International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 9, Issue.7, July 2021 E-ISSN: 2347-2693

Dynamic Information Validation Scheme in Internet of Things: Software Agent based Approach

Sharanappa P. H.[1*], Mahabaleshwar S. Kakkasageri[2]

[1,2]Dept. of Electronics and Communication Engineering, Basaveshwar Engineering College (Autonomous) Bagalkot - 587102, Karnataka, India

[*]*Corresponding Author:phsharanu@gmail.com, Tel.: +91-99008-55255*

DOI: https://doi.org/10.26438/ijcse/v9i7.110 | Available online at: www.ijcseonline.org

Received: 19/Jul/2021, Accepted: 20/Jul/2021, Published: 31/Jul/2021

***Abstract*—** The Internet of Things (IoT) and its various application domains are drastically changing people's lives by providing intelligent services that will eventually become an intrinsic part of their daily environment. The data flows received from various actuators and sensors are used to power the IoT services. The accuracy and security of sensor data supplied across the Internet of Things system is a vital aspect in ensuring that IoT services work properly. As a result, data validation in a remote IoT network is becoming increasingly important. Even though the immediate option of establishing duplicate identical systems can provide validation, real-world change limitations can make this difficult, if not impossible. So here in this paper we have proposed an intelligent validation scheme. We have evaluated the performance and effectiveness of proposed scheme by comparing with an existing technique that uses BL and KP-ABE scheme. In terms of time necessary to sense the data, data gathering time, time required to validate the data, and end to end delay, the suggested method outperforms the existing validation methodology.

***Keywords*—**Ingternet of Things, Information Validation, Multi agents

I. INTRODUCTION

Introduction In a distributed sensor system, reliability, trust, and validating the acquired data has become a major concern. Despite the obvious answer of installing duplicate identical systems to offer validation, real-world change limits will make this more difficult, if not unattainable. The number of smart phone apps that use sensors to achieve a range of goals has increased. Even if they are created by professionals or businesses, there are few applications that publish or expose how the sensors' data is processed. This is a problem, especially when these applications are expected to be utilized in situations where the data is required to be used to categorize daily life activities or when employed in medical settings. Because of the multiprocessing architecture of mobile devices, which limit computational capacity and battery life, sensor data is sometimes worthless in its raw form, demanding further processing to make it representative of the event or object that it is designed to measure. To ensure that sensor data is suitable for feeding into higher-level algorithms, validation subtasks must be included in the recording and subsequent processing of this data. Furthermore, using sensor data to feed higher-level algorithms must guarantee a minimum degree of inaccuracy, which is defined as the difference between the output of these applications, which are based on limited computing mobile platforms, and the output of a gold standard. This work introduces and explores intelligent validation systems and methodologies, as well as their evaluation in an application that leverages sensor data to offer meaningful output to its users, in order to address this issue. The algorithms are detailed, as well as their applications. The mathematical method is frequently studied to ensure that the results are as trustworthy as possible.

The remaining part of the paper is organized as follows: Related works are discussed in Section II; Section III addresses the proposed validation approach; Section IV and V includes simulation and result analysis as well as a crucial comparison with existing schemes respectively; Conclusions are drawn in Section VI.

II. RELATED WORK

The most widely used data validation algorithms are reviewed, along with their application scenarios, and a classification for these algorithms is proposed based on the detection of faulty data and data correction methods along with some assisted tools. Despite the wide variety and types of data validation algorithms available, there is a lack of publicly available information on the validity of many mobile applications. Furthermore, even within their respective categories, it is difficult to present a critical comparison of the discussed approaches since their efficiency is highly dependent on their specific application [1].

A novel data validation scheme for WSN, where the authors used qualitative methods such as temporal correlation, heuristic rule, spatial correlation, modified z-score and Chauvenet's criterion [2]. The performance is

assessed using real data samples from a WSN prototype for environmental monitoring that have been injected with various forms of data faults like out-of-range faults, struck-at faults, and outliers and spike faults. The authors provide a state-of-the-art assessment of data validation methods for recent works, as well as a summary of their constraints and challenges[3]. To evaluate the cost of detecting errors at the sensor board level, they proposed an energy dissipation model and examine the effects of various data validation approaches on the sensor board's energy consumption.

Sensor data validation scheme for real-time IoT/WSN level is implemented based on adaptive threshold is discussed in [4]. Where the approach will detect various types of sensed data errors and introduces a mechanism to classify them as an error or an event. In a big-data scenario, defining a framework for incorporating validation actions at all stages of the measurement process that includes measurement, retrieval of data, storing and organizing the data, processing the data and presenting it especially for an Industry 4.0 application[5].

A sensor validation approach that uses sensory substitution to improve the robustness, stability, and dependability of sensed data by detecting false positives and negatives and corroboration of true positives and negatives. A fog-based validation layer approach that uses sensory substitution is presented [6]. The accuracy and security of sensor data transmitted over the IoT system is a critical factor in ensuring that IoT services work properly. An "apriori" knowledge of the behavior based mechanism that detects the irregular sensor events is presented in [7].

The practical methods for validating and correcting sensor data, as well as how to use defective sensor data to train cloud-based models that can predict sensor life are proposed [8]. The work also explores different approaches for validating the accuracy of prediction models, the operational sequence of the proposed work is: data gets received by the data validation and life prediction layer; the sensor life prediction layer adds corresponding reliability parameters; sensor data validation layer validates the data and in case of errors it adds an inference block; an error matrix block is added if the life prediction layer reports an error; interpolation block is added if an error is reported by the validation block. The completed data structure is saved and moved on to the next layer.

A trust based validation scheme for identifying the misbehaving or selfish node is proposed in [9]. If the Cluster head becomes a greedy or malicious node, the device would be unable to detect the fault and recover from it. As a result, spotting greedy or misbehaving CH is critical. Identifying defective data from sensors, on the other hand, is an important problem in data collection. The scheme also devises a method for detecting false data for mobile WSN. An artificial neural network based sensor fault detection and validation scheme is discussed in [10]. The relative merits of three types of ANNs for activity classification are contrasted using PCA-based y-indices to quantify the differences between groups of sensor readings in a time rolling window. By creating and using a proposed 'y-index' metric, it describes a technique that explicitly defines which sensor in the system is at fault.

The concept of modeling the impact of both spatial and temporal correlation was tested on a geophysical field simultaneously and demonstrated the Sliding Window Outlier Detector, a semi-supervised outlier detection system that constructs a spatio-temporal model of data obtained from the sliding window background and uses this model to identify and classify outliers[11]. A distance-based scheme that detects outliers in data streams is presented in [12]. It deals with the sliding window model. The outlier queries are executed to detect the anomalies using two algorithms. the first algorithm gives precise answer to the queries and has more space requirements, whereas in the second one it has got less memory requirement and gives approximate response depending on the exact estimations with statistical assurance.

A well-organized summary of the different methods for detecting outliers in temporal data that have been suggested [13]. It also includes a discussion of various data types, outlier concepts and a brief introduction to the techniques. Finally, different applications for which these techniques found efficient are discussed. Implementation of a formalized way to identify outliers in spatio-temporal lattice data with an intention of identifying useful and relevant outliers in climate datasets is discussed in[14]. It emphasizes the importance of clarifying basic data structure. Using a global climate dataset as a case study, two types of spatio-temporal outliers are defined that are based on the three factors that are suggested in identifying an outlier.

A framework for recognizing noteworthy geographically based events from the repository of user-generated data is introduced [15]. It includes efficient algorithms for detecting geographically based knowledge bursts, assigning demographic variables to them, and defining collections of descriptive keywords. The results of the proposed scheme are discussed in BlogScope. The authors describes the basics of outlier detection in sensor networks and identifies major criterion related to the classification of schemes for outlier detection. It also addresses important features and description of the outlier detection methods in the current scenario using the proposed technique-based taxonomy [16].

Methods increase the validity of life logging data in an IoT-enabled healthcare system by focusing on life logging physical activity (LPA) is discussed in [17]. In IoT healthcare environments, the LPAV-IoT model is offered as a rule-based adaptive LPA validation (LPAV) model for reducing irregular uncertainty (IUs) and predicting data consistency. For studying major aspects affecting LPA validity, a methodology based on four layers and three modules in LPAV-IoT is proposed. Current research efforts on information management strategies used in

VANETs for safety-related applications, such as gathering, aggregation, validation, and dissemination, as well as future research and development directions are summarized in [18].

A two-stage technique that combines data validation and reconstruction approaches with contamination event identification techniques is discussed in [19]. The fundamental concept is to offer a valid dataset after the first stage (i.e., without errors or missing values) so that contamination sources introduced inside the building envelope can be detected and isolated in the second stage. For data validation and dynamic uncertainty estimates of a self-validating sensor, a self-validating strategy based on grey bootstrap method is developed. The self-validating sensor's failure detection, isolation, and recovery (FDIR) based on the GM(1,1) predictor can concurrently identify and isolate faults while also completing failure recovery with high accuracy and speed [20].

Nonlinear principal component analysis is compared using two standard approaches. Auto-associative Neural Network and Kernel PCA are the two techniques. The validity of sensor data is discussed using these methods[21]. The outcome is a novel method for mapping nonlinear components of a set of data obtained from a nonlinear quasi-static system. For the detection of a fault in a group of sensors, the technique employs a Bayesian network. The relationships and independencies among all of the sensors are represented by this Bayesian network. A second Bayesian network isolates the malfunctioning sensor from the rest of the false positives. This isolation is achieved incrementally, i.e., a chance of failure vector is provided at any time, and the quality of the belief measures improves as the computation time is increased [22].

Creation of a self-validation technique in an intelligent sensor that meets the following criteria: The current state of the measurement, its uncertainty, and the measurement's confidence, as well as the ability to detect conditions of the measured variable when the process is running and when it is halted is mentioned[23]. The performance of two different types of predictors was assessed. The first employs Taylor's series, whereas the second employs autoregressive functions. For the pressure sensor, a neural network-based data validation method has been designed and validated. Back propagation algorithm is used to produce an estimated value for neural networks, and then a fault detection method called sequential probability ratio test is used to determine the sensor's trueness[24].

For the online measurements validation and confidence evaluation of multifunctional sensors, a novel technique combining fuzzy logic and polynomial predictive filters is suggested. The prediction errors are created by comparing the predicted outputs to actual measurements, and then some fuzzy logic rules are used to calculate the confidence level so that to some extent, it reflects the measurement quality. The proposed method for ensuring the accuracy of measurements can detect incorrect data under faults[25].

The final validated measurement value, that uses a multivariable relevant vector machine with fault detection, isolation, and recovery technology, in which polynomial predictive filters with low computation complexity are used for data validation and then incorrect measurements are validated or corrected on the fly. it also provides a novel random fuzzy variable-based uncertainty evaluation technique for on-line verified uncertainty estimate that completely considers the negative impact of various errors [26]. A secure cloud transfer via key policy attribute-based encryption (KPABE) is discussed in [27]. The user's identity is first verified. The user data is then encrypted using the KP-ABE technique. Finally, Burrows-Abadi-Needham (BAN) logic is used to validate data and protect privacy.

Authors present an intelligent strategy for IoT data collection using an agent-based methodology [28]. The suggested research work's key contribution is to design and create an intelligent system for IoT data collection. An extensive survey on architectures, applications and research issues in IoT are provided in [29]. Here in this things, items and platforms are required to disclose personal or confidential data. In this scenario, the question of how to strike a compromise between privacy and security remains unanswered. Furthermore, detecting harmful items is critical during the communication process. The detection of a malicious object may result in a delay. A hybrid confidentiality technique that combines the Advanced Encryption Standard, Elliptic Curve Cryptography, and Message- Digest techniques is mentioned in [30]. Geo encryption, or location-based encryption, is also combined with a hybrid method to ensure that all devices are secure. The hybrid algorithm ensures high data transmission confidentiality for the Internet of Things.

A hybrid method for maintaining data privacy is presented in [31]. A mixture of the Message Digest algorithm, Elliptic Curve Cryptography, and Advanced Encryption Standard techniques is used in the proposed algorithm design. The hybrid technique works by having the destination IoT node share its geotag with the source node.

A. Proposed Work

The proposed model uses two types of agents: Sensor Manager Agent (SMA), Information Collection Agent (ICA), Cloud Manager Agent (CMA). ICA is mobile agent, where as CMA and SMA are static agents. In the proposed validation scheme, the data is to be validated against out of range faults, stuck at faults and outliers.

The scheme operates in the following sequence:

1) The values that are sensed periodically from the environment are collected by ICA.

2) On arrival of ICA, Cloud Manager Agent validates the ICA data against out of range faults, stuck at fault and outliers.

3) If CMA validates the incoming data then the same is stored in the cloud.

4) Otherwise one of the above said faults might have occurred and the data is discarded.

B. Our Contributions

Our contributions for the proposed scheme includes the following.

- Design of intelligent scheme for information validation.
- Avoid Employing the agent based approach for validation of data by detecting the above said faults.
- Elimination of storing the invalid redundant data.

The proposed scheme of information validation has been compared with data validation algorithm discussed below.

III. INTELIGENT INFORMATION VALIDATION

Here in this section network environment has been described and brief explanation of the agents and the detail agency for information validation in IoT is presented.

A. Network Environment

The For Proposed system of validation, the environment shown in Figure 1 is considered, where number of IoT devices are considered to be sensor nodes, which constitute the physical layer. These devices will be able to communicate with the cloud environment through the internet gateway by the service provided by the network layer. Each of the devices are equipped with different modules like GPS, onboard circuitry required for data processing etc.. The devices also have the static agent SMA for managing the sensing. ICA can communicate to both SMA and CMA shown in the Figure 2.

Each device has its own agent platform and agency which contains software agents. Software agents are defined as the independent piece of code or program that gets invoked on the agent platform of its respective host. These agents utilize their Knowledge Base to implement a particular task without disturbing the functionality of the host on which it is running.

Agents can be static or mobile. The agents that are static may be platform dependent, can be created, utilized and deleted as and when it is required. Whereas mobile agents are the programs that are expected to be independent of the platform on which it is supposed to run. In real time these can be designed, migrated, utilised and removed whenever necessary.

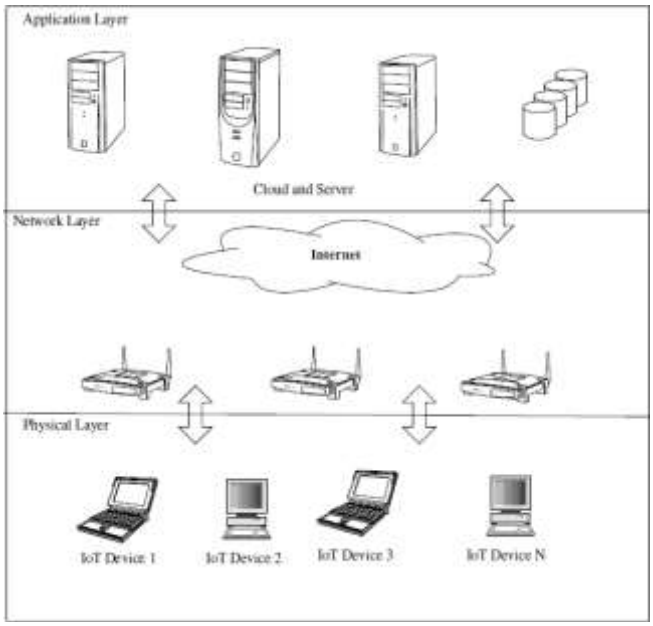

Figure 1. Network Environment

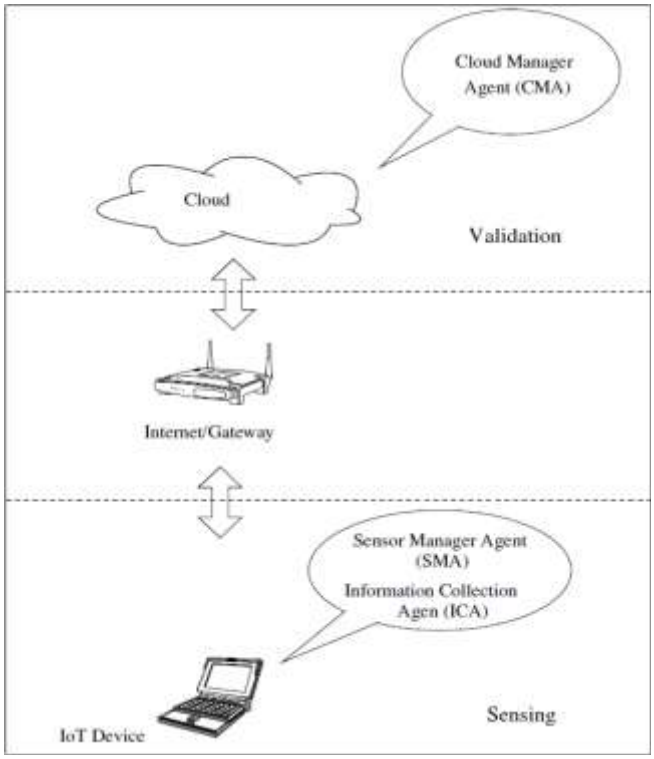

Figure 2. Validation Scenario

B. Definitions

In this section, different types of faults that can be encountered during validation process are defined, that assist in the proper perception and depiction of vital information validation process.

- ***Out-of-range faults*** are defined as sensor data samples that considerably depart from the predicted range of values. Sensor values that are physically impossible to achieve in the deployed location are referred to as out-of-range errors.

- ***Struck-at faults*** are a collection of data samples that show little or no variation over a longer length of time than predicted. The data is frozen or held at a specific value. It could be inside or outside of the intended range. Constant fault and struck-at fault are two terms for the same thing.
- ***Outliers*** are isolated data samples that differ significantly from the rest of the sample but still fall within the predicted range of values.
- The rate of change in data sample gradient over time is substantially faster than expected with ***Spike fault***. It happens when at least a few data samples are combined in a succession.

C. Agent Model

The agent model proposed here for information validation uses mobile and static agents. The model consists of Knowledge Base (KB), static agents and mobile agents. ICA is a mobile agent, SMA and CMA are static agents. The agency is shown in Figure 3.

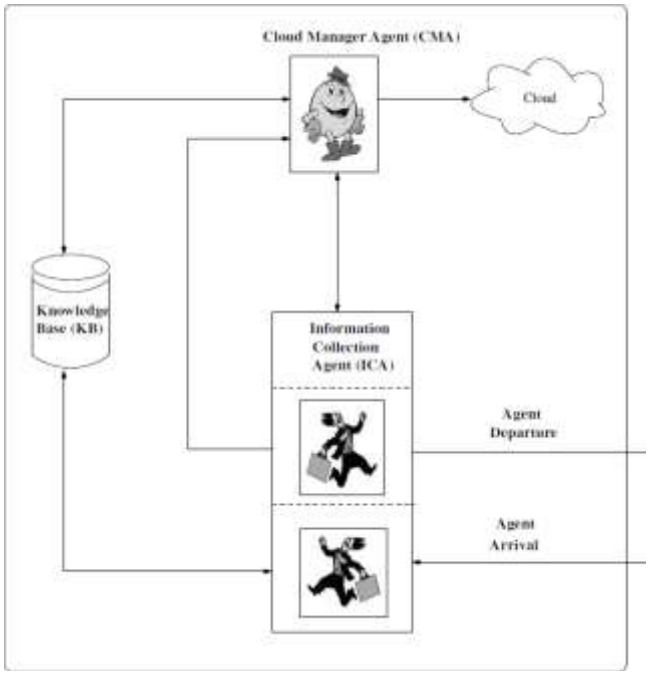

Figure 3. Agency Model

1) *Knowledge Base (KB):* Knowledge Base consists of information regarding device or node identity (ID) and IDs of the neighbours, bandwidth available for communication, present and past information gathered, etc. Knowledge *base is updated and read by ICA and CMA.*
2) *Sensor Manager Agent (SMA):* Sensor Manager Agent is a static agent that executes on every sensor node or device. As and when the sensor senses the physical change in the environment for example temperature, light, humidity etc., it is responsible for invoking the ICA.
3) *Information Collection Agent (ICA):* ICA is a mobile agent that execute on every device. It is platform independent agent. Once triggered by SMA, it carries the sensed information to knowledge base and to CMA. On its arrival knowledge base is updated and the same is read by CMA.
4) *Cloud Manager Agent (CMA):* It is a static agent responsible for validating the information received by ICA. It reads the knowledge base for the past values of sensor data and compares it with the present value and checks it for its validity. CMA is designed to detect and validate above mentioned faults or errors using the following methods.

a) *Rule Based Method:* It uses a heuristic rule, based on domain knowledge or sensor sensing range to verify and validate the data. Out-of-range faults can be detected using this technique.

let x be the sensed sample of data
let δ_{min} and δ_{max} be the threshold values
δ_{min} and δ_{max} are decided based on the sensing range of the sensor or even based on domain knowledge.

if ($\delta_{min} \leq x \leq \delta_{max}$)
then x is likely to be valid sample
else x is likely to be fault sample

Algorithm 1. Out of range fault detection

```
1  : Input : sensed data set (x[N]), δmin , δmax
2  : Output: Data set x[N] with status updated.
3  : Begin
4  : for i ← 0 to N do i ← i+1
5  :     if (x[i] ≥ δmin AND x[i] ≤ δmax ) then
6  :         x[i] is valid sample
7  :     else
8  :         x[i] is invalid sample
9  :     end if
10 : end
```

b) *Temporal Correlation Method* evaluates and validates data by comparing data from different sensor nodes. This method can adopt either zero difference criteria or threshold value criteria to check the data for its validity. In zero difference criteria it checks whether the difference between successive samples of data from a sensor device is zero for multiple time instances. If the difference is zero, then the sample is likely to be at fault else likely to be valid.

let x_i be the sensed sample of data at time t_i
let x_{i+1} be the sensed sample of data at time t_{i+1}
let x_{i+2} be the sensed sample of data at time t_{i+2}

if $x_{i+1} - x_i = 0$
$x_{i+2} - x_{i+1} = 0$
(if this happens successively for multiple time instances)
then the sensed sample is likely to be fault
else valid sample.

In threshold value criteria it checks for the difference between successive data is more than the threshold value. If it is more, then the data sample is likely to be fault else it is valid.

Algorithm 2. Stuck at fault detection using zero difference

```
1  : Input : sensed data set (x[N])
2  : Output: Data set x[N] with status updated.
3  : Begin
4  : for i ← 0 to N do i ← i+1
5  :         if (|x[i] - x[i+1]| = 0 ) then
6  :             x[i] is invalid sample
                 x[i+1] is invalid sample
7  :         else
8  :             x[i] is valid sample
                 x[i+1] is valid sample
9  :             end if
10 : end
```

c) Spatial Correlation Method compares sensor node data with data from neighbouring nodes, SCM checks and validates information. On comparison if the difference between data is less than the threshold then the data sample is likely valid else it is at fault. To detect the faults among the data samples of neighbour nodes following technique is used.

let N_i and N_j be neighbours
x_i be sensed sample of data by node N_i
x_j be sensed sample of data by node N_j
$\hat{x}_i$ be the expected value of N_i based on x_j
that can be estimated by least square error.
if $(x_i - \hat{x}_i) < \delta$ then x_i is likely good sample
else x_i is likely fault.

Algorithm 3. Stuck at fault detection using threshold value

```
1  : Input : sensed data set (x[N])
2  : Output: Data set x[N] with status updated.
3  : Begin
4  : for i ← 0 to N do i ← i+1
5  :         if (|x[i] - x[i+1]| > δ) then
6  :             x[i] is invalid sample
                 x[i+1] is invalid sample
7  :         else
8  :             x[i] is valid sample
                 x[i+1] is valid sample
9  :             end if
10 : end
```

Algorithm 4. Stuck at fault detection using Spatial Correlation Method

```
1  : Input : sensed data set (x[N])
2  : Output: Data set x[N] with status updated.
3  : Begin
4  : for i ← 0 to N do i ← i+1
5  :         if (|(x[i] - x̂_i )| < δ ) then
6  :             x[i] is valid
7  :         else
8  :             x[i] is invalid
9  :             end if
10 : end
```

d) Statistical Method uses mean, median, mode, standard deviations, and other metrics to verify and validate data. To check and validate data, we employed SM such as Modified Z-Score (MZS). Outliers can be detected using this technique, here the z-score is defined as shown in "1".

$$z_i = \frac{y_i - \bar{y}}{s} \quad (1)$$

where y_i = data sample, $\bar{y}$ = mean of data and s = standard deviation.

Modified Z-Score (MZS) uses Median Absolute Deviation (MAD) and is calculated using "2" as shown below.

$$MAD = median(|x_i - \tilde{x}|) \quad (2)$$

where $\tilde{x}$ = median of data samples and x_i = data sample. The Modified Z-Score (MZS) is given by "3" as shown.

$$M_i = \frac{0.6745(x_i - \tilde{x})}{MAD} \quad (3)$$

For any data x_i if its MZS i.e. $M_i > 3.5$ then the data sample is at fault(outlier).

Algorithm.5 Outlier detection using Statistical Method (Modified Z-Score)

```
1  : Input : sensed data set (x[N])
2  : Output: Data set x[N] with status updated.
3  : Begin
4  : for i ← 0 to N do i ← i+1
5  :         MAD = median(| x_i − x̃ |)
6  :         M_i = 0.6745 × (| x_i − x̃ |) / MAD
7  :         if (M_i > 3.5 ) then
8  :             x[i] is invalid
9  :         else
10 :             x[i] is valid
11 :             end if
12 : end
```


IV. SIMULATION AND PERFORMANCE PARAMETERS

The recommended agent based validation scheme has been simulated using Dev. C++ environment. In this section simulation inputs and performance parameters are discussed.

A. Simulation Inputs

For the proposed scheme of validation, the inputs considered for simulation purpose are shown in Table 1 below.

Table 1. Simulation Inputs

Sl. No.	Input parameters	Specifications
1	No. of nodes : N	10-50
2	Sensing area: L×B	500m×500m
3	Sensing range: SR	25m, 50m
4	Data packet size(PS)	48kb

B. Performance Parameters

To evaluate the performance of the proposed research work we have considered the following parameters.

- Sensing time: It is defined as the time taken by the sensor manager agent to sense the physical change in the environment like temperature, humidity, light etc., and is expressed in milliseconds (ms).
- Gathering time: It is the time taken for information collection agent to gather the information sensed by the sensor manager agent to carry to the knowledge base and CMA. It is expressed in milliseconds (ms)
- Data validation delay: It is defined as the time taken for cloud manager agent to validate the received information using the above mentioned algorithms, and is expressed in milliseconds (ms)
- Average computational delay: It is the time average time taken by the agency to validate the data using different algorithms mentioned above. It is measured in milliseconds (ms).
- End to end delay: It is defined as the total time taken for the agency to sense, gather and validate the information and is expressed in milliseconds(ms).

V. RESULT ANALYSIS

To evaluate and analyse the performance of the scheme proposed, the metrics mentioned above are considered and are plotted across the graph for varying number of nodes and Sensing Range(SR). The first 5 results plotted below corresponds to the proposed validation technique, and the remaining results corresponds to the comparison of the proposed work with the existing Ban Logic and KP-ABE (BL-KPABE)scheme.

The time taken to detect the out of range faults is plotted against number of IoT devices (nodes) for different sensing range in Figure 4 below. It is evident that the time taken to detect the faults increases with increase in number of nodes.

Figure 5 shows the time taken for detecting the stuck at fault using zero difference method. It can be seen that the time increases with increase in number of nodes.

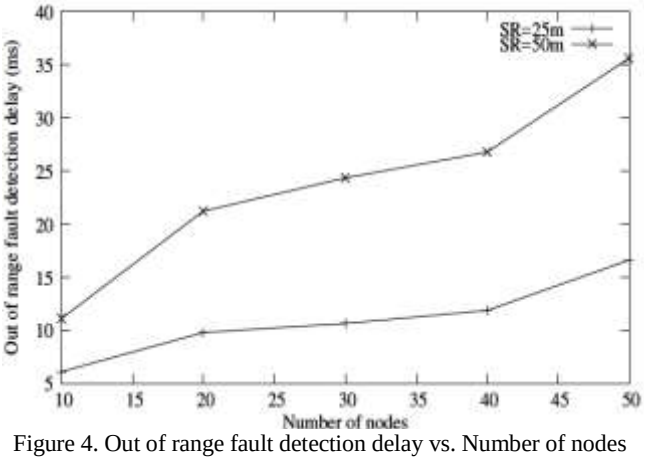

Figure 4. Out of range fault detection delay vs. Number of nodes

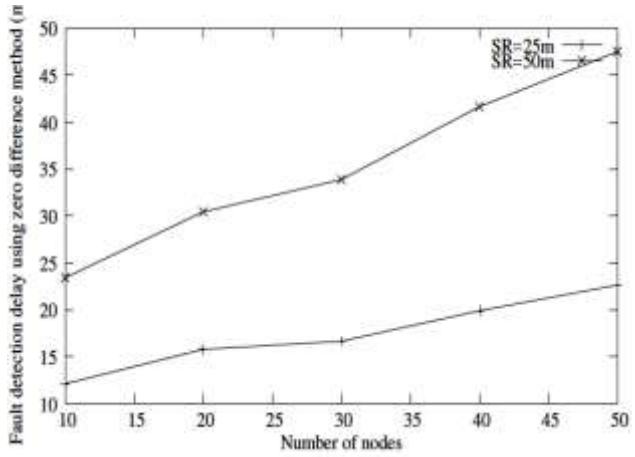

Figure 5. Stuck at fault detection delay vs. Number of nodes (zero difference method)

Detection of stuck at fault can also be done using threshold value method and time taken for the same is depicted in Figure 6 against varying number of nodes with different sensing range. It is observed that, as the number of nodes increases the time taken also increases.

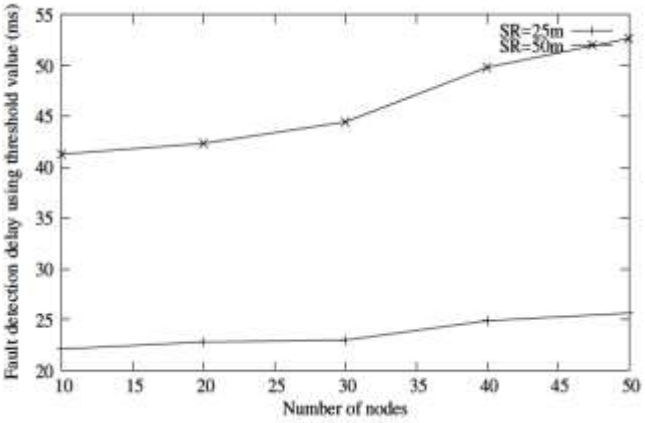

Figure 6. Stuck at fault detection delay vs. Number of nodes (threshold value method)

Spatial Correlation Method (SCM) can also be applied for detection of stuck at faults. The time taken for the same is shown in Figure 7 below. It is evident from the graph that the detection time is more if number of devices increases.

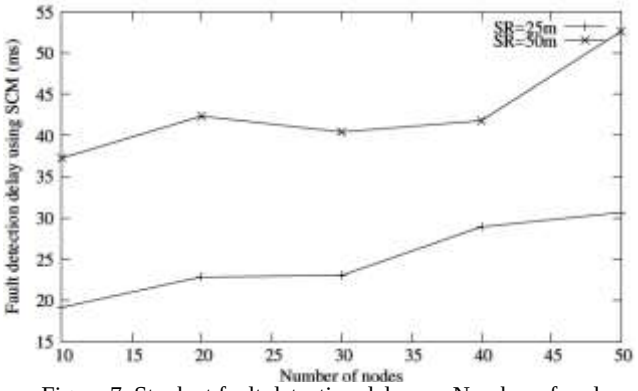

Figure 7. Stuck at fault detection delay vs. Number of nodes

Outlier detection time using the Modified Z-Score (MZS) method is depicted in Figure 8. It is seen that the detection time is more if number of nodes increases.

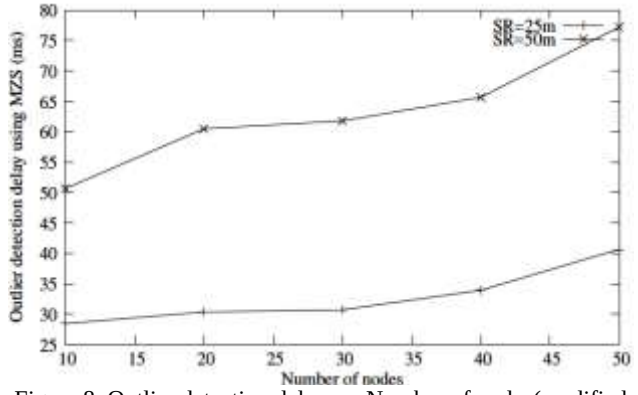

Figure 8. Outlier detection delay vs. Number of nodes(modified z-score method)

The proposed scheme of validation is compared with the existing technique that uses BL and KP-ABE scheme. The following results reveal that the proposed scheme performs better in terms of data sensing time, gathering delay, data validation time, computational delay and end to end delay.

Figure 9 depicts that the proposed scheme requires lesser sensing time when compared with the work under comparison.

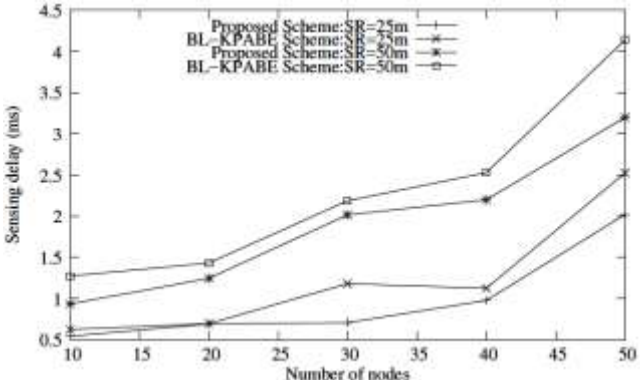

Figure 9. Sensing delay vs. Number of nodes

Figure 10 shows the analysis of time taken to gather the data against varying number of nodes, and it is clear that as the number of nodes increases, gathering time also increases. The proposed work takes less time for the same when compared with the existing technique.

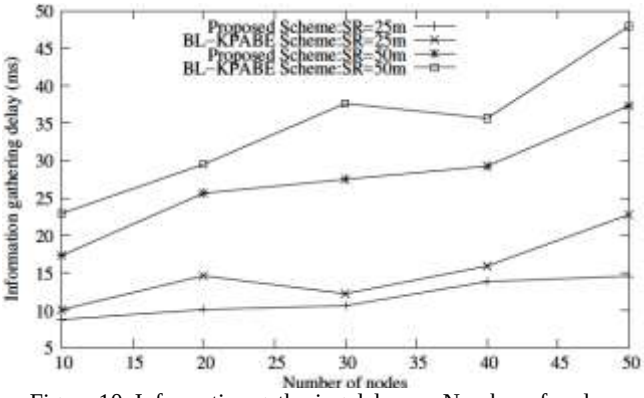

Figure 10. Information gathering delay vs. Number of nodes

Time taken to validate the data using the algorithms mentioned above is plotted against number of nodes in Figure 11 and it is observed that the validation time increases with increasing nodes. The work proposed here outperforms compared with the existing technique that uses BL and KP-ABE scheme.

The average computational delay for the validation process by the proposed agent scheme plotted against number of nodes in Figure 12 and it is found lesser than that of the existing standard.

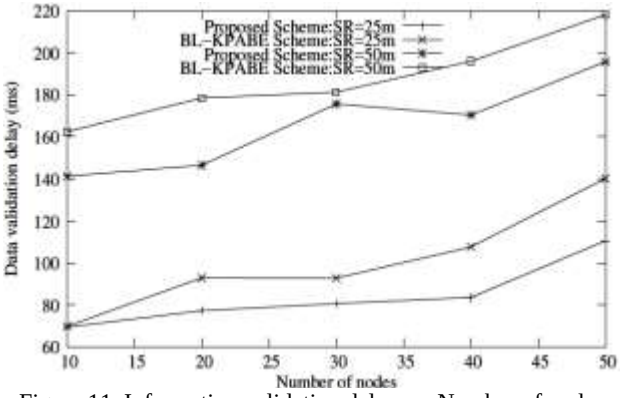

Figure 11. Information validation delay vs. Number of nodes

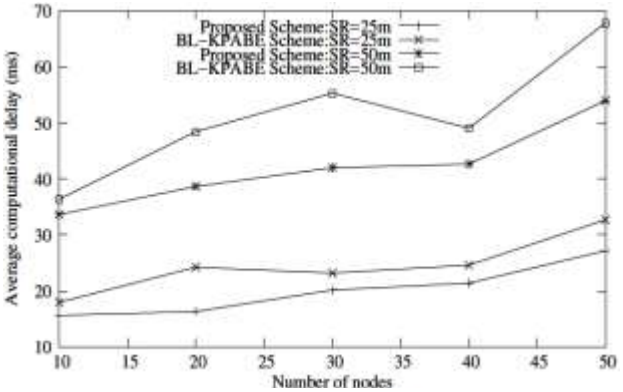

Figure 12. Average computational delay vs. Number of nodes

Figure 13 shows the total time taken to sense, gather and validate the data and is referred as end to end delay. The same is depicted for increasing number of nodes. It can be seen that the proposed model has lesser end to end delay and performs better that the work under comparison.

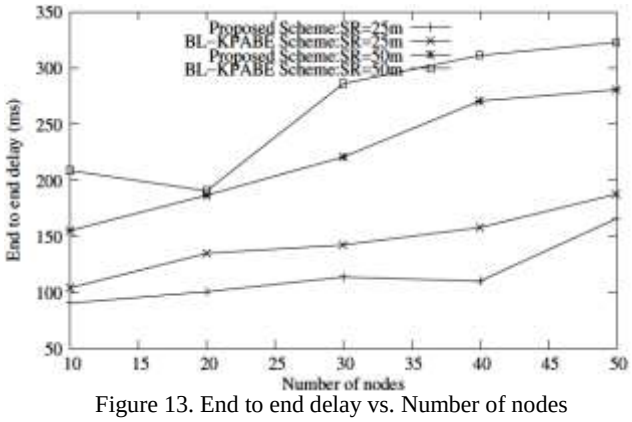

Figure 13. End to end delay vs. Number of nodes

VI. CONCLUSION

In this research work we proposed an agent based intelligent data validation scheme for validating the huge data that gets generated across the IoT network. It appears from the results that the proposed algorithm is more adaptive, can be implemented easily for real-time applications. Further the proposed algorithm is compared with the data validation with BL and KP-ABE scheme and found to be more efficient in terms of time taken to sense the data, data gathering time, time required to validate the data and end to end delay.

REFERENCES

[1] I. M. Pires, N. M. Garcia, N, Pombo, F. Florez-Revuelta, N. D. Rodriguez, *"Validation Techniques for Sensor Data in Mobile Health Applications"*, Journal of Sensors, Vol. **2016**.

[2] J. Ravichandran, A. I. Arulappan, *"Data Validation Algorithm for Wireless Sensor Networks"*, International Journal of Distributed Sensor Networks, Vol. **2013**.

[3] N. A. M. Alduais, J. Abdullah, A. Jamil, L. Audah and R. Alias, *"Effect of Data Validation Schemes on the Energy Consumptions of Edge Device in IoT /WSN"*, In the Proceedings of the 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Limassol, , pp. **77-81**, **2018**.

[4] N. A. M. Alduais, J. Abdullah, A. Jamil, L. Audah and R. Alias, *"Sensor Node Data Validation Techniques for Realtime IoT/WSN Application"*, In Proceedings of the 14th International Multi-Conference on Systems, Signals & Devices (SSD), Marrakech, , pp. **760-765**, **2017**.

[5] G. D'Emilia and A. Gaspari, *"Data Validation Techniques for Measurements Systems Operating in a Industry 4.0 Scenario a Condition Monitoring Application"*, 2018 Workshop on Metrology for Industry 4.0 and IoT, Brescia, , pp. **112-116**, **2018**.

[6] L. Russell, F. Kwamena and R. Goubran, *"Towards Reliable IoT: Fog-Based AI Sensor Validation"*, 2019 IEEE Cloud Summit, Washington, DC, USA, pp. **37-44**, **2019**.

[7] H. Sandor, B. Genge and Z. Szanto, *"Sensor data validation and abnormal behavior detection in the Internet of Things"*, In the Proceedings of the 16th RoEduNet Conference: Networking in Education and Research (RoEduNet), Targu Mures, pp. **1-5**, **2017**.

[8] V. Chacko and V. Bharati, *"Data Validation and Sensor Life Prediction Layer on Cloud for IoT"*, In the Proceedings of the IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, pp. **906-909**, **2017**.

[9] M. Sutaone, P. Mukherj and S. Paranjape, *"Trust-based Cluster head validation and outlier detection technique for Mobile Wireless Sensor Networks"*, In the Proceedings of the 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, pp. **2066-2070**, **2016**.

[10] Y. Zhang, C. M. Bingham, M. Gallimore, Z. Yang and J. Chen, *"Applied sensor fault detection and validation using transposed input data PCA and ANNs"*, In the Proceedings of the 2012 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Hamburg, pp. **269-274**, **2012**.

[11] A. Appice, P. Guccione, D. Malerba, A. Ciampi, *"Dealing with temporal and spatial correlations to classify outliers in geophysical data streams"*, Information Sciences: an International Journal, Vol.**285(1)**, pp. **162-180**, **2014**.

[12] F. Angiulli, F. Fassetti, *"Detecting Distance-Based Outliers in Streams of Data"*, In the Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management, CIKM, Lisbon, Portugal, **2007**.

[13] M. Gupta, J. Gao, C. C. Aggarwal and J. Han, *"Outlier Detection for Temporal Data: A Survey"*, IEEE Transactions on Knowledge and Data Engineering, Vol. **26**, No. **9**, pp. **2250-2267**, **2014**, doi: 10.1109/TKDE.2013.184.

[14] S. Yuxiang, X. Kunqing, M. Xiujun, J. Xingxing, P. Wen and G. Xiaoping, *"Detecting Spatio-temporal Outliers in Climate Dataset: A Method Study"*, In the Proceedingsof the 2005 IEEE International Geoscience and Remote Sensing Symposium. IGARSS '05., Seoul, Korea (South), **2005**.

[15] M. Mathioudakis, N. Bansal, N. Koudas, *"Identifying, Attributing and Describing Spatial Bursts"*, In the Proceedings of the VLDB Endowment, Vol. **3**. No. **1**, pp. **1091-1102**, **2010**.

[16] Y. Zhang, N. Meratnia and P. Havinga, *"Outlier Detection Techniques for Wireless Sensor Networks: A Survey"*, In the Proceedingsof the IEEE Communications Surveys & Tutorials, Vol. **12**, No. **2**, pp. **159-170**, Second Quarter **2010**.

[17] P. Yang, D. Stankevicius, V. Marozas, Z. Deng, E. Liu, A. Lukosevicius, F. Dong, L. Xu, G. Min, *"Lifelogging Data Validation Model for Internet of Things Enabled Personalized Healthcare"*, IEEE Transactions on Systems, Man and Cybernetics: Systems, Vol. **48**, No. **1**, pp. **50-64**, **2018**.

[18] M.S. Kakkasageri, S.S. Manvi, *"Information Management in Vehicular Ad hoc Networks: A Review"*, Journal of Network and Computer Applications, Vol. **39**, pp **334-350**, **2014**.

[19] M. A. Cuguero, M. Christodoulou, J. Quevedo, V. Puig, D. Garcia and M. P. Michaelides, *"Combining contaminant event diagnosis with data validation/reconstruction: Application to smart buildings"*, In the Proceedings of the 22nd Mediterranean Conference on Control and Automation, Palermo, Italy, pp. **293-298**, **2014**.

[20] Y. Chen, J. Yang and S. Jiang, *"Data validation and dynamic uncertainty estimation of self-validating sensor"*, In the Proceedings of the 2015 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Pisa, Italy, pp. **405-410**, **2015**.

[21] R. Sharifi, R. Langari, *"A Hybrid AANN-KPCA Approach to Sensor Data Validation"*, In the Proceedings of the seventh WSEAS International Conference on Applied Informatics and Communications, Athens, Greece, pp.**85-91**, **2007**.

[22] P. H. Ibarguengoytia, L. E. Sucar and S. Vadera, *"Real time intelligent sensor validation"*, In the Proceedings of the IEEE Transactions on Power Systems, Vol. **16**, No. **4**, pp. **770-775**, **2001**.

[23] J. Rivera-Mejia, E. Arzabala-Contreras and A. G. Leon-Rubio, *"Approach to the validation function of intelligent sensors based*

on error's predictors", In the Proceedings of the 2010 IEEE Instrumentation & Measurement Technology Conference Proceedings, Austin, TX, USA, pp. **1121-1125**, **2010**.

[24] B. Mounika, G. Raghu, S. Sreelekha and R. Jeyanthi, *"Neural network based data validation algorithm for pressure processes"*, In the Proceedings of the 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, India, pp. **1223-1227**, **2014**.

[25] Z. Shen and Q. Wang, *"Data validation and confidence of self-validating multifunctional sensor",Journal of* SENSORS IEEE, Taipei, Taiwan, pp. **1-4**, **2012**.

[26] Q. Wang, Z. Shen and F. Zhu, *"A multifunctional self-validating sensor"*, In the Proceedings of the 2013 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Minneapolis, MN, USA, pp. **1283-1288**, **2013**.

[27] Narander Kumar and S. Jitendra Kumar, *"Secure Data Validation and Transmission in Cloud and IoT Through Ban Logic and KP-ABE"*, International Journal of Sensors, Wireless Communications and Control.

[28] P. H. Sharanappa, M. S. Kakkasageri, *"Intelligent Information Gathering Scheme in Internet of Things (IoT)"*, In the Proceedings of the 11th International Conference on Advanced Computing (ICoAC), Department of Computer Technology, Anna University, MIT Campus, Chennai, India, pp.**136-140**, **2019**.

[29] P. M. Chanal, M. S. Kakkasageri, *"Security and Privacy in IoT: A Survey"*, Journal of Wireless Personal Communication, Springer, Vol. **115**, No. **3**, pp.**1667–1693**, **2020**

[30] P. M. Chanal, M. S. Kakkasageri, *"Hybrid Algorithm for Data Confidentiality In Internet of Things"*, In the Proceedings of the 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IIT Kanpur, India, **2019**.

[31] P. M. Chanal, M. S. Kakkasageri, *"Preserving Data Confidentiality in Internet of Things"*, Journal of SN Computer Science, Springer, Vol. **2**, No. **1**, pp. **1-12**, **2021**.

AUTHORS PROFILE

Sharanappa P. H. received his B.E. Degree in Electronics and Communication Engineering, M.Tech Degree in Digital Electronics and Communication from Visvesvaraya Technological University Belgaum, Karnataka, India. He is pursuing his Ph.D Degree in Internet of Things. He has 13 years of experience in teaching. Presently, he is working as Assistant Professor in Department of Electronics and Communication Engineering, Basaveshwar Engineering College, Bagalkot, Karnataka, India. He has published 3 papers in international conferences and 2 papers in international journals. His areas of interest are wireless networks and Internet of Things. He is a member of IETE India.

Dr. Mahabaleshwar S. Kakkasageri received his B.E. Degree from Karnataka University, M.Tech. degree in Digital Communication and Ph.D. degree from the Visvesvaraya Technological University, Belgaum, Karnataka, India. He has experience of 16 years in teaching. His research interests are Vehicular Ad hoc Networks, Wireless Networks, and Internet of Things. He has published 50 papers in national and international conferences, 25 papers in national and international journals, and 07 books/books chapters. He is a member of IEEE and IETE. He is a reviewer and programme committee member for many journals and international conferences, respectively. He received "Seed Money to Young Scientist for Research" from VGST Karnataka in 2015.

