2022 IEEE International Conference on Edge Computing and Communications (EDGE) | 978-1-6654-8140-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/EDGE55608.2022.00017

CCEI-IoT: Clustered and Cohesive Edge Intelligence in Internet of Things

Chinmaya Kumar Dehury*, Praveen Kumar Donta†, Schahram Dustdar†, and Satish Narayana Srirama‡

*Institute of Computer Science, University of Tartu, Tartu, Estonia. Email: chinmaya.dehury@ut.ee
†Distributed Systems Group, TU Wien, Vienna, Austria. Email: {pdonta,dustdar}@dsg.tuwien.ac.at
‡School of Computer and Information Sciences, University of Hyderabad, Hyderabad, India. Email: satish.srirama@uohyd.ac.in

***Abstract*—The recent advancement in edge computing allows IoT devices to be smart enough to collect the surrounding data and react to the environment based on a predefined logic set or the instructions from a remotely located control center such as cloud computing. However, the intelligence in this smart environment is primarily implemented in the central controller, allowing no or little room for IoT devices to collaborate, share knowledge, and take advantage of peer devices' knowledge with zero or minimal reliance on the central controller. This leaves the whole system not scalable and inefficient in energy consumption. In this paper, we propose an approach where edge intelligence is imposed in a clustered and cohesive manner. This enables the IoT devices to form one or more clusters based on the intelligence they possess and operate collaboratively with or without any intervention of the remote control center. This allows IoT devices to efficiently collect and react to the environment, resulting in better service quality and further reducing overall energy consumption.**

***Index Terms*—Edge Intelligence, IoT, clustered intelligence, edge computing, smart environment, sensors.**

I. Introduction

The tiny, affordable, and ability for mass production of sensors equipped with the capability to exchange the data with other sensors made it possible to establish a sophisticated internet of things (IoT) infrastructure. Dedicated sensors, such as CO_2 detector, light sensor, fire detection, proximity sensors, etc., are attached to existing electrical and electronic appliances to make them smart enough to react to any change in the surrounding environment. In general, such sensors send the environmental data through multiple gateways and computing environment to a centrally located controller (generally in a cloud environment), which in turn is responsible for implementing and ensuring the enforcement of the higher level of automation logic [1], [2] , as shown in Figure 1.

When the number of collectors, which collect/sense the surrounding data, and reactors/smart appliances, which react to the collected/sensed environmental data, increases to hundreds of thousands, it would become a cumbersome task for the central controller to manage [3]–[5]. Further, the conventional rule-based autonomous system may not be efficient enough. Edge intelligence is one of the candidate solutions that enables the IoT devices to be equipped with the required logic [6], [7]. This would allow edge devices to not only process a certain amount of data but also can share and reuse other edge devices' knowledge, enabling edge-to-edge transfer learning [8]. The recent advancements in the application of edge intelligence [5], [9]–[11] mainly focus on enabling the IoT devices intelligent enough to react to the surrounding data, based on a set of predefined rules that are confined only to the device itself. Further, such smart appliances do not consider the impact they bring after reacting to the environment. For instance, the smart bulb may not consider if there is any change in the luminosity level of the environment after it is turned ON.

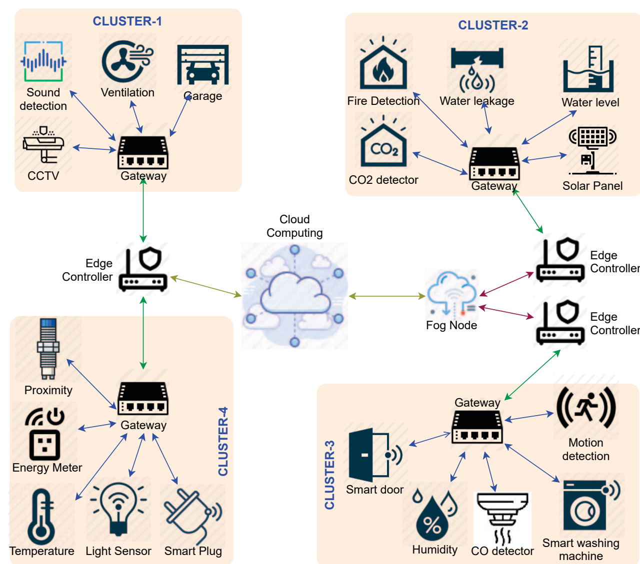

Fig. 1: An abstract view of edge-cloud environment.

Intelligence is imposed mainly on the central controller and IoT devices in a master-slave approach [12]–[14].As a result of this advancement, the smart environments are becoming more centralized and IoT devices mainly depend upon the central controller, leaving no room for collaboration among IoT devices in close proximity. This introduces several research challenges, such as higher energy consumption, higher network congestion and inefficient implementation of automation system, lesser scalability [4], [15] and inefficient resource utilization [16], [17]. Higher network activity and inefficient resource utilization lead to higher energy consump-

2767-9918/22/$31.00 ©2022 IEEE
DOI 10.1109/EDGE55608.2022.00017

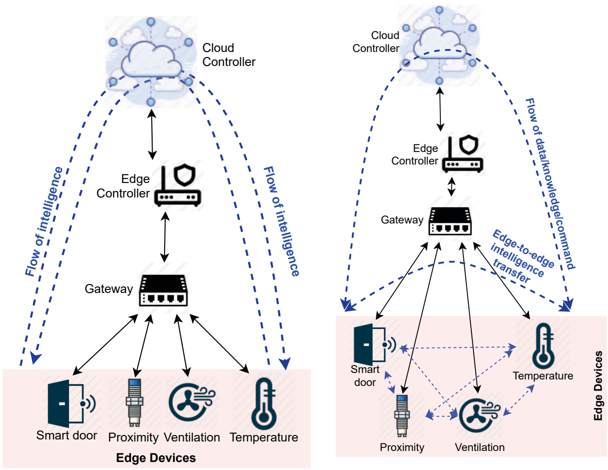

Fig. 2: An illustrative example of the motivation for device-centric and intelligence-centric clustering strategies

tion, which is the primary goal of this paper, as discussed in Section III. Recent research also focuses on the insight of the confluence of edge computing and Artificial Intelligence (AI) and discussed the need of AI for edge computing to enable the edge intelligence for several smart environments [18].

In this paper, we propose a clustered and cohesive edge intelligence that allows IoT devices and smart appliances to sense and react to the environment in a collaborative manner through the exchange of their own intelligence with or without the involvement of a central controller. This would obviously reduce the dependency of edge devices on central controller and hence reduce the network activity with the cloud. Lesser network activity further infers lesser energy consumption [19]. As shown in Figure 1, multiple IoT devices may form a cluster based on their location. Such a cluster of IoT devices would collect the environmental data and react to the environment collaboratively, improving the users' experience.

A. Motivation and Contribution

In the conventional approach, edge devices are embedded with the required intelligence to collect and process the surrounding data. While doing so, such edge devices are entirely managed and controlled by the central controller usually residing in a cloud environment, as shown in Figure 2a. Hence all data, knowledge, and control instructions or the commands go through intermediate devices such as gateways and edge controllers. This also applies to devices with the capability to communicate among other edge devices. This leads to several challenges, such as energy waste during communication, higher degrees of reliance on the central controller, and inability to take advantage of device-device communication.

To address the above-discussed challenges, the conventional approaches allow the devices to be clustered, following device-centric clustering strategies based on their location, types, and other characteristics. The device-centric clustering allows the cluster head to collect the data from other edge devices and send the collected data to the central controller. However, forming and managing the clusters become a cumbersome task when the number of devices increases significantly. In addition, managing intelligence embedded atop the devices further introduces another layer of complexity. This motivates us to shift from a device-centric clustering strategy to an intelligence-centric clustering strategy that makes the edge intelligence clustered and cohesive. This would allow the edge devices to exchange their knowledge among others without a central controller, as shown in Figure 2b.

Upon achieving the above, edge devices will be clustered primarily considering the intelligence they are equipped with, not by their type or location. The advantages of such an approach can be realized while managing a large number of edge devices. This would also reduce the reliance on the central controller as the edge devices may operate based on the knowledge of other peer edge devices and the current state of the surrounding environment. This approach would also allow the data to be with in the edge environment and make data movement through cloud optional. The following sections further discuss the proposed Clustered and Cohesive Edge Intelligence in the IoT environment (CCEI-IoT).

Based on the above motivations, the main contributions of this research can be summarized as follows:

- The current state of the arts towards clustered edge intelligence are examined and proposed a mechanism to cluster the edge intelligence with an objective to further minimize the energy consumption.
- The concerned problem on energy minimization is mathematically formulated with an objective function and the associated constraints.
- To evaluate the performance, the proposed CCEI-IoT is simulated using a lightweight Java-based discrete event simulator.
- This paper also discusses the open challenges that remain unaddressed, such as intelligence discoverability and observability, security, computation and communication overhead.

The rest of the paper is organized as follows: Section II summarizes the recent state-of-the-art on edge intelligence. Section III formulates the concerned problem and models the whole system followed by the proposed solution in Section IV. The performance evaluation results and the open challenges are discussed in Section V and VI, respectively. Section VII presents the concluding remark and summarizes the future works.

II. Related work

Edge intelligence promises to address several limitations that cloud computing posses, such as higher latency, more network congestion, reduced QoS and many more. This brings the intelligence from central controller to the edge device taking advantage of the computing and storage resource available at edge. However, some of the issues on communication overhead with central controller and the energy efficiency of edge devices remain unaddressed [20]–[22]. The recent advancements mainly focuses on the enabling the edge device to form the cluster and reduces the reliance on central controller [8], [23], [24].

One of the research dimension of the edge intelligence is how to cluster the underlined IoT devices more effective manner, as discussed in [23]. Clustering helps the IoT infrastructure at the edge of the network to minimize the energy consumption without compromising the QoS [23]. Clustering at the edge takes different parameters into account such as mobility or location of the edge device. For instance Bartoletti et al. [24] present location-based analytics in 5G network that cluster the people based on their type of mobility, such as biking, cycling, walking, etc.

Lu et al. [8] demonstrate the advantage of edge-to-edge transfer learning among edge cameras that reduces the bandwidth and storage usage. However, the proposed mechanism entirely focuses on a very specific use case and does not generalize the clustering of edge intelligence. The other limitation lies within its approach on device-centric learning collaboration. Such approach makes the scalability of collaborative learning cumbersome when the number of edge camera increases significantly. A similar approach is applied on the smoke detection in foggy surveillance environments by Khan et al. [25]. However, the proposed mechanism relies entirely on the on-board computing and storage resource and does not take advantage of idle resources available with nearby edge devices.

Many research results from the edge Intelligence perspective show that clusters of edge devices serve as the backbone for the implementation of edge intelligence as the devices within the cluster are able to share the data and knowledge around them and help the cluster-head to take a collaborative and effective decision [26]. However the heterogeneity, trust worthiness of the underlined IoT devices and their local knowledge have become one of the primary hurdles to realize maximum capacity of edge intelligence. The authors attempted to address this issue by incorporating the blockchain technology [27]. Considering the importance of an efficient and secure service management, Zhang et al. [28] proposed a cross-domain sharing inspired distributed heterogeneous edge resource scheduling in industrial IoT. The blockchain technology is applied to meet the edge resource transaction consensus requirement for the edge nodes.

III. Problem Formulation

Figure 1 shows a multi-layered IoT architecture of a smart environment. A smart environment in this paper may refer to a smart home, a smart building, a smart community, or a smart city. In general, the IoT devices at the edge are managed by the control center present in the cloud computing environment. Some of the IoT devices sense the surroundings and send the data to the control center. Based on the predefined logic, the control center commands the smart appliances to change their state. A smart environment consists of mainly two categories of IoT devices: (a) *Collector* and (b) *Reactor*.

TABLE I: List of notations

Notation	Description
C	$C = \{c_1, c_2, \dots, c_n\}$, the set of n number of collector types.
c_i^j	The collector at index j of type $c_i, 1 \leq j, 1 \leq i \leq n$.
R	$R = \{r_1, r_2, \dots r_m\}$, the set of n number of Reactor types.
r_i^j	The reactor at index j of type $r_i, 1 \leq j, 1 \leq i \leq m$.
$\ddot{C}$	$\ddot{C} = \{\dot{c}_1, \dot{c}_2, \dots, \dot{c}_q\}$, the set of q clusters.
$b(r_i^j, \dot{c}_k)$	Indicates if the reactor $r_i^j \in R$ belong to cluster $\dot{c}_k \in \ddot{C}$.
$b(c_i^j, \dot{c}_k)$	Indicates if the collector $c_i^j \in C$ belong to cluster $\dot{c}_k \in \ddot{C}$.
$S^t(I)$	$S^t(I), I \in \{C \cup R\}$ be the state of an IoT device I, at time t.
$e^t(I)$	$e^t(I), I \in \{C \cup R\}$, be the energy consumption of the IoT device I at time t.
$\dot{e}_k^t$	Energy consumption of a cluster $\dot{c}_k$ at time t.
E^t	The total energy consumption at time t.
$A(c_i^j)$	The set of dependent collectors and reactors.
$B(r_a^b)$	The set of dependent collectors in the same cluster where reactor r_a^b belongs to.

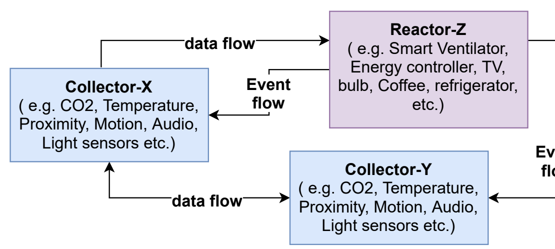

Fig. 3: Overview of data and event flow among collectors and reactors.

A. Collectors and Reactors

Collectors are the sensors that collect the surrounding data by simply sensing the environment. The collection of data is both event-driven and time-driven. The collected data are essentially sent to the other reactors and the same collection event is sent to other collectors in the same cluster. Let $C = \{c_1, c_2, \dots, c_n\}$ be the set of n number of collector types. $c_i = \{c_i^1, c_i^2, \dots\}, 1 \leq i \leq n$ be the set of collectors of same type based on the specific data to be collected.

A *Reactor* device, on the other hand, only reacts to the collected data sent by the collectors in the same cluster, e.g., a smart bulb in a room reacts (or turns ON) when a collector senses the presence of a person. In addition to this, a reactor may also receive the command from the central control center. The same reaction event is also sent to other nearby collectors and reactors in the same cluster. Let, $R = \{r_1, r_2, \dots r_m\}$ be the set of m number of reactor types, where $r_j = \{r_j^1, r_j^2, \dots\}, 1 \leq j \leq m$ be the set of reactors of same type.

B. Intelligence Clusters

A set of collectors and reactors forms a cluster. All the cluster members share their data, the related events, and reaction events among others in the same cluster. A smart environment may consist of a large number of clusters. Some of the examples of clusters can be (a) all the IoT devices in the kitchen of a smart home, (b) IoT devices in an apartment of a smart community, (c) IoT devices in the corridor of a smart building, (d) IoT devices in a shopping center of smart city, etc. Clusters are created statically, either based on the context, location, dependencies among devices, or time. A collector or reactor can be a member of multiple clusters. Let $\ddot{C} = \{\dot{c}_1, \dot{c}_2, \ldots, \dot{c}_q\}$ be the set of q clusters. The boolean functions $b(c_i^j, \dot{c}_k)$ and $b(r_i^j, \dot{c}_k)$ indicates if a collector $c_i^j \in C$ and reactor $r_i^j \in R$ belong to cluster $\dot{c}_k \in \ddot{C}$, respectively. Mathematically,

$$b(I, \dot{c}_k) = \begin{cases} 1 & \text{If the collector or the reactor } I \\ & \text{belongs to the cluster } \dot{c}_k \in \ddot{C} \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

Let $S^t(I), I \in \{C \cup R\}$ be the state of an IoT device I at time t in the smart environment, where I represents an IoT device that could be a *collector* or a *reactor*. The state of an IoT device represents the state of the surrounding environment. For instance, for a light sensor, the current luminosity level of the environment is considered as the state of this collector. However, for a reactor, the state represents the current capacity at which the reactor is working. The state of a reactor must satisfy the condition $0 < S^t(r_i^j) \leq 100\%, 0 < t, \forall r_i^j \in R$. For example, a ventilator may operate at 40% of its full capacity, which is the efficiency of the ventilator. Based on the state of each IoT device, the state of a cluster at time t, denoted by $\dot{s}_k^t = \langle S^t(I) | \forall I \in \{C \cup R\}, b(I, \dot{c}_k) = 1 \rangle$, can be defined as a tuple of states of all the devices in that cluster at time t.

The state of the IoT devices are also used to form the dependency graph. A collector c_i^j is said to be dependent on a reactor r_a^b, only when a change in the state of the reactor leads to a change in the state of the collector. Similarly, the reverse relationship holds true. In other word, a reactor is said to be dependent on a collector, only when a change in the state of the collector leads to a change in the state of the reactor. Mathematically the relationship between collector c_i^j and reactor r_a^b can be represented as

$$S^{\Delta}(r_a^b) \Leftrightarrow S^{\Delta}(c_i^j) \tag{2}$$

, where $S^{\Delta}(r_a^b) = \left| S^t(r_a^b) - S^{t-1}(r_a^b) \right|$ and $S^{\Delta}(c_i^j) = \left| S^t(c_i^j) - S^{t-1}(c_i^j) \right|$.

Definition 1. *$A(c_i^j)$ be the set of collectors and reactors that are dependent on the state of collector c_i^j in the same cluster. Mathematically,*

$$A(c_i^j) = \{I \in \{C \cup R\} | S^{\Delta}(c_i^j) \Rightarrow S^{\Delta}(I), S^{\Delta}(c_i^j) > 0, \\ S^{\Delta}(I) > 0, b(c_i^j, \dot{c}_k) = b(I, \dot{c}_k) = 1, \forall c_i^j \in C\} \tag{3}$$

Definition 2. *$B(r_a^b)$ be the set of collectors only in the same cluster, whose data changes when r_a^b reacts.*

$$B(r_a^b) = \{I \in \{C\} | S^{\Delta}(r_a^b) \Rightarrow S^{\Delta}(I), S^{\Delta}(r_a^b) > 0 \\ , S^{\Delta}(I) > 0, b(r_a^b, \dot{c}_k) = b(I, \dot{c}_k) = 1, \forall c_i^j \in C\} \tag{4}$$

C. Objective

Some of the objectives behind making the edge intelligence clustered and cohesive are minimizing the energy consumption, reducing the reliance on the central controller that resides in the cloud, reducing the network congestion between the edge and the central controller, leveraging the autonomy level of the edge device. However, in this paper, minimizing the energy consumption of the IoT devices by reducing the reliance on the central controller is considered as the primary objective towards making the edge intelligence clustered and cohesive.

Let $e^t(I), I \in \{C \cup R\}$, be the energy consumption of the IoT device I at time t, when the device is either collecting the data or reacting to the environment. Using this value, the energy consumption of a cluster $\dot{c}_k$ at any given point of time t, can be defined as

$$\dot{e}_k^t = \sum_{\forall c_i^j \in \dot{c}_k} e^t(c_i^j) + \sum_{\forall r_a^b \in \dot{c}_k} e^t(r_a^b) \tag{5}$$

In general, the collectors consume significantly less energy compared to that of the reactors and may not play a major role in minimizing the total energy of the whole smart environment, which is also discussed in the performance evaluation section. For instance, a collector that collects the surrounding temperature consumes a negligible amount of energy compared to that of a smart bulb. Let E^t denotes the total energy consumption of all the reactors, after neglecting the total energy consumption of all collectors. Considering the energy consumption of the reactors only, the objective of this research is to minimize the E^t at time t by taking a clustered and cohesive decision on changing the state of the collectors and reactors.

Objective:

$$Minimize \quad E^t = \sum_{\forall r_a^b \in R} e^t(r_a^b) \tag{6}$$

Constraints:

$$\sum_{\forall \dot{c}_k \in \ddot{C}} b(r_a^b, \dot{c}_k) \geq 1 \tag{7}$$

$$\nexists \dot{c}_k \in \ddot{C}, b(r_a^b, \dot{c}_k) * A(c_i^j) < 1, \forall c_i^j \in C, b(c_i^j, \dot{c}_k) = 1 \tag{8}$$

$$\forall \dot{c}_k \in \ddot{C}, \forall r_a^b \in R, \; if \; b(r_a^b, \dot{c}_k) = 1, B(r_a^b) \neq \phi \tag{9}$$

$$0 < S^t(r_a^b) \leq 100\%, 0 < t, \forall r_a^b \in R \tag{10}$$

The concerned objective in making edge intelligence clustered and cohesive is to minimize the overall energy consumption. However, this must satisfy the following constraints:

- Constraint (7) ensures that each reactor must belong to at least one cluster.
- It is essential to ensure that the cluster where the reactor r_a^b belongs to does contain at least one collector and must satisfy the condition in Definition 1 as given in Constraint (8)
- Similar to the above, Constraint (9), ensures that the cluster where the reactor r_a^b belongs to contains at least one collector, $c_i^j \in C$, whose data changes when the reactor r_a^b reacts to the environment.
- Constraint (10) ensures that the state of each reactor must be a valid value between 0 and 100.

IV. Clustered and Cohesive Edge Intelligence

Based on the above formulation, this section proposed Clustered and Cohesive Edge Intelligence in the IoT environment (CCEI-IoT), a solution focusing on minimizing the overall energy consumption.

Two algorithms are created, that work in two stages, to achieve the objective mentioned in Eq. (6). In the first stage, a data-event flow graph is created by considering the dependencies among the collectors and reactors, as given in Algorithm 1. In the second stage, the reactors are controlled strategically to minimize the energy, as given in Algorithm 2.

Figure 3 shows the flow of data and events among collectors and reactors. For example, the *Collector-X* (e.g. motion sensor) may send its environmental data to the *Reactor-Z*(e.g. smart bulb) and *Collector-Y*(e.g. light sensor) in the same cluster. In response, the smart bulb may be turned ON and send the event to another *Collector-Y* (i.e., light sensor). Based on the event, the light sensor may collect the new luminosity level. The change in the luminosity level would indicate if the smart bulb should remain in the ON state or OFF state. Let $\vec{R}^t(r_i^j)$ be the set of state values received by the reactor r_i^j from other collectors in the same cluster at time t.

Based on the above description, two sets $A(c_i^j)$ and $B(r_i^j)$ for all the IoT devices in each cluster are constructed, as given in Algorithm 1. Algorithm 1 starts by initializing the sets $A(c_i^j)$ and $B(r_i^j)$ as empty sets. As depicted in Figure 3, the fundamental conditions imposed while constructing the sets $A(c_i^j)$ and $B(r_i^j)$ are that a collector sends the surrounding data to other collectors and reactors in the same cluster. In contrast, a reactor sends its events to only other collectors in the same cluster. By iterating the same process over all the clusters, $A(c_i^j)$ and $B(r_i^j)$ for all the IoT devices in the smart environment can be constructed, as given in Algorithm 1.

For each collector, c_i^j in a cluster $\dot{c}_k$, the algorithm finds a set of dependent collectors and reactors in the same cluster $\dot{c}_k$. This set of devices are then merged with the current set of $A(c_i^j)$ using union operation, as in Line 3–6. Similarly, for each reactor r_i^j in the cluster $\dot{c}_k$, the algorithm finds a set of dependent collectors from the same cluster $\dot{c}_k$ and merged with the set $B(r_i^j)$, as in Line 7–10. The above-mentioned steps are iterated over all the clusters present in set $\ddot{C}$, as in Line 2–11. It is obvious that the two sets $A(c_i^j)$ and $B(r_i^j)$ may consist of devices from multiple clusters. It is assumed that the dependent set of collectors and reactors for each collector is known. Similarly, for each reactor, the set of dependent collectors is assumed to be known. Both the sets $A(c_i^j)$ and $B(r_i^j)$ are further used to minimize the overall energy consumption of the smart environment, as given in Algorithm 2.

Algorithm 1: Data-event flow graph

```
   Input: Set of clusters C̈
1  A(c_i^j) = {φ}, ∀c_i^j ∈ C ; B(r_i^j) = {φ}, ∀r_i^j ∈ R;
2  foreach Cluster ċ_k in C̈ do
3      foreach Collector c_i^j ∈ ċ_k do
4          tmp = Find the set of dependent collectors and
             reactors in the same cluster, as per the
             Definition 1.;
5          A(c_i^j) = A(c_i^j) ∪ tmp;
6      end
7      foreach Reactor r_i^j ∈ ċ_k do
8          tmp = Find the list of dependent collectors in
             the same cluster, as per the Definition 2.;
9          B(r_i^j) = B(r_i^j) ∪ tmp;
10     end
11 end
```

Algorithm 2: Energy minimization through CCEI-IoT

```
   Input: Set of clusters C̈.
1  Time t = 0;
2  Populate the sets A(c_i^j) and B(r_i^j) using Algorithm 1;
3  while TRUE do
4      foreach c_i^j ∈ ċ_k do
5          if S^t(c_i^j) ≠ S^{t-1}(c_i^j) then
6              ∀tmp ∈ A(c_i^j), S^{t+1}(tmp) ⟵ S^t(tmp);
7          end
8      end
9      foreach r_i^j ∈ ċ_k do
10         if R⃗^t(r_i^j) ≠ Φ then
11             React to the environment iff
                 ∀tmp ∈ B(r_i^j), S^t(tmp) ≠ S^{t+1}(tmp);
12             If r_i^j reacted based on above statement,
                 calculate E using Eq. (6);
13         end
14     end
15     Wait for the current time cycle t to end;
16 end
```

Algorithm 2 presents the proposed mechanism to minimize the energy consumption through a clustered and cohesive intelligence. Precisely, we try to use reactors or smart appliances only when their impact on the environment and the state of the other collectors satisfy a predefined threshold value. The predefined threshold value is assumed to be calculated by the service provider based on the context and the environment. Algorithm 2 starts by populating the sets $A(c_i^j)$ and $B(r_i^j)$ using the Algorithm 1, as in Line 2. Algorithm 1 constructs the data and event flow graph by examining the logical interconnections among the collectors and reactors, as shown in Figure 3. Upon constructing $A(c_i^j)$ and $B(r_i^j)$, each collector continuously compares the current state value with the previous one. In case of any change, the state value is forwarded to all other collectors and reactors, as in Line 6. On the other hand, upon receiving the state value from other reactors, as in Line 10, the reactor reacts to the environment as it is intended to be, *if and only if* the reaction brings an impact to the surrounding and changes the state value of any other collectors in set $B(r_i^j)$, as in Line 11. In this case, the total energy consumption will be calculated using Eq. (6), as given in Line 12. All the steps from Line 3 to 14 are continuously followed in each time cycle t, as in Line 15. By doing so, the reactors depend on the current state of the environment and depend on the future state of the environment, which minimizes the E and improves the efficiency of the whole smart environment.

In order to minimize the energy consumption and make the edge devices more autonomous, the proposed CCEI-IOT solution strategically controls the reactors. This is achieved by firstly, allowing the reactors to react to the environment only when the reaction brings a significant change and secondly, by reducing the reliance on the central controller. The proposed solution is further verified through simulation in a small-scale environment, as discussed in the next section.

V. Performance Evaluation

This section evaluates the performance of the proposed CCEI-IoT using a lightweight Java-based discrete event simulator. The performance of the CCEI-IoT is compared with the conventional automation system, where the reactors are triggered based on the predefined events or at a regular time interval. However, the proposed CCEI-IoT considers the after-react impact on the environment through the other collectors and decides whether to react. The average energy consumption of all the collectors and reactors is used as the performance matrix.

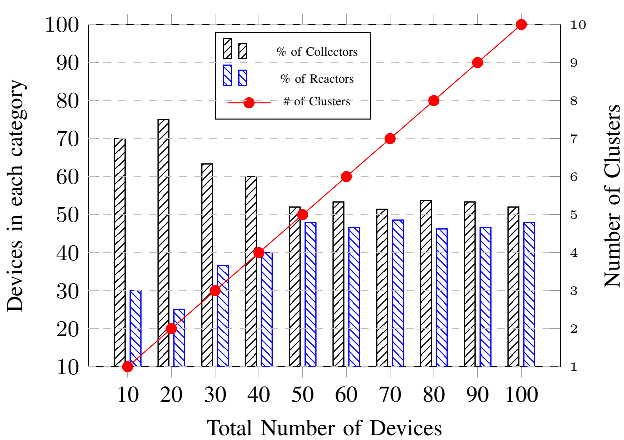

Fig. 4: Percentage of collectors and reactors.

For the performance, a total of 100 IoT devices is considered, where each 10 devices form one cluster, resulting in a total of 10 clusters. As a whole, out of 100 IoT devices, 52% IoT devices are considered as collectors and 48% IoT devices are considered as reactors, as shown in Figure 4. In each cluster, the number of collectors is more than the number of reactors. For instance, from the 30 IoT devices, 63.3% devices are collectors and 36.6% of the devices are reactors.

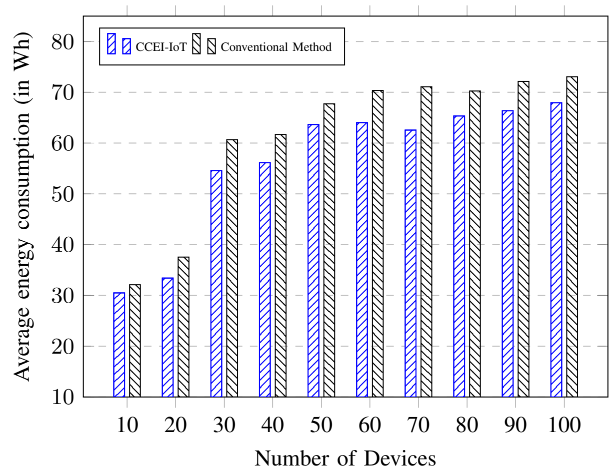

Fig. 5: Average energy consumption of IoT devices.

With the above set of collectors and reactors, the average energy consumption is calculated using both the conventional method and the proposed CCEI-IoT method, as shown in Figure 5. The unit of the energy consumption is in Watt-hour (Wh). It is assumed that the average energy consumption of a reactor is significantly higher than that of a collector. In the simulated environment, the minimum and maximum energy consumption of a collector is set to 10^{-3} *Wh* and 10^{-2} *Wh*, respectively. On the other hand, the minimum and maximum energy consumption of a reactor are set to 10 *Wh* and 300 *Wh*, respectively. The energy consumption of each IoT device is randomly assigned. With the above configuration, it is observed that the average energy consumption for the devices in cluster 1 or the first 10 IoT devices is 32.1 *Wh* using conventional method and 30.49 *Wh* in the case of the proposed CCEI-IoT method. When the number of IoT devices increases to 100, the average energy consumption is approximately 67.9 *Wh*, which is approximately 7% less than the average energy consumption using the conventional method. The improvement is due to the fact that each reactor not only depends on the time or the event from other collectors but also depends on if their reaction brings a visible impact to the environment.

A further in-depth behavior related to energy consumption can be observed by comparing the percentage of energy consumed by collectors and reactors, as shown in Figure 6. For a total of 10 IoT devices, the total energy consumption is observed to be 321 *Wh*, of which more than 90% energy consumed by only the reactors and less than 3% of the total energy consumption is made by the collectors only. When the

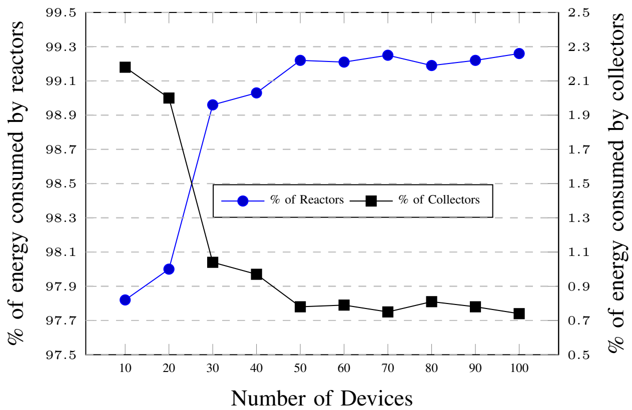

Fig. 6: Percentage of energy consumption by collectors and reactors.

number of IoT devices increases to 100, the reactors consumed more than 99% of total energy consumption. This shows an almost negligible impact by the collectors on the total energy consumption of the whole smart environment.

For further confirmation on the differences, the percentage of energy consumption by collectors and reactors in each cluster is observed, as shown in Figure 7(a). To support this, Figure 7(b), shows the device distribution of all the clusters. The number of devices varies between $7-13$, a total of 100 devices in all clusters. It is observed from Figure 7(a), energy consumption by the reactors ranges between $97\%-99.7\%$, whereas the energy consumption by collectors ranges between $0.3\%-2\%$ for each cluster. This confirms the fact that the reactors primarily consume a major portion of the energy.

VI. Challenges in CCEI-IoT

Despite numerous advantages, making the edge intelligence clustered attracts several challenges, including intelligence discoverability, observability, handling dynamic environment, etc. These challenges are summarized along with the future research directions as shown below.

- *Intelligence discoverability and observability*: The intelligence discoverability enables the devices to be able to find or locate the newly added intelligence or the upgraded intelligence in the edge infrastructure. The intelligence discoverability is inspired by the existing concept of semantic web service discovery, device discovery in IoT. On the other hand, another key challenge is the ability to observe a specific intelligence. Observability allows other devices to deduce the current state of a particular intelligence, such as its availability and reachability.
- *Dynamic environment*: In such an environment, several reactors and the collector can be mobile in nature. In such a highly dynamic environment, it is crucial to ensure that the reactors and collectors exchange their knowledge and data on the surrounding environment on a real-time basis. This dynamic environment also has a great impact on ensuring device-to-device communication.
- *Computation overhead*: By implementing a clustered edge intelligence, another layer of computation overhead atop the edge intelligence is introduced. The edge devices are further responsible for processing the cluster information at a regular time interval. Each edge device needs to update the database of other cluster members. This introduces a computation overhead atop the edge infrastructure, which needs to be minimized to realize the benefits of clustered edge intelligence.
- *Communication overhead*: The edge devices need to constantly share the surrounding data and their knowledge with peers, which attracts communication overhead on the network. Such communications should be strategically minimized so that the overall energy consumption for a particular reactor is minimum.
- *Security*: When the edge devices communicate with only a central controller, it is relatively easy to ensure the security of the knowledge and data. However, when an edge device listens to multiple edge devices and reacts to their data/knowledge, it is essential to establish and maintain secure communication among those devices. It is a highly challenging task for any reactor to validate the data source and ensure that the source device is not an intruder [5].

VII. Conclusions and Future Works

In this paper, we investigate conventional edge intelligence where IoT devices mostly depend on remotely located central controller and do not take advantage of the intelligence obtained from the IoT devices' data in proximity. Accordingly, a clustered and cohesive edge intelligence for an internet of things environment (CCEI-IoT) is proposed. The IoT devices in the proximity form a cluster and share the surrounding data and events among others in the same cluster. With such capability, a smart appliance or a reactor not only gets activated based on a specific event but also considers if its reaction brings a visible impact to the environment through other collectors/sensors. While doing so, all IoT devices' overall energy consumption is observed through a small-scale simulation.

The advantage can be realized with the large-scale implementation in a smart environment, which is one of the future works. Further, from this large-scale implementation, the energy consumption due to high processing time in edge device, efficiency of IoT devices, and users experience of the whole system can be observed.

References

[1] X. Zhang, Y. Wang, S. Lu, L. Liu, W. Shi *et al.*, "Openei: An open framework for edge intelligence," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 1840–1851.

[2] C. K. Dehury and S. N. Srirama, "Personalized service delivery using reinforcement learning in fog and cloud environment," in *Proceedings of the 21st International Conference on Information Integration and Web-Based Applications & Services*, ser. iiWAS2019. Association for Computing Machinery, 2019, p. 522–529.

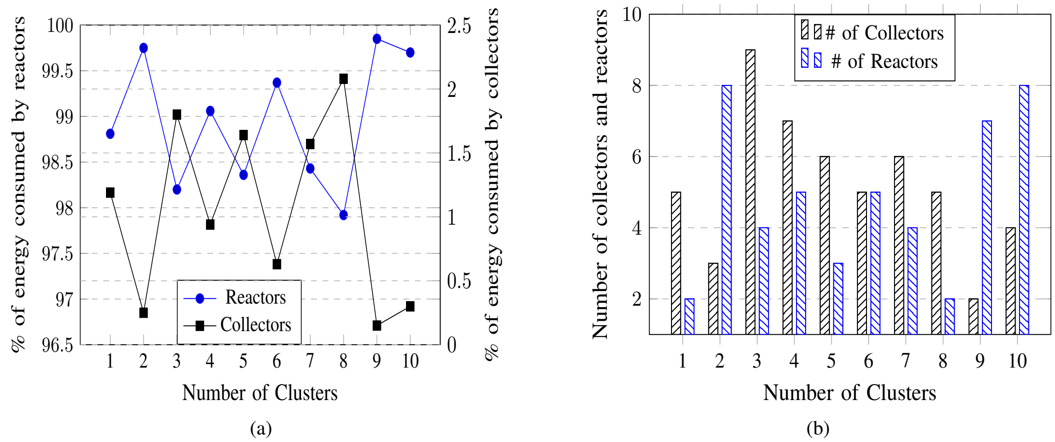

Fig. 7: Cluster-wise (a) energy consumption (b) device distribution

[3] M. J. Baucas and P. Spachos, "Using cloud and fog computing for large scale iot-based urban sound classification," *Simulation Modelling Practice and Theory*, vol. 101, p. 102013, 2020.

[4] Z. Zhao, G. Min, W. Gao, Y. Wu, H. Duan, and Q. Ni, "Deploying edge computing nodes for large-scale iot: A diversity aware approach," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3606–3614, 2018.

[5] C. K. Dehury, S. N. Srirama, P. K. Donta, and S. Dustdar, "Securing clustered edge intelligence with blockchain," *IEEE Consumer Electronics Magazine*, 2022 *(Accepted)*.

[6] Y. Zhang, X. Ma, J. Zhang, M. S. Hossain, G. Muhammad, and S. U. Amin, "Edge intelligence in the cognitive internet of things: Improving sensitivity and interactivity," *IEEE Network*, vol. 33, no. 3, pp. 58–64, 2019.

[7] C. Avasalcai, B. Zarrin, and S. Dustdar, "Edgeflow-developing and deploying latency-sensitive iot edge applications," *IEEE Internet of Things Journal*, 2021.

[8] C.-H. Lu and X.-Z. Lin, "Toward Direct Edge-to-Edge Transfer Learning for IoT-Enabled Edge Cameras," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4931–4943, Mar. 2021.

[9] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, 2019.

[10] M. Ali, A. Anjum, M. U. Yaseen, A. R. Zamani, D. Balouek-Thomert, O. Rana, and M. Parashar, "Edge enhanced deep learning system for large-scale video stream analytics," in *2018 IEEE 2nd International Conference on Fog and Edge Computing (ICFEC)*, 2018, pp. 1–10.

[11] T. Kasidakis, G. Polychronis, M. Koutsoubelias, and S. Lalis, "Reducing the mission time of drone applications through location-aware edge computing," in *2021 IEEE 5th International Conference on Fog and Edge Computing (ICFEC)*, 2021, pp. 45–52.

[12] M. A. Zamora-Izquierdo, J. Santa, J. A. Martínez, V. Martínez, and A. F. Skarmeta, "Smart farming iot platform based on edge and cloud computing," *Biosystems engineering*, vol. 177, pp. 4–17, 2019.

[13] Y. Xiao, G. Shi, Y. Li, W. Saad, and H. V. Poor, "Toward self-learning edge intelligence in 6g," *IEEE Communications Magazine*, vol. 58, no. 12, pp. 34–40, 2020.

[14] C. Kumar Dehury and S. Narayana Srirama, "An efficient service dispersal mechanism for fog and cloud computing using deep reinforcement learning," in *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, 2020, pp. 589–598.

[15] G. Ortiz, M. Zouai, O. Kazar, A. Garcia-de Prado, and J. Boubeta-Puig, "Atmosphere: Context and situational-aware collaborative iot architecture for edge-fog-cloud computing," *Computer Standards & Interfaces*, vol. 79, p. 103550, 2022.

[16] W. Y. B. Lim, J. S. Ng, Z. Xiong, J. Jin, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "Decentralized Edge Intelligence: A Dynamic Resource Allocation Framework for Hierarchical Federated Learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 536–550, Mar. 2022.

[17] Z. Lin, S. Bi, and Y.-J. A. Zhang, "Optimizing AI Service Placement and Resource Allocation in Mobile Edge Intelligence Systems," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2021.

[18] B. Parekh and K. Amin, "Edge Intelligence: A Robust Reinforcement of Edge Computing and Artificial Intelligence," in *Innovations in Information and Communication Technologies (IICT-2020)*, ser. Advances in Science, Technology & Innovation. Cham: Springer International Publishing, 2021, pp. 461–468.

[19] C. Chang, A. Hadachi, and S. N. Srirama, "Adaptive edge process migration for iot in heterogeneous fog and edge computing environments," *International Journal of Mobile Computing and Multimedia Communications (IJMCMC)*, vol. 11, no. 3, pp. 1–21, 2020.

[20] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738–1762, Aug. 2019.

[21] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, "Edge Intelligence: The Confluence of Edge Computing and Artificial Intelligence," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, Aug. 2020.

[22] Y. Zhang, H. Huang, L.-X. Yang, Y. Xiang, and M. Li, "Serious Challenges and Potential Solutions for the Industrial Internet of Things with Edge Intelligence," *IEEE Network*, vol. 33, no. 5, pp. 41–45, Sep. 2019.

[23] L. Song, K. K. Chai, Y. Chen, J. Schormans, J. Loo, and A. Vinel, "QoS-Aware Energy-Efficient Cooperative Scheme for Cluster-Based IoT Systems," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1447–1455, Sep. 2017.

[24] S. Bartoletti, L. Chiaraviglio, S. Fortes, T. E. Kennouche, G. Solmaz, G. Bernini, D. Giustiniano, J. Widmer, R. Barco, G. Siracusano, A. Conti, and N. B. Melazzi, "Location-Based Analytics in 5G and Beyond," *IEEE Communications Magazine*, vol. 59, no. 7, pp. 38–43, Jul. 2021.

[25] K. Muhammad, S. Khan, V. Palade, I. Mehmood, and V. H. C. de Albuquerque, "Edge Intelligence-Assisted Smoke Detection in Foggy Surveillance Environments," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1067–1075, Feb. 2020.

[26] T. Rausch and S. Dustdar, "Edge Intelligence: The Convergence of Humans, Things, and AI," in *2019 IEEE International Conference on Cloud Engineering (IC2E)*, Jun. 2019, pp. 86–96.

[27] C. Qiu, X. Wang, H. Yao, Z. Xiong, F. R. Yu, and V. C. M. Leung, "Bring Intelligence among Edges: A Blockchain-Assisted Edge Intelligence Approach," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, Dec. 2020, pp. 1–6, iSSN: 2576-6813.

[28] K. Zhang, Y. Zhu, S. Maharjan, and Y. Zhang, "Edge Intelligence and Blockchain Empowered 5G Beyond for the Industrial Internet of Things," *IEEE Network*, vol. 33, no. 5, pp. 12–19, Sep. 2019.