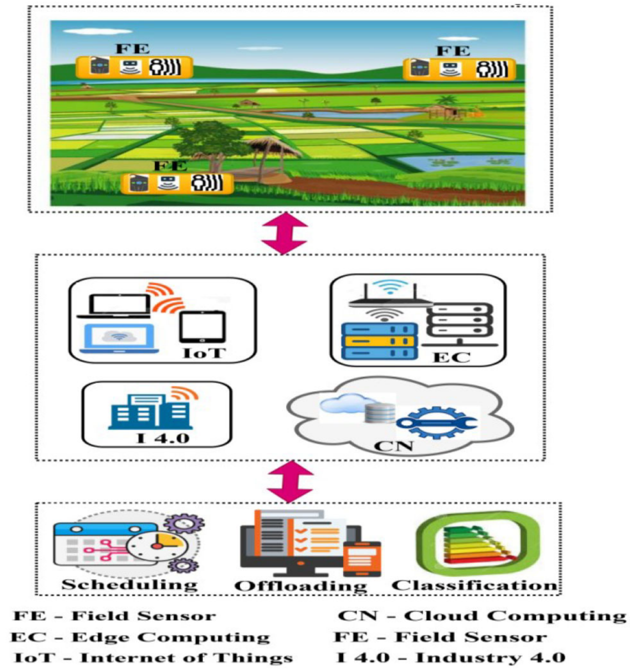

Fig. 1. Architecture of industry framework with ISOF.

In Fig. 1, a model of smart farm assimilating IoT, EC, and CN along with industry 4.0 architecture is presented. The field sensors (FEs) are responsible for gathering the information from the land and transmit to the processing platform of cloud. The communication and processing technologies connected to the smart farms are responsible for information scheduling, offloading, and classification. This is required for environment awareness and to meet the requirements of the farm. The following section describes the operations of the framework independently.

A. Learning-Driven Processing

This operative part of ISOF is responsible for reducing the stagnancy in processing allocated information. Let a fertilizer deploying unit of a smart farm be equipped with M sowing machines. The machines are administered using a control unit. The control unit assigns processing information for each machine and is responsible for "start" and "stop" of the machine. The productions (t_p) of the machines are defined in which N numbers of information are allocated to each machine. Control unit assigns information in a sequential manner, i.e., a new information is queues with the awareness of the completion time (t_c) of the current information such that $\sum_{i=1}^{N} t_{ci} = t_p$. The sum of all the completion time of the allocated information is the production time of the machine. As all the machines are operated using a centralized control unit, information exceeding t_p is considered as failure/stagnant. High is the number of failed/stagnant information, less is the performance of the machines and hence the production.

Therefore, the fundamental stage in optimizing information processing is to ensure $t_{cN} \not> t_p$. To ensure the validity of the above condition, a learning assisted information scheduling is imposed in this operational part. t_c of the machines vary with the time and length of the information. Let $\rho_a = 1$ indicate the chances of a machine being available for accepting a successive N. Now, the processing dynamicity (Δp) with time t_p is estimated using

$$\Delta p = \sum_{i=1}^{N} \rho_{ai}.\rho_{si}.\rho_{Mi} \tag{1}$$

where ρ_{si} is the start time instance for accepting the ith information and ρ_{Mi} is the chance of assigning the ith information to the Mth machine. From (1), the processing dynamicity relies on the operating time period of the machine. In a conventional scheduling process, the N information is assigned in the increasing order of Δp for the first-come–first-serve information. The information are assigned sequentially and there is always a lag between t_c and t_s due to information assignment where $(t_{ci} - t_{si+1})$ is a time constraint. This time constraint is to be suppressed to achieve nonstagnant information scheduling. Here, different from the conventional scheduling method, the machine availability time is accounted for processing the incoming information. Based on the availability time, uniform processing is achieved by distributing the consecutive set of N over the machines with less $(t_{ci} - t_{si+1})$ time. If U_M represents the uniform processing time of M machines, then

$$U_M = \sum_{i=x}^{N} \frac{1}{(N-i)} \left(t_{ci} - t_{si} * t_p\right)^2. \tag{2}$$

Here, x represents the information that is fed in the second order of any available machine. Similarly, U_M holds only if $\rho_a = 1$ else the uniform processing count decreases. Considering the uniformity requirement of the machines to handle (x to N) information, the objective is defined as per

$$\min[U_M(x, \rho_a, \rho_M)] = \left\| \left(U_M^{t_{s1}}, U_M^{t_{s2}}, U_M^{t_{s3}}, \ldots, U_M^{t_{sN}}\right)^{t_p} \right\|. \tag{3}$$

In (3), the minimum requirement for processing the information (x to N) is given if $\rho_a = 1$ and $\rho_M \to 1$, i.e., machine is available and any $i \in [x, N]$ is assigned to the xth machine. Equation (3) is satisfied by continuous $(t_{ci} - t_{si+1})$ monitoring to maximize Δp. If Δp is maximized for a considerable set of M, then U_M for which continuous-time monitoring is aided by the recurrent learning process. A recurrent learning process computes the reward (R_M) of a machine. The reward of a machine relies on the stagnant process and completed information along t_p through t_c. Equation (4) is used to compute the reward of the machine as

$$R_M = \sum_{i=1}^{N} \frac{n_{l_i}}{c_M} + (1-\rho_M)\left[R_M(i-1) + \sum_{i=1}^{N} \rho_M \frac{n_{l_i}}{c_M}\right] - \frac{R}{N} \tag{4}$$

where c_M is the processing speed of the machine M. Here, the current reward of a machine is estimated on the basis of its previous reward $R_M(i-1)$. The value of K is the stagnant information observed at the end of cumulative t_c. In the above equation, the values of $(1-\rho_M)[R_M(i-1)+\sum_{i=1}^{N} \rho_M(n_{l_i}/c_M)]$ and K are eventually monitored as an outcome of the learning process. Let o be the output of the learning process with an intermediate solution that is represented in Fig. 2(a).

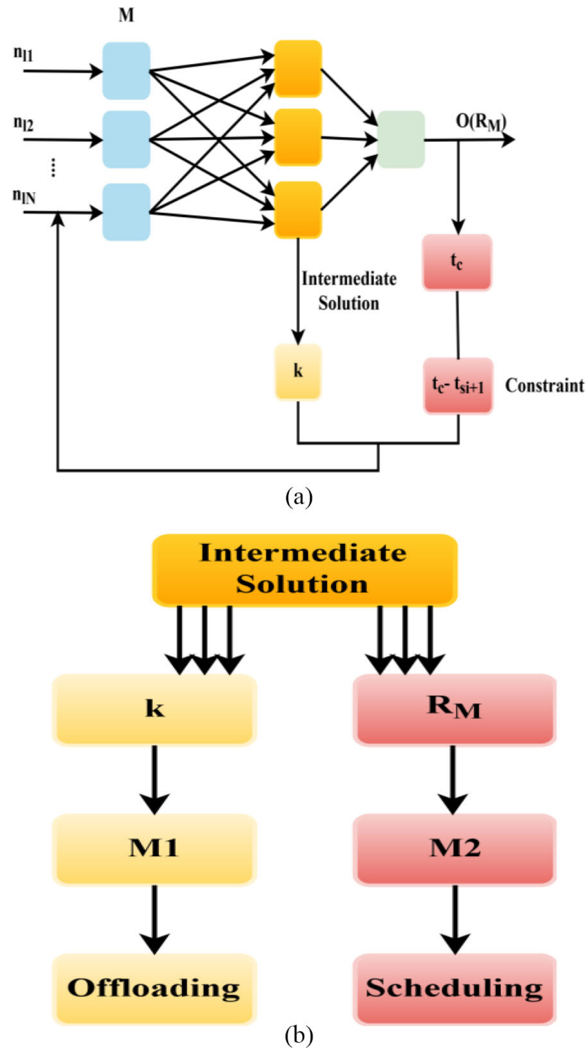

Fig. 2. Learning process. (a) Intermediate solution of k. (b) Intermediate solution analysis.

In Fig. 2(a), the conventional learning process with an intermediate selection of k is represented. The series of R_M is given by (5) for all the intermediate solutions

$$\left.\begin{aligned} R_{M1} &= \tfrac{n_{l_1}}{c_M} \\ R_{M2} &= \textstyle\sum_{i=1}^{2} \tfrac{n_{l_i}}{c_M} + (1-\rho_M)\left[R_{M1} + \sum_{i=1}^{2} \rho_M \tfrac{n_{l_i}}{c_M}\right] - \tfrac{k}{2} \\ &\vdots \\ R_{MN} &= \textstyle\sum_{i=1}^{2} \tfrac{n_{l_i}}{c_M} + (1-\rho_M)\left[R_{MN-1} + \sum_{i=1}^{N} \rho_M \tfrac{n_{l_i}}{c_M}\right] - \tfrac{k}{2} \end{aligned}\right\}. \tag{5}$$

Here, k is incremented if the condition $t_{cN} \not> t_p$ fails or $(t_{c_i} - t_{si+1})$ increases for each R_M estimated. In both the cases, k increases that refers to the need for offloading the information. In the learning process [Fig. 2(a)], the intermediate solution results in classifying k and $(t_c - t_{si+1})$. Both the factors are

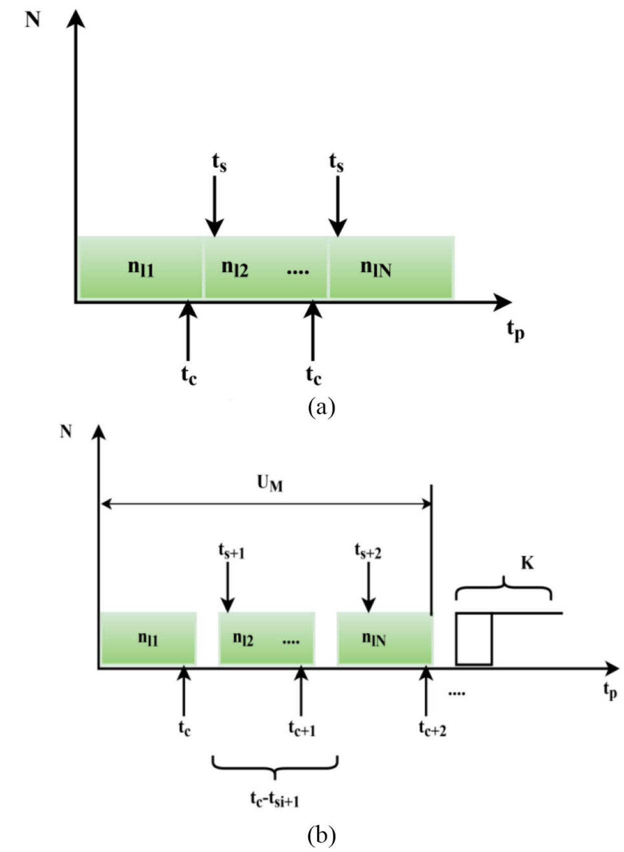

Fig. 3. Scheduling process. (a) Conventional scheduling. (b) U_M scheduling.

TABLE I
MONITORED VALUE OF THE CLASSIFIED INFORMATION FOR DIFFERENT ITERATIONS

Iteration	U_M	k Information	$(t_{c_i} - t_{si+1})$ information
100	94	12	4
200	80	17	21
300	65	6	19
400	73	2	0
500	62	23	9
600	41	14	23
700	56	12	15
800	60	23	3
900	45	27	24
1000	74	0	18

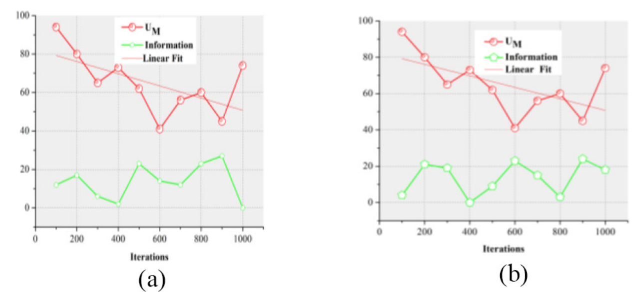

Fig. 4. (a) Classified k information ratio. (b) Classified $(t_{c_i} - t_{si+1})$ information ratio.

used for rationalizing n_l in the concurrent execution phases. In Fig. 2(b), $M1$ and $M2$ represent the machines that requires offloading and that proceeds with scheduling correspondingly. In the intermediate solution, machines that are categorized under k are discarded for R_M estimation. For the next set of U_M, the machines satisfying the above-mentioned conditions are grouped to handle n_l. Now, the scheduling process is modified from conventional to rationalized methods as illustrated in Fig. 3(a) and (b), respectively.

In a conventional/ideal scheduling process, t_c of the current information and t_s of the consecutive information are the same [Fig. 3(a)]. In the practical case, this is not feasible as ρ_a and ρ_M chances have to be true in processing information. Therefore, the information entering the execution phase in $(t_c - t_{si+1})$ needs to be scheduled. Similarly, the information that exceed t_p are classified under k [Fig. 3(b)] are overloaded to prevent stagnancy. As per the learning process, both scheduling and overloading are balanced by the control unit and edge devices concurrently to retain the information delivery rate. In Table I, the number of recurrent iteration, the number of U_M machines, and their corresponding information under k and$(t_{c_i} - t_{si+1})$ are tabulated.

Fig. 4(a) and (b) portrays the classified information under k and $(t_{c_i} - t_{si+1})$ under the unified operating time of the machines. The number of unified machines under the same time varies with the learning iterate and hence the classified information. The linear line in the graphs represents the variation of different unified machine time where the operating time decreases linearly.

B. Attributes-Based Offloading

As mentioned in the previous section, the identified information under k are offloading to the available M to retain the rate of production. In the offloading phase, the delayed process is rescheduled by assigning the information to another M in order to reduce the delay. High is the number of information completed, less is the production in terms of time and information count. The delay in processing $d(s, t_s)$ is given by

$$\left.\begin{array}{r} d(n_l, t_c) = t_c(n_l) - t_{sc} \text{ and} \\ \max(n_l, t_c) = \max_{i \in N}\{d(n_l,\ t_c)\} \end{array}\right\}. \quad (6)$$

The above equation represents the excepted delay for processing an information of length n_l whose actual scheduled completion time in t_{sc}. Therefore, if any n_l achieves a $\max(n_l, t_c)$ is preestimated in the intermediate solution level of the learning process and is rescheduled. Information rescheduling/offloading increases the complexity of the control unit and hence, the process is pursued with the aid of edge nodes. The edge nodes are programmed with the classification of [k and $(t_c - t_{si+1})$] to encourage rescheduling and process update. The control unit moves on with the sequential U_M for processing N information. The rescheduling/offloading process is assisted with edge features using the machines that differ from U_M. The machines that do not satisfy U_M are independent of scheduling time and hence are controlled by the edge layer in industry 4.0. Let $RS = \{n_1, n_2, \ldots, n_j\}, n \in N$ and $j \notin n$ in U_M. The excepted time delay of the information in

RS is a determined by (6) as $(t_c - t_{si+1})$ high in the intermediate solution state. There are three cases to be analyzed for assigning the information to M that is available. The cases for rescheduling/ offloading are given as follows.

Case 1: If $t_c < t_{si+1} < t_{s+1}$, i.e., the start time of an information $\in RS$ lies between completion and start time of a machine in U_M.

Analysis 1: In this case, t_c and t_{s+1} are estimated for a machine in U_M and the difference between the two remains constant. The machines that are not classified under U_M are identified in the order of t_c. The information [with high $d(n_l, t_c)$] is classified in the increasing order of their n_l. Now, the information are concurrently/sequentially arranged by the edges nodes to the machines that are not satisfying U_M. The condition based scheduling as per case 1 is illustrated in Fig. 5(a).

Here, the information exceeding $(t_c - t_{si+1})$ or discarded as per k are offloading to the machines $\in U_M$. The information that fall in $(t_c - t_{si+1})$ category as per the conditions is sequentially fed to the U_M machines that retains Δp. If the start time of an information lies exactly between the completion and start time of the scheduled information, then the completion time of

$$t_c(n_l) = (\max\{a + (t_s + 1)\rho_M, (2 + t_{s+1})a\} + a) - \sum_{i=1}^{N-nl-k} \frac{t_c}{i}. \quad (7)$$

The time estimated in (6) lies between $(1 \times t_c)$ and $(N \times t_c)$ of all $N \in U_M$. Therefore, a minimum of $(t_p/2)$ and a maximum of $(t_p/(N - nl - k))$ rescheduling are preformed in case 1.

Case 2: $t_{si+1} = t_c$ or $t_{si+1} = t_{s+1}$, i.e., the start time of an information is equal to the completion start time of an information $\in U_M$.

Analysis 2: If the condition as in Fig. 5(b) is true, then t_{si+1} is augmented to the machine $\in U_M$. This helps to retain Δp of the machines that increases the rate of processing. Similarly, this is similar to the ideal case as illustrated in Fig. 3(a). Therefore, the delay is modeled as the difference between start and stop time of the machine. A machine experiences an improved n_l handling and is monitoring by the control unit. The edge devices reschedule the information to filter $t_c = t_{si+1}$ to move the logs of that information to the control unit. It also recommends a rescheduling process to the control unit to accommodate the filtered information in U_M.

Contrarily, if $t_{si+1} = t_{s+1}$ [Fig. 5(c)], then the information belongs to the governance of edge devices. It differentiated the operating time of the machines based on U_M. The coinciding information of length n_l is allocated. The delay for the process is estimated as the difference between two successive t_c. The start time is not considered as the information is classified before the time is augmented with processing delay. Therefore, both rescheduling and offloading are performed in this case. Here, the number of offloaded information is k. Similarly, the k information offloading to machines $\in U_M$ in the increasing order of $t_c - t_{si+1}$ mapping the information in the first-come–first-serve basis.

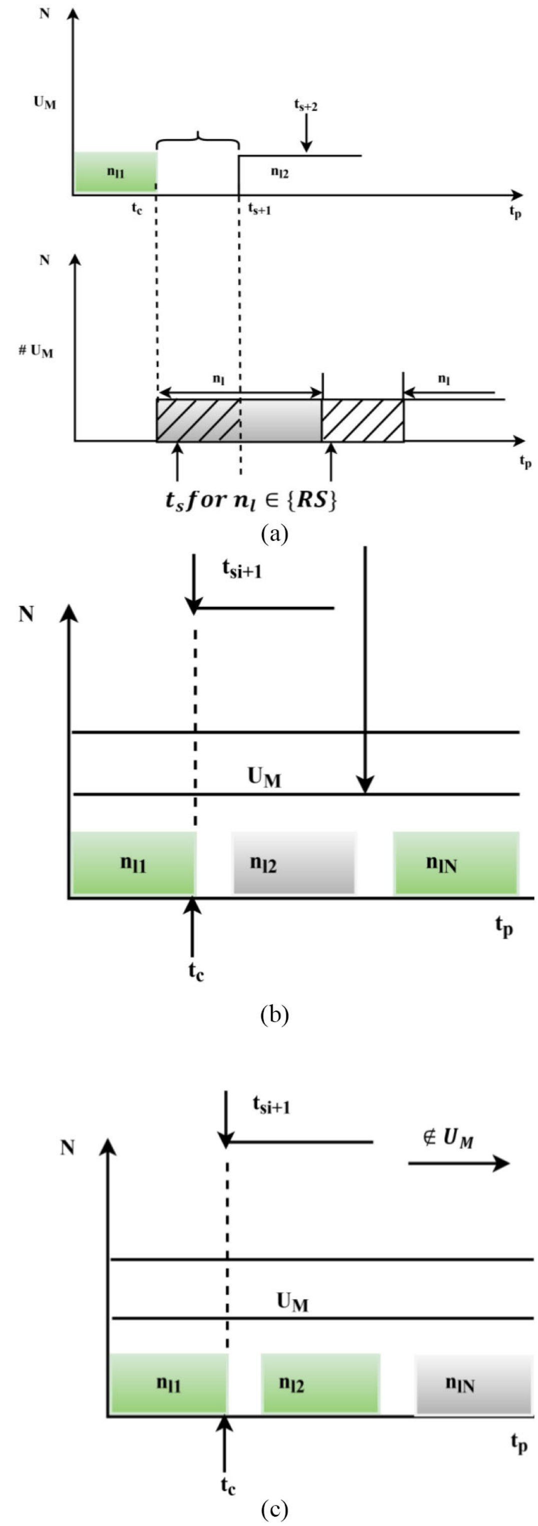

Fig. 5. (a) $t_c < t_{si+1} < t_{s+1}$ case scheduling. (b) $t_{si+1} = t_c$ case. (c) $t_{si+1} = t_{s+1}$ case.

Case 3: If the completion time of an information exceeds the production time, i.e., $t_c > t_p$.

Analysis 3: Consider the process illustrated in Fig. 3(b) where t_c of an information exceeds the production time. In this scenario, the number of information whose $t_c > t_p$ is categorized as k. In this scheduling/offloading process, the information that are identified as k before the last operation t_s are rescheduling/offloading as discussed in cases 1 and 2. The total cycles required to process N information is (t_p/N) and this varies with the discrepancy in $(t_c - t_{si+1})$ and n_l. The last but one slot $((t_p/N) - 1)$ is modeled to accept all the N that is classified under k. The edge devices perform a rescheduling operation over the k information that is not aided by U_M. Therefore, with the increase in t_c and completion of the information, the incoming/leftover information are reordered based on their length. Hence, the high length information is preferred first for processing. The mapping discussed in case 2 is reversed such that high length information is mapped by a machine with $\min(t_c < t_{si+1})$ time. Therefore, the information with less n_l are processed toward t_p, reducing the chances of t_c exceeding t_p. In this case, delay varies with the highest information to the lowest information length for which the difference between two successive t_c decrease with swift t_s. The offloaded k information are retained on the basis of their length to reduce processing fault in a given t_p.

C. Classification of Processed Information

The audit logs of the processed information and production information are stored in the distributed and ubiquitous communication layer of the industry 4.0 architecture. This is accessible through IoT supportive devices for remote access and processing rule formulation. The logs provide necessary information regarding the production and its efficiency. Hence, obtaining optimal data/information regarding the production becomes mandatory for improving efficiency. The business layer of industry 4.0 relies on the information provided from the lower layers to retain the audit reports are augmented based on relevance in t_p. The relevance between two successive useful audit log, $R(I)$ is computed as

$$R(I) = t_r \frac{N}{c_M} + t_u \frac{nl \times t_s}{\left(\frac{N}{c_M} - nlt_s\right)} - \Delta \tag{8}$$

where Δ represents the replication of the same log in different t_p, and t_r and t_u are the request and update time for the audit logs. The value of Δ is estimated as

$$\Delta = \left[\frac{t_p}{N} + \sum_{i=1}^{N} Ii(\rho_M.\rho_a)\right] \quad \forall N \in U_M \quad \text{and} \notin U_M. \tag{9}$$

The replication value Δ estimated in (9) is updated in all t_p/N for the available machine that has processed an information. The replication is identified for t_c of individual N and t_p. At the end of t_p, $R(I)$ is assessed for all I updated in t_c with the final t_p value. The received report is then summarized such that $(\Delta_{t_c} \cap \Delta_{t_p})$ is sorted as an individual list of audits. This differentiates $R(I)$ with respect to k and $t_c - t_{si+1}$ categorized information. Similarly, the replications, i.e., $(\Delta_{t_c} \cup \Delta_{t_p})$ are identified to map them into $R(I) \in k$ or $R(I) \in (t_c - t_{si+1})$ records. The time of update of the replicated value determines its belonging $R(I)$. The possible combinations and conditions for classifying the replications are tabulated in Table II.

TABLE II
CONDITIONS AND RELEVANCE SET

$\Delta_{t_c} \cup \Delta_{t_p}$	Condition	Relevance Set
Value	$t_c < t_{si+1}$	$R(I) \in t_c$
	$t_{si+1} < t_{s+1}$	$R(I) \in (t_c - t_{si+1})$
Value	$t_{si+1} = t_c$	$R(I) \in t_c$
	$t_{si+1} = t_{s+1}{}^*$	$R(I) \in k$
Value	$t_c > t_p{}^*$	$R(I) \in k$

TABLE III
EXPERIMENTAL CONFIGURATION AND VALUES

Configuration	Value
Sowing Drones	16
t_p (min)	420
Number of Control Units	5
Edge Devices	3
Time for Processing	30-45mins
Update Interval	25mins

In $t_{si+1} = t_{s+1}$ condition, if scheduling is reordered on the basis condition, if scheduling is reordered on the basis increasing $(t_c - t_{si+1})$, then $R(I) \in (t_c - t_{si+1})$. Similarly, if $t_c > t_p$, then scheduling is performed on the basis of descending n_l and hence $R(I) \in t_c$. If the above process is not performed, then $R(I) \in k$. This differentiation helps to achieve better classification of the audit logs based on relevance. The classification of $R(I) \in k$ or t_c or $(t_c - t_{si+1})$ helps to reorganize the production scheduling to improve the efficiency. This also augments in distributing k over the available M based on n_l and $(t_c < t_{si+1})$ reducing the backlogs.

III. PERFORMANCE ASSESSMENT

The performance of the proposed framework is tested with 200 sowing machines controlled by a smart device with human interaction in a large-scale farm with automated irrigation facility. The control unit initiates the irrigation system and then the sowing machine in the sequential order with respect to the line of movement. Drones are used as sowing machine in the smart farm that is capable of covering 7 kms. The control information end time of the drone takes place if it covers 3.5 kms. The sow time for a single drone is 30–45 min including the return time. The control unit transmits information for initiating the start/stop of the drone. Improper and unsynchronized information exchange from the control unit to the sowing drone results in stagnancy and failure of the process. In Table III, a detailed set of experimental configuration and values is presented.

Some performance metrics, such as delayed information, average processing time, average audit data, and k information distribution, are analyzed as a comparative study. For comparison, the existing methods service platform for service

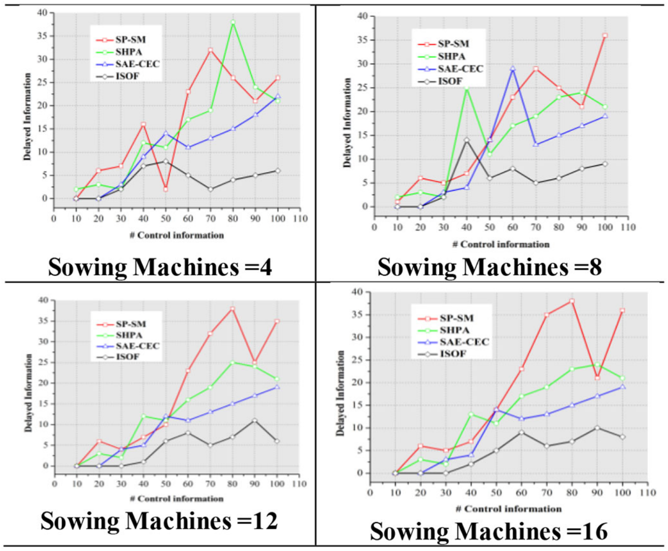

Fig. 6. Delayed information analyses.

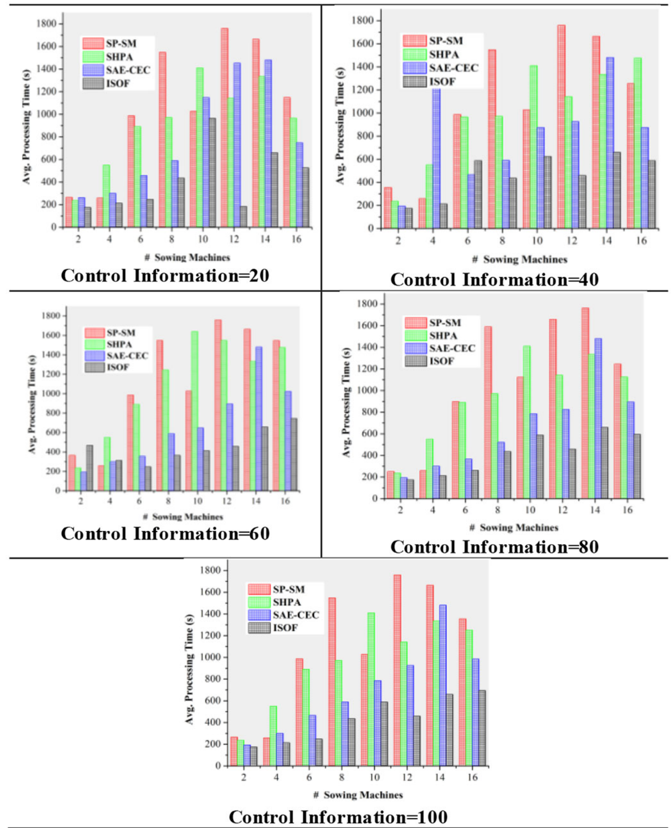

Fig. 7. Average processing time analyses.

management (SP-SM) [14], smart heterogeneous precision agriculture (SHPA) [15], and selecting algorithm for edge server and cooperation of EC (SAE-CEC) [17] are considered for the above-mentioned metrics.

A. Delayed Information Comparison

The delay information comparison is performed for the varying number of sowing machines (4, 8, 12, and 16) for the varying control information.

Fig. 6 presents a comparative study on delayed information observed in the existing methods and proposed framework. The process that fails $t_c \ngtr t_p$ is classified under k that requires $d(n_l, t_c)$ time for accomplishing the information task. These tasks are offloaded to other machines to reduce the backlogs at the end of t_p. To reduce the backlogs in information processing, offloading is performed on the basis of $(t_c - t_{si+1})$ and n_l per cases 2 and 3 of the ISOF phase. The main goal of this analysis is to reduce the chances of any N that fails $t_c \ngtr t_p$ condition. There are two other condition in which RS requires offloading, namely: $t_{si+1} = t_{s+1}$ and $t_c > t_p$. These two conditions are suppressed by mapping the information based on time factor and length. However, the proposed ISOF faces some backlogs depending on the number of U_M participation and due to the nonachievement of the ideal condition as illustrated in Fig. 3(a). The maximum number of k due to high $(t_c - t_{si+1})$ is suppressed at the intermediate level and R_M estimation. The precautionary measures are classified using the learning process to ensure acceptable rate of information processing reducing stagnancy at each (t_p/N) processing slot. Therefore, the number of N that is delayed is less comparatively.

B. Average Processing Time Comparison

The processing time is estimated for the varying information count along with the varying sowing machines.

The average information processing time with respect to the uniform machines operation U_M is presented as a comparison in Fig. 7. ISOF encourages concurrent and parallel information processing capability by classifying time unified and asynchronously operating machines. The operations of the machines are segregated on the basis of cases 1 and 2 as per the identified condition. Henceforth, the sequential information processing follows case 1 and $t_{si+1} = t_c$ condition of case 2. In these cases, the average processing time is $([\sum t_c/N])$. In a nonsequential/random processing, the time is estimated as per $t_{si+1} = t_{s+1}$ condition of cases 2 and 3. In these two cases, offloading requires an additional time to confine information within $t_c < t_p$. Therefore, an additional time of $(t_c - t_{si+1})$ is augmented to the $(t_s \sim t_c)$ time determining the processing time of N. This time is less as there is no relevant queening or stagnant information unlike conventional processing methods.

C. Audit Data Comparison

The size of audit data received in the business layer guided by the proposed ISOF is high compared to the other methods (Fig. 3). The initial data are classified based on k, t_c, and $(t_c - t_{si+1})$. In a default processing of $(t_c - t_{si+1})$ and t_c,the audit log is transmitted at regular intervals by the control unit. The edge device facilitated k information classification and the information rescheduling process is updated at irregular update intervals due to time delay in rescheduling. As shown in Fig. 8, the proposed ISOF reduces the number of information classified under k and hence, the information are completed

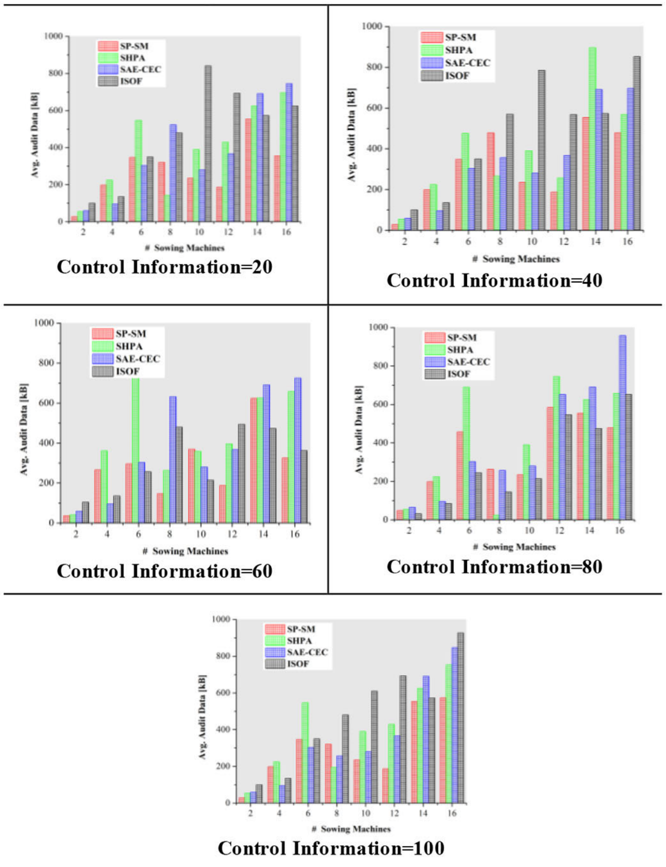

Fig. 8. Audit data analyses.

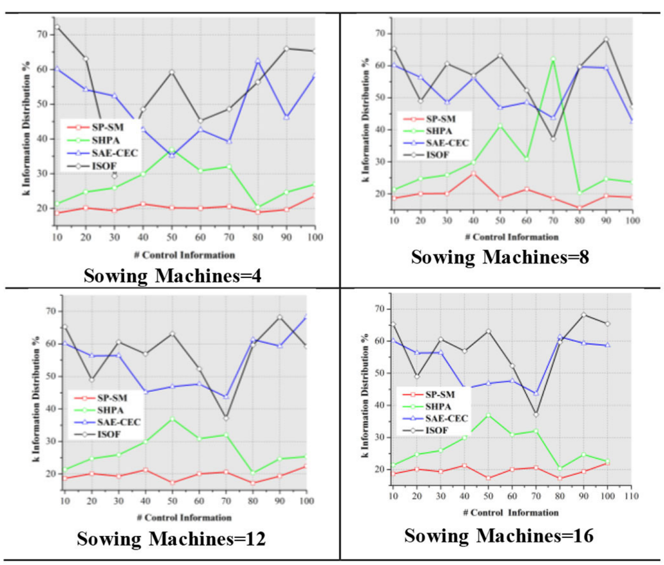

Fig. 9. k information distribution ratio analyses.

in t_c or extended in t_c confined within t_p. Therefore, the rate of audit log in regular and intermediate interval is high.

D. K Information Distribution Comparison

The classified k is to be assigned with another machine $\notin U_M$ to improve the rate of processing and reducing backlogs. The stagnant information are identified in the intermediate solution stage of the learning process and are then assigned

TABLE IV
EXPERIMENTAL VALUES WITH RESPECT TO VARYING SOWING MACHINES

Metrics	Sowing Machines	SP-SM	SHPA	SAE-CEC	ISOF
Delayed Information	4	26	21	22	6
	8	36	20	19	9
	12	35	25	15	6
	16	36	22	20	8
k Information Distribution %	4	23.62	26.96	58.21	65.26
	8	18.96	23.69	42.5	47.21
	12	22.36	25.36	68.32	59.21
	16	21.98	22.54	58.62	65.44

TABLE V
EXPERIMENTAL VALUES WITH RESPECT TO VARYING INFORMATION

Metrics	Information	SP-SM	SHPA	SAE-CEC	ISOF
Avg. Processing Time (s)	20	1148.2	965.85	748.1	524.75
	40	1254.7	1475.23	874.36	587.59
	60	1547.1	1475.32	1024.32	745.32
	80	1244.63	1125.49	894.52	594.12
	100	1354.62	1250.37	984.36	694.24
Avg. Audit Data [Kb]	20	356	696	745	625
	40	478	568	691	852
	60	325	658	725	362
	80	478	658	958	652
	100	574	753	847	927

to machine that follows asynchronous time for operations. If a machine with high R_M is identified under U_M or different from U_M, then the information is assigned for processing. In ISOF, the identified process as k is less and hence, it is mapped to the machines with $\{R_M\}$. However, the identified k is assigned to a machine based on two satisfying conditions: 1) $\min\{t_c - t_{si+1}\}$ or 2) max n_l. The information sorted from max to min n_l is assigned to the machine with highest reward and its $\rho_M = 1$. The information requests are processed with both uniformly operating machines and asynchronous machines at different t_s and t_c. These help to increase and retain the rate of k distribution in the proposed ISOF (Fig. 9). The experimental values are tabulated in Tables IV and V with respect to the varying sowing machines and information, correspondingly.

Based on the simulation analysis, the advantage of this framework is that it inherits the advantages of IoT and EC technologies with interoperable feature to aid information processing, information classification, offloading, and periodic updates. The performance of the proposed framework is tested in a corn farm and some common metrics, such as delayed information, processing time, audit data, and information distribution are analyzed for proving the reliability of the framework.

IV. Conclusion

This letter ISOF is designed for leveraging the performance of the industry 4.0 paradigm in a smart farming environment. The proposed framework strengthens scheduling, processing, and offloading of the agriculture-based industry 4.0 architecture. This is achieved by employing machine learning-assisted classification and edge-assisted offloading for reducing the stagnancy in information processing. Besides, the information distribution and classification process in the architecture are assisted by the interoperable IoT and EC support for improving the efficiency of the sowing drones in the smart agriculture systems configured with industry 4.0.

References

[1] P. Zheng *et al.*, "Smart manufacturing systems for Industry 4.0: Conceptual framework, scenarios, and future perspectives," *Frontiers Mech. Eng.*, vol. 13, no. 2, pp. 137–150, 2018.

[2] J. Wan *et al.*, "Software-defined industrial Internet of Things in the context of industry 4.0," *IEEE Sensors J.*, vol. 16, no. 20, pp. 7373–7380, Oct. 2016.

[3] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A survey on the role of IoT in agriculture for the implementation of smart farming," *IEEE Access*, vol. 7, pp. 156237–156271, 2019.

[4] D. Shadrin, A. Menshchikov, D. Ermilov, and A. Somov, "Designing future precision agriculture: Detection of seeds germination using artificial intelligence on a low-power embedded system," *IEEE Sensors J.*, vol. 19, no. 23, pp. 11573–11582, Dec. 2019.

[5] H. Sharma, A. Haque, and Z. A. Jaffery, "Maximization of wireless sensor network lifetime using solar energy harvesting for smart agriculture monitoring," *Ad Hoc Netw.*, vol. 94, Nov. 2019, Art. no. 101966.

[6] K. Janc, K. Czapiewski, and M. Wójcik, "In the starting blocks for smart agriculture: The Internet as a source of knowledge in transitional agriculture," *NJAS Wageningen J. Life Sci.*, vols. 90–91, Dec. 2019, Art. no. 100309.

[7] P. O. Skobelev, E. V. Simonova, S. V. Smirnov, D. S. Budaev, G. Yu Voshchuk, and A. L. Morokov, "Development of a knowledge base in the 'smart farming' system for agricultural enterprise management," *Proc. Comput. Sci.*, vol. 150, pp. 154–161, 2019.

[8] S. Alzu'Bi, B. Hawashin, M. Mujahed, Y. Jararweh, and B. B. Gupta, "An efficient employment of Internet of multimedia things in smart and future agriculture," *Multimedia Tools Appl.*, vol. 78, no. 20, pp. 29581–29605, 2019.

[9] X. Bai, L. Liu, M. Cao, J. Panneerselvam, Q. Sun, and H. Wang, "Collaborative actuation of wireless sensor and actuator networks for the agriculture industry," *IEEE Access*, vol. 5, pp. 13286–13296, 2017.

[10] H. Gosnell, N. Gill, and M. Voyer, "Transformational adaptation on the farm: Processes of change and persistence in transitions to 'climate-smart' regenerative agriculture," *Glob. Environ. Change*, vol. 59, Nov. 2019, Art. no. 101965.

[11] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, and E.-H. M. Aggoune, "Internet-of-Things (IoT)-based smart agriculture: Toward making the fields talk," *IEEE Access*, vol. 7, pp. 129551–129583, 2019.

[12] P. K. Thornton *et al.*, "A framework for priority-setting in climate smart agriculture research," *Agricult. Syst.*, vol. 167, pp. 161–175, Nov. 2018.

[13] M. Colezea, G. Musat, F. Pop, C. Negru, A. Dumitrascu, and M. Mocanu, "CLUeFARM: Integrated Web-service platform for smart farms," *Comput. Electron. Agricult.*, vol. 154, pp. 134–154, Nov. 2018.

[14] G.-A. Musat *et al.*, "Advanced services for efficient management of smart farms," *J. Parallel Distrib. Comput.*, vol. 116, pp. 3–17, Jun. 2018.

[15] Y. E. M. Hamouda and M. M. Msallam, "Smart heterogeneous precision agriculture using wireless sensor network based on extended Kalman filter," *Neural Comput. Appl.*, vol. 31, no. 9, pp. 5653–5669, 2018.

[16] N. Ahmed, D. De, and I. Hussain, "Internet of Things (IoT) for smart precision agriculture and farming in rural areas," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4890–4899, Dec. 2018.

[17] X. Li, J. Wan, H.-N. Dai, M. Imran, M. Xia, and A. Celesti, "A hybrid computing solution and resource scheduling strategy for edge computing in smart manufacturing," *IEEE Trans. Ind. Informat.*, vol. 15, no. 7, pp. 4225–4234, Jul. 2019.

[18] A. L. Diedrichs, F. Bromberg, D. Dujovne, K. Brun-Laguna, and T. Watteyne, "Prediction of frost events using machine learning and IoT sensing devices," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4589–4597, Dec. 2018.

[19] L. Liu *et al.*, "PestNet: An end-to-end deep learning approach for large-scale multi-class pest detection and classification," *IEEE Access*, vol. 7, pp. 45301–45312, 2019.

[20] F. Bu and X. Wang, "A smart agriculture IoT system based on deep reinforcement learning," *Future Gener. Comput. Syst.*, vol. 99, pp. 500–507, Oct. 2019.

[21] V. C. F. Aiken, J. R. R. Dórea, J. S. Acedo, F. G. D. Sousa, F. G. Dias, and G. J. D. M. Rosa, "Record linkage for farm-level data analytics: Comparison of deterministic, stochastic and machine learning methods," *Comput. Electron. Agricult.*, vol. 163, Aug. 2019, Art. no. 104857.

[22] Y. Feng, D. Gong, Q. Zhang, S. Jiang, L. Zhao, and N. Cui, "Evaluation of temperature-based machine learning and empirical models for predicting daily global solar radiation," *Energy Convers. Manag.*, vol. 198, Oct. 2019, Art. no. 111780.

[23] P. Filippi *et al.*, "An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning," *Precision Agricult.*, vol. 20, no. 5, pp. 1015–1029, Aug. 2019.

[24] A. Goldstein, L. Fink, A. Meitin, S. Bohadana, O. Lutenberg, and G. Ravid, "Applying machine learning on sensor data for irrigation recommendations: Revealing the agronomist's tacit knowledge," *Precision Agricult.*, vol. 19, no. 3, pp. 421–444, 2017.

[25] L. Han *et al.*, "Modeling maize above-ground biomass based on machine learning approaches using UAV remote-sensing data," *Plant Methods*, vol. 15, no. 1, Apr. 2019, Art. no. 10.