Received: 7 January 2020 | Revised: 17 June 2020 | Accepted: 30 July 2020

DOI: 10.4218/etrij.2020-0009

ORIGINAL ARTICLE

ETRI Journal WILEY

Model of dynamic clustering-based energy-efficient data filtering for mobile RFID networks

Viet Minh Nhat Vo | **Van Hoa Le**

Hue University, Hue, Vietnam

Correspondence
Viet Minh Nhat Vo, Hue University, Hue, Vietnam.
Email: vvmnhat@hueuni.edu.vn

Data filtering is an essential task for improving the energy efficiency of radio-frequency identification (RFID) networks. Among various energy-efficient approaches, clustering-based data filtering is considered to be the most effective solution because data from cluster members can be filtered at cluster heads before being sent to base stations. However, this approach quickly depletes the energy of cluster heads. Furthermore, most previous studies have assumed that readers are fixed and interrogate mobile tags in a workspace. However, there are several applications in which readers are mobile and interrogate fixed tags in a specific area. This article proposes a model for dynamic clustering-based data filtering (DCDF) in mobile RFID networks, where mobile readers are re-clustered periodically and the cluster head role is rotated among the members of each cluster. Simulation results show that DCDF is effective in terms of balancing energy consumption among readers and prolonging the lifetime of the mobile RFID networks.

KEYWORDS
Data filtering, dynamic clustering, energy efficiency, mobile RFID networks

1 | INTRODUCTION

The integration of radio-frequency identification (RFID) and wireless sensor networks (WSN) is a growing trend because this integrated model has a broad and diverse range of applications where the advantages of both technologies can be exploited and applied [1]. In the real world, RFID technologies have been adopted in many industrial applications, such as supply chain management, highway tolls, traffic management, and smart home development [2]. WSNs have traditionally been deployed in harsh environments to collect data [3]. Therefore, the integration of RFID and WSNs is an optimal solution in which both technologies complement and support each other.

The RFID and WSN integration model has generated an advantageous infrastructure consisting of a network of RFID readers, which is known as an RFID network. An RFID network consists of readers (reader-integrated nodes or simply "nodes"), tags, and a base station. A tag consists of a chip and an antenna mounted on a target object from which data are read. A reader scans tags to collect data and then transmits the collected data to a base station. RFID networks are suitable for processing and distributing data in dynamic and hierarchical environments, but they also face many challenges in terms of real-time performance, energy efficiency, redundant data, anti-collision, and authentication efficiency. Among these challenges, filtering data for both data cleaning and energy efficiency is a key issue in terms of resource utilization and energy consumption [4]. Redundant data are typically created by reading tags in overlapping coverage areas, where copies of a single

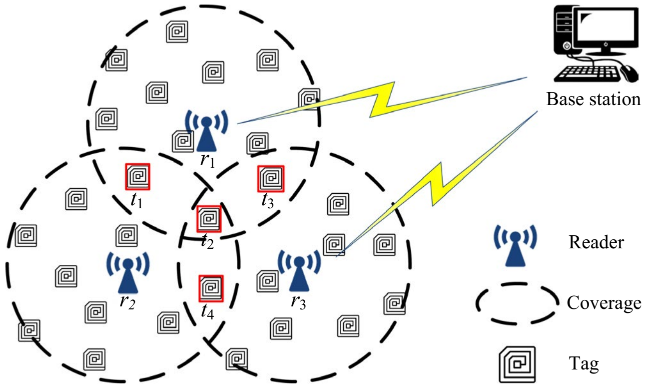

FIGURE 1 Example of overlapping reader coverage in an RFID network

object are created by multiple readers reading the same tag. As shown in Figure 1, in an RFID network of three readers, the same t_2 tag is covered by all three readers r_1, r_2, and r_3. Therefore, the tag t_2 is interrogated by the three readers and the same data from tag t_2 are transmitted to the base station by all three readers. Therefore, data duplication occurs at the base station. Removing this redundancy is necessary because it does not provide any useful information.

Redundant data reduction contributes to the efficient use of resources in an RFID network. Most proposals for data filtering focus on clustering approaches to limit the filtering of points, meaning only cluster heads are responsible for data filtering [4]. However, this approach quickly depletes the energy of cluster heads quickly and reduces the lifetime of RFID networks. Furthermore, most previous studies have assumed that readers are fixed and interrogate mobile tags in a workspace. However, there are several applications in which readers are mobile and interrogate fixed tags in a specific area. An RFID reader can be mounted on a human body or integrated into a personal terminal, such as a smart phone or tablet. Such mobile devices can then provide mobile RFID reader services for accessing tags and retrieving information [5].

This article considers data filtering in a mobile RFID network, where mobile RFID readers move arbitrarily in an interrogation area to retrieve information from static tags. The main contributions of this paper are threefold. First, a dynamic clustering algorithm is proposed to re-cluster readers periodically. Re-clustering is performed because mobile readers can change clusters as they move away from their original location. Second, a solution of cluster head rotation (CHR) is proposed to balance the energy consumption of readers. The criteria for selecting a cluster head are the highest residual energy and shortest distance to a base station. Finally, combining these proposals yields the dynamic clustering-based energy-efficient data filtering (DCDF) algorithm.

The remainder of this paper is organized as follows. Section 2 summarizes and analyzes related works. Based on unresolved issues, Section 3 presents the proposed DCDF algorithm, which includes a K-means-based clustering (KMC) algorithm (Section 3.2), dynamic reader clustering (DRC) algorithm (Section 3.3), and CHR algorithm (Section 3.4). Simulations and analysis are presented in Section 4. Finally, our conclusions are summarized in Section 5.

2 | RELATED WORKS

Clustering has been studied extensively in WSNs with a focus on energy conservation [6]. Studies related to clustering-based data filtering in RFID networks have the same objective. Choi and Park [7] proposed an in-network phased filtering mechanism (INPFM) that filters out global redundant data at nodes along routing paths. Because the INPFM is based on multi-hop routing, it checks for redundant data at all nodes until a redundancy is detected. However, this leads to a high cost of processing and transmitting, which results in rapid energy depletion for nodes.

Kim and others [8] introduced a cluster-based in-network filtering (CLIF) scheme in which readers are clustered. A cluster member is assigned the role of the cluster head, and other members of the same cluster send data to the cluster head. Therefore, a cluster head collects data from its cluster members, filters the collected data, and transmits the filtered data through a routing path to a base station. In this method, the data reading area is divided into intra- and inter-cluster regions. As shown in Figure 2, a clustered RFID network consists of cluster heads that establish two transmission paths from CH1 and CH2 to the base station through CH7. Each cluster is represented by the coverage of its cluster head, where the readers in the reading area of a cluster head belong to that cluster. The coverage radius represents the residual energy of the readers, meaning a selected cluster head is the reader with the greatest reading area. Considering the intra-cluster redundancy generated by readers in the same cluster (eg, tag t1 in the overlapping area of the cluster members of CH1), CLIF reduces processing and transmission costs by only sending data from cluster members to the corresponding cluster heads. In the case of inter-cluster redundancy, which originates from tags in the common overlapping coverage areas of neighboring clusters (eg, tag t2 in the overlapping area between cluster members of CH1 and CH2), CLIF uses the same method as INPFM, meaning it is subject to high processing and transmission costs.

The energy-efficient in-network RFID data filtering scheme (EIFS) [9] improves upon CLIF by using feedback messages. If an intermediate cluster head along a routing path detects inter-cluster redundancy, then a feedback message is sent back to the associated cluster heads. Then, one of the associated cluster heads is selected to filter inter-cluster redundant data and the other cluster heads forward inter-cluster data through the selected cluster head by altering their routing paths. As shown in Figure 2, inter-cluster

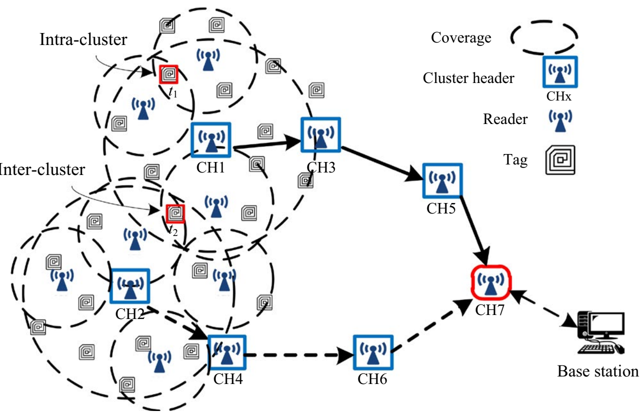

FIGURE 2 Examples of intra- and inter-cluster redundancy

redundancy is detected at CH7. A feedback message is generated by CH7 and sent back to CH1 and CH2. One of these cluster heads (eg, CH2) is assigned to filter redundant data. This approach allows EIFS to avoid unnecessary comparisons and eliminate inter-cluster redundant data transmissions from a source cluster head to an intermediate cluster head, which detects redundant data. EIFS has been demonstrated to be more effective than INPFM and CLIF in terms of the energy required for communication and computation.

An improvement of EIFS is in-network RFID duplicate data filtering (IRDF) [10], where intra-cluster filtering is conducted using EIFS, but inter-cluster filtering is performed at neighboring cluster heads, instead of at intermediate cluster heads. In this manner, IRDF eliminates feedback messages, which increase the delay of data transmission. IRDF has been proven to be the best method in terms of energy consumption among existing clustering-based data filtering methods [10].

Another approach similar to IRDF is two-phase aggregation (2PA) [11], where inter-cluster redundancy is handled by neighboring cluster heads. Specifically, neighboring cluster heads frequently exchange messages with each other to detect inter-cluster redundancy and eliminate detected redundant data prior to transmitting data to a base station. As shown in Figure 2, CH1 exchanges messages with CH2 to detect inter-cluster redundancy. One of these two cluster heads (eg, CH2) is then assigned to filter out detected redundant data. 2PA also eliminates inter-cluster redundancy, but increases the cost of exchanging messages among neighboring cluster heads.

All of the methods discussed above target fixed RFID networks, where readers are fixed and interrogate mobile tags. However, there are several applications in which readers are mobile and interrogate fixed tags in a specific area. In such applications, readers are mounted on human bodies to monitor movement [12] or attached to smartphones or tablets to query information regarding items/goods [5].

There have also been studies on mobile RFID networks, such as constructing a hierarchical wireless master-slave RFID reader architecture for smart home service systems [13], proposing localization algorithms for tracking humans [12] or objects [14] in specific areas, suggesting anti-collision solutions for interrogation between readers and tags in high-density areas [15,16], and ensuring the security of mobile RFID systems based on authentication [17]. However, there have been very few studies related to data cleaning, particularly clustering-based data filtering in mobile RFID networks. Therefore, additional research in this area is required.

In summary, the following three issues need to be resolved:

- Previous studies on clustering-based data filtering have only considered fixed RFID networks, where readers are assumed to be fixed. However, readers can be mobile in many applications. Therefore, dynamic clustering is necessary.
- The number of readers per cluster has a significant impact on the lifetime of a cluster. Therefore, balancing the number of readers between clusters is necessary for improving the lifetime of an entire network.
- The highest residual energy is typically the primary criterion for selecting a cluster head among cluster members, but the distance to a base station also has a significant effect on the energy consumption of a cluster head.

Our proposal for dynamic clustering-based energy-efficient data clustering will resolve these issues.

3 | MODEL FOR DYNAMIC CLUSTERING-BASED ENERGY-EFFICIENT DATA FILTERING

3.1 | Network model

Consider a rectangular workspace in which an RFID network is deployed such that mobile readers can cover all tags. The following conditions are assumed in this paper:

- Readers have an equal reading radius that does not decrease, regardless of their energy reduction over time.
- Readers are mobile within the workspace, but do not leave the area.
- Readers have the same initial energy, and their batteries cannot be replaced frequently.
- Tags are fixed and evenly distributed in the workspace, meaning the numbers of tags within the interrogation area of each reader are approximately equal.
- There is only one base station in the workspace. This station receives data transmitted from readers, is responsible for initial clustering and re-clustering when there is a significant change in the position or number of readers in each

cluster, and is required for CHR.

Our main goal is to prolong the lifetime of network by reducing the energy consumption of readers for various operations, such as receiving, processing (eg, filtering), and transmitting data.

Reducing the energy consumption of an RFID network can be achieved by clustering, where readers are clustered and only cluster heads filter data from their cluster members prior to sending the data to a base station. In our data filtering model, the filtering of intra-cluster redundant data is performed by cluster heads, while inter-cluster data filtering is performed using the same mechanism as IRDF [10]. The goal of data filtering at cluster heads is to reduce the energy consumption of cluster members, but this rapidly depletes the energy of cluster heads. One solution to this problem is to rotate the cluster head role, where the cluster member with the highest residual energy and shortest distance to a base station is selected for the role of a cluster head (Section 3.4).

Initial clustering of the RFID network is performed based on the K-means algorithm (Section 3.2), where readers are distributed into clusters such that the numbers of readers in each cluster are approximately equal. A threshold $\overline{n} + \Delta n$ is defined to determine if a reader can join a cluster, where $\overline{n}$ is the average number of readers per cluster and Δn is the acceptable deviation. This threshold is also used for the dynamic clustering of mobile readers (Section 3.3) if they move out of their current clusters.

3.2 | Energy model

The energy consumed by a cluster member is only used for transmitting data read from its tags, but the energy consumed by a cluster head is used for receiving data from its cluster members, processing collected data, and transmitting filtered data to base stations. Therefore, we propose the energy model shown in Figure 3, where the energy consumptions for receiving (E_{Rx}), processing (E_{Px}), and transmitting (E_{Tx}) m bits data over distance d are defined as

$$E_{\mathrm{Rx}} = m \times E_{\mathrm{elec}}, \tag{1}$$

$$E_{\mathrm{Px}} = m \times E_{\mathrm{elec}}, \tag{2}$$

$$E_{\mathrm{Tx}} = \begin{cases} m \times (E_{\mathrm{elec}} + \varepsilon_{\mathrm{fs}} d^2) & \text{if } d < d_0 \\ m \times (E_{\mathrm{elec}} + \varepsilon_{\mathrm{mp}} d^4) & \text{if } d \geq d_0 \end{cases}. \tag{3}$$

Respectively, where E_{elec} is the fixed energy value spent for receiving, processing, and transmitting 1 bit data. $\varepsilon_{\mathrm{fs}}$ and $\varepsilon_{\mathrm{mp}}$ denote the amplification coefficient for the free space model and multipath fading model, respectively. Additionally,

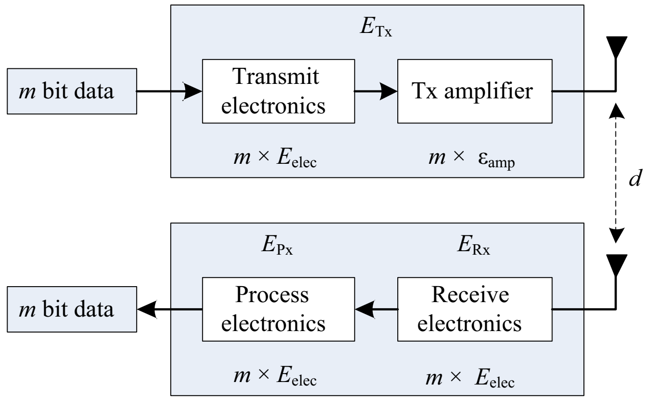

FIGURE 3 Energy model

$$d_0 = \sqrt{\frac{\varepsilon_{\mathrm{fs}}^2}{\varepsilon_{\mathrm{mp}}}}. \tag{4}$$

When the data transmission distance is greater than the threshold d_0, the energy consumption increases sharply, so the maximum communication radius for common readers is set to d_0.

The re-clustering and CHR that will be discussed below also consume energy. We assume that the energy consumed by these activities is shared equally among all readers. Therefore, each reader consumes an equal amount of energy for each activity (E_{Ax}). For simplicity, the value of E_{Ax} is defined as a constant.

3.3 | Initial clustering

The number of clusters in an RFID network plays an important role in reducing communication distances. Jin and others [18] demonstrated that the number of cluster heads is approximately 10% of the total number of nodes. Based on a predefined number of clusters (K), initial clustering is performed using the K-means algorithm. Therefore, we call this clustering process the KMC algorithm. The KMC algorithm consists of the following three steps:

Step 1: Randomly select K readers for cluster head roles. (Lines 2 to 7).
Step 2: Distribute readers to clusters. A reader (r_i) is distributed to a cluster (C_k) if the distance from r_i to the cluster head of C_k (cc_k) is the smallest and the number of readers in C_k does not exceed the threshold $\overline{n} + \Delta n$. (Lines 9 to 20).
Step 3: Select cluster heads. Based on the locations of the cluster members in a cluster, the member in the most central location is selected to be a cluster head. (Lines 22 to 24).
Step 4: Repeat Step 2 until there is no change in cluster heads.

Algorithm 1: KMC algorithm	
Input:	- set of readers $R = \{r_i \mid i = 1, 2, \ldots, N\}$; - number of clusters K;
Output:	- set of clusters $C = \{C_k \mid k = 1, 2, \ldots, K\}$, //where $C_k = \{cc_k, r_j \mid j = 1, 2, \ldots, \lvert C_k \rvert\}$
Process:	
1	$C := \varnothing$;
2	// randomly select K readers for cluster head roles **for** $k := 1$ **to** K **do**
3	$i :=$ random$(1, N)$;
4	$cc_k := r_i$;
5	$C_k := \{r_i\}$;
6	$C := C \cup C_k$;
7	**end**
8	**while** (no change for cluster heads) **do**
9	**for** $i := 1$ **to** N **do**
10	cur := cluster(r_i); // get the cluster of r_j
11	new := 1; min := ∞;
12	**for** $k := 1$ **to** K **do**// seek the nearest cluster for r_i
13	**if** (distance(r_i, cc_k) < min) **&&** ($\lvert C_k \rvert < \overline{n} + \Delta n$) **then**
14	min := distance(r_i, cc_k);
15	new := k;
16	**end**
17	**end**
18	$C_{cur} := C_{cur} \setminus \{r_i\}$; // remove r_j from its current cluster
19	$C_{new} := C_{new} \cup \{r_i\}$;// add r_j into a new cluster
20	**end**
21	**end**
22	**for** $k := 1$ **to** K **do**
23	$cc_k :=$ centralPosition(C_k); //cc_k is the reader in center of C_k
24	**end**

The complexity of the KMC algorithm mainly depends on the distribution of readers into clusters (Lines 9 to 20), which has a complexity of O($K \times N$). This operation is repeated until the selected cluster heads become stable. Therefore, the complexity of the KMC algorithm is O($K \times N$) for one loop.

3.4 | DRC

The re-clustering of mobile readers can be performed periodically or when there is a significant change in the locations and numbers of readers in each cluster. In the first case, readers are re-clustered after a fixed period (eg, ΔT). This approach is simple, but re-clustering may not be necessary because readers may not have moved significantly in the last period. Therefore, it is necessary to accurately estimate the value of ΔT after which there are likely to be significant changes in the locations and numbers of readers in each cluster. This value of ΔT depends on the direction and speed of readers, as well as the radius of clusters. In the case of re-clustering when a reader leaves its cluster, the cluster head must send a notification to the base station to request re-clustering. This approach avoids unnecessary re-clustering, but incurs additional costs (eg, consumed energy and throughput) for exchanges between cluster heads and base stations. In this study, periodic dynamic clustering was selected based on its simplicity.

The steps of the DRC algorithm can be summarized as follows:

Step 1: Identify out-of-cluster readers. For a set of N readers, denoted as $R = \{r_i \mid i = 1, 2, \ldots, N\}$, that move in a rectangular workspace, each reader maintains its coordinates. During each period ΔT, some readers (r_j) may move out of their clusters (ie, distance$(r_j, cc_k) > th_{radius}$, where cc_k is the cluster head of r_j and th_{radius} is a given coverage radius threshold). These readers are added to a set of out-of-cluster readers $R_{out} \subseteq R$ and must be re-clustered. (Lines 2 to 8)

Step 2: Re-cluster out-of-cluster readers. For each reader r_j in R_{out}, the distances from r_j to the associated clusters (the clusters that r_j leaves and joins) are calculated. Reader r_j is assigned to cluster C_k if the distance from r_j to the cluster head of C_k (cc_k) is the shortest, but this does not increase the number of readers in C_k, meaning $\lvert C_k \rvert \le \overline{n} + \Delta n$. (Lines 9 to 21)

Algorithm 2: DRC algorithm	
Input:	- set of N readers $R = \{r_i \mid i = 1, 2, \ldots, N\}$; - set of K clusters $C = \{C_k \mid k = 1, 2, \ldots, K\}$; // where $C_k = \{cc_k, r_j \mid j = 1, 2, \ldots, \lvert C_k \rvert\}$;
Output:	- K clusters after re-clustering C;
Process:	
1	$R_{out} := \varnothing$;
2	**for** $k := 1$ **to** K **do**
3	**for each** $r_j \in C_k$ **do**
4	**if** distance$(r_j, cc_k) > \mathrm{th}_{radius}$ **then**
5	$R_{out} := R_{out} \cup \{r_j\}$;
6	**end**
7	**end**
8	**end**
9	**for each** $r_j \in R_{out}$ **do**
10	cur := cluster(r_j);// the current cluster of r_j
11	new := 1; min := ∞;
12	**for** $k := 1$ **to** K **do**// seek the nearest cluster for r_j
13	**if** (distance(r_j, cc_k) < min) **&&** ($\lvert C_k \rvert < \overline{n} + \Delta n$) **then**
14	min := distance(r_j, cc_k);
15	new := k;

16	**end**
17	**end**
18	$C_{\mathrm{cur}} := C_{\mathrm{cur}} \setminus \{r_j\}$; // remove r_j from its current cluster
19	$C_{\mathrm{new}} := C_{\mathrm{new}} \cup \{r_j\}$;// add r_j to a new cluster
20	**end**

The complexity of the DRC algorithm depends on how many readers move out of their clusters (Lines 2 to 8). Such readers incur a complexity of O($K \times |C_k|$) and are added to new clusters (Lines 9 to 20) with a complexity of O($K \times |R_{\mathrm{out}}|$). These two actions are independent, meaning the complexity of the DRC algorithm is O($K \times \max(|R_{\mathrm{out}}|, |C_k|)$).

3.5 | CHR

Cluster heads quickly deplete their energy because they constantly receive, filter, and transmit data. Therefore, the role of cluster head should rotate among cluster members to prolong network lifetime. The CHR algorithm is proposed to seek the best new cluster head among the members of a cluster. The criteria for selecting a cluster head are the highest residual energy and shortest distance to the base station. Therefore, the objective function (F_{CH}) is formulated as follows:

$$F_{\mathrm{CH}} = \min \alpha \times \frac{E_{\mathrm{total}}}{E_j} + (1 - \alpha) \times \frac{d_j}{\bar{d}}, \tag{5}$$

where E_j and d_j are the residual energy and distance to the base station, respectively, for cluster member j.$\bar{d}$ is the average distance from cluster members to the base station and α ($0 \le \alpha \le 1$) is a weighting factor.

It should be noted that the residual energy of the selected cluster head, denoted as E_j, must be greater than the total consumed energy (E_{total}) for receiving, filtering, and transmitting data from cluster members in a round, meaning $E_j > E_{\mathrm{total}}$.

Algorithm 3: CHR algorithm	
Input:	- $C_k = \{cc_k, r_j \mid j = 1, 2, \ldots, |C_k|\}$;
Output:	- C_k with new cluster head;
Process:	
1	min := ∞;
2	$\bar{d}$:= average distance from members in C_k to the base station;
3	E_{total} := total energy consumed by members in C_k;
4	**for** j := 1 **to** $|C_k|$ **do**
5	$F_{\mathrm{CH}} := \alpha \times \frac{E_{\mathrm{total}}}{E_j} + (1 - \alpha) \times \frac{d_j}{\bar{d}}$;
6	**if** ((cur < min) **&&** ($E_j > E_{\mathrm{total}}$)) **then**
7	min := F_{CH};
8	$cc_k := r_j$;
9	**end**
10	**end**

The complexity of the CHR algorithm depends on the complexity of searching the cluster members of C_k to find the best cluster head (Lines 4 to 10), which is O($|C_k|$).

3.6 | DCDF

Based on the initial clustering results from the KMC algorithm (Line 2), the energy consumption of each cluster head and cluster member after each period ΔT are defined by Lines 8 and 10, respectively. The DRC algorithm (Line 13) is also called to check if there are any out-of-cluster readers and re-cluster such readers. The CHR algorithm (Line 15) is then called for each cluster to rotate the role of the cluster head among cluster members. The DCDF algorithm is presented below.

Algorithm 4: DCDF algorithm	
Input:	- set of readers $R = \{r_i \mid i = 1, 2, \ldots, N\}$; - number of clusters K;
Output:	- set of clusters $C = \{C_k \mid k = 1, 2, \ldots, K\}$; // where $C_k = \{cc_k, r_j \mid j = 1, 2, \ldots, |C_k|\}$
Process:	
1	timer := 0;
2	C := KMC(R, K);// initial clustering by KMC
3	**while** (simulation time is not over) **do**
4	**if** timer > ΔT **then**
5	**for** i := 1 **to** N **do**
6	**if** (r_i is a cluster head) **then**
7	C_k := cluster of r_i;
8	//the consumed energies of cluster head r_i $E_i := E_i - \sum_{j \neq i}^{|C_k|} E_{Rx} + E_{\mathrm{Px}} + E_{\mathrm{Tx}}$;
9	**else**
10	//the consumed energies of member r_i $E_i := E_i - (E_{\mathrm{Rx}} + E_{\mathrm{Tx}})$;
11	**end**
12	**end**
13	C := DRC(R, C); //re-cluster the out-of-cluster readers
14	**for** k := 1 **to** K **do**
15	C_k := CHR(C_k); //rotate the cluster head role
16	**end**
17	timer := 0;
18	**end**
19	timer := timer + 1;
20	**end**

The KMC (Line 2), DCR (Line 5), and CHS (Line 7) algorithms are independent, meaning the complexity of the DCDF algorithm is $O(K \times N)$ for a round ΔT. This complexity is low, so the energy for consumed by the DCDF algorithm (E_{Ax}) is negligible. In the following simulations, we simply assume a value of zero.

4 | SIMULATION AND ANALYSIS

In our simulations, DCDF was compared to IRDF, which is the best existing algorithm in terms of consumed energy for communication among clustering-based data filtering mechanisms [10]. Figure 4 presents a simulated RFID network in which 100 readers move arbitrarily in a 200 m × 200 m square workspace with a base station at the location of (200, 200). One thousand tags are evenly distributed in this area. The simulation duration is 100 s. After each round ($\Delta T = 1$ s), the mobile readers are checked to determine if some move out of their cluster (by calling the DRC algorithm) and the cluster head role in each cluster is rotated among its cluster members (by calling the CHR algorithm). The other simulation parameters are listed in Table 1.

Our simulation objectives include comparing the following characteristics:

1. The energy consumption of cluster heads and members when applying DCDF.
2. The energy efficiency between IRDF and DCDF based on the energy consumption of cluster heads over the simulation time and when increasing the number of tags in the reading area.
3. The energy balance of readers between IRDF and DCDF, in which the energy balance index (EBI) is defined based on Jain's formula [19] as follows:

$$\mathrm{EBI} = \frac{\sum_{i=1}^{N} {E_i}^2}{N \sum_{i=1}^{N} (E_i)^2}, \tag{6}$$

where N is the number of readers and E_i is the residual energy of reader r_i. A balance is achieved when EBI = 1.

4.1 | Energy consumption of cluster heads and members when applying DCDF

The simulation results in Figure 5 reveal that the average consumed energy of cluster heads is always less than that of readers (approximately 1.5%). This is achieved via the rotation of the cluster head role, where cluster head re-selection is performed periodically and the cluster member with the highest energy is selected as the cluster head. However, as shown in Table 2, the rate of CHR is relatively high with an average value of 0.75. This high CHR results in the extra consumption of additional energy [20].

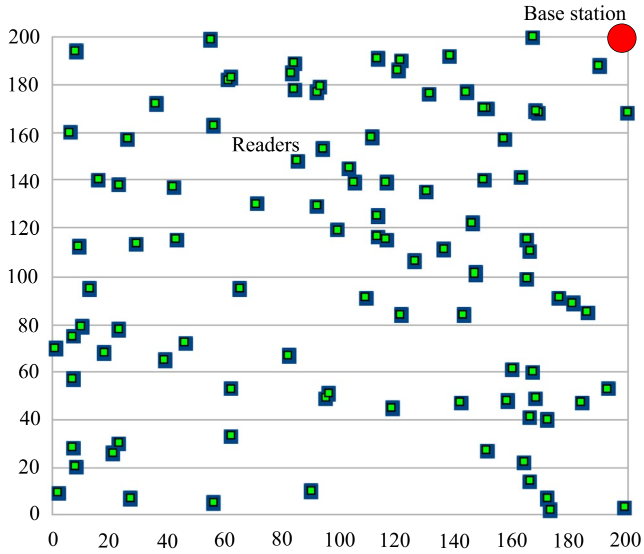

FIGURE 4 Simulation network

TABLE 1 Simulation parameters

Parameters	Values
Initial energy of readers	5 kj
E_{elec}	5×10^{-7} j/byte
E_{fs}	1×10^{-7} j/byte
E_{mp}	0.5×10^{-7} j/byte
α	0.7
$\Delta n/n$	0.2
Reading range of readers	5 m
Transmission data of tags	1000 bytes
K	10
th_{radius}	10 m
Number of tags	1000

4.2 | Comparison of the energy efficiencies of DCDF and IRDF

Regarding the energy consumption of cluster heads for DCDF and IRDF, Figure 6 reveals that DCDF consumes less energy than IRDF. For example, at the beginning of the simulation (10 s), the difference in the average consumed energy between DCDF and IRDF is approximately 6.6%, but this increases to 66% at the end of the simulation (100 s). This is because the energy consumption of cluster heads in DCDF is shared among cluster members. Additionally, limiting the number of readers per cluster also prolongs the lifetimes of cluster heads.

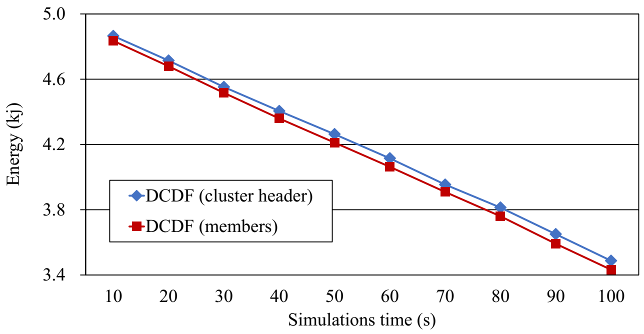

FIGURE 5 Energy consumption of cluster heads and cluster members when applying DCDF

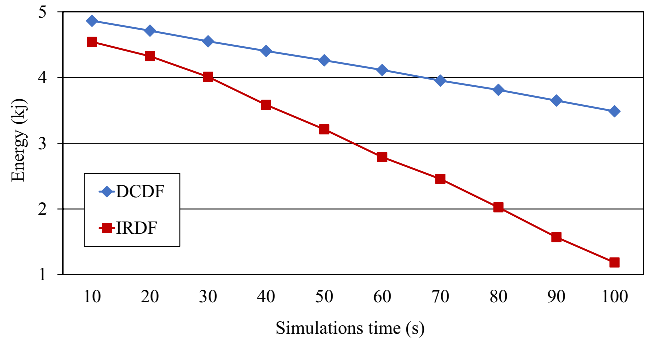

FIGURE 6 Energy consumption of cluster heads for DCDF vs. IRDF over simulation time

TABLE 2 Number of readers in each cluster over 100 s of simulation

Simulation time (s)	10	20	30	40	50	60	70	80	90	100
Cluster 1	10	10	10	9	10	10	11	10	10	10
Cluster 2	9	10	11	10	9	11	9	10	10	9
Cluster 3	11	9	9	10	11	10	11	9	11	11
Cluster 4	10	10	10	10	10	9	9	10	9	10
Cluster 5	10	9	10	11	10	10	10	9	9	10
Cluster 6	10	10	10	10	10	10	10	10	10	11
Cluster 7	9	10	9	11	10	9	10	10	10	9
Cluster 8	11	11	11	10	11	11	11	11	11	11
Cluster 9	9	10	9	10	9	9	9	10	9	10
Cluster 10	11	11	11	9	10	11	10	11	11	9

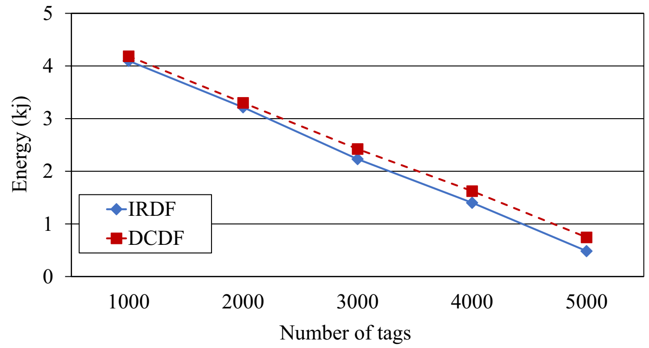

FIGURE 7 Energy consumption of cluster heads in DCDF vs. IRDF when increasing the number of tags

TABLE 3 Energy of cluster heads at 50 s

Cluster	1	2	3	4	5	6	7	8	9	10
Energy (k_j)	4.08	4.10	4.10	4.13	4.10	4.13	4.12	4.11	4.10	4.10

Another comparison in terms of energy efficiency between DCDF and IRDF when increasing the number of tags is presented in Figure 7, where the average energy consumption of readers in DCDF is lower than that in IRDF (from 2% to 35% when increasing from 1000 to 5000 tags over 50 s of simulation).

4.3 | Comparison of energy balancing between DCDF and IRDF

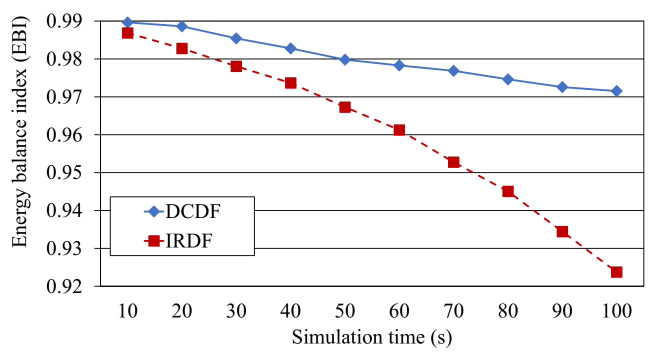

FIGURE 8 EBI values of IRDF vs. DCDF over simulation time

Another benefit of the rotation of the cluster head role and balancing the number of readers per cluster is an increase in the energy balance among readers when applying DCDF compared to IRDF. Specifically, as shown in Table 2, the numbers of readers per cluster are evenly distributed and the energy levels of the cluster heads extracted at 50 s are very consistent (Table 3). This contributes to making the *EBI* (Equation 6) approach 1, as shown in Figure 8.

However, the energy balance decreases with increasing simulation time. This is because the initial energy of the readers is consistent (5 kj), so the EBI is approximately 0.99 at 10 s. With increasing simulation time, the energy of the readers is consumed at different rates, so the EBI decreases. However, compared to the EBI of IRDF, the EBI of DCDF is always greater and this distance increases with simulation time. This demonstrates a clear advantage for DCDF.

5 | CONCLUSION

This paper proposed the DCDF model for mobile RFID networks. DCDF consists of three main stages: initial clustering based on the KMC algorithm, dynamic clustering of mobile readers based on the DRC algorithm, and rotation of the cluster head role based on the CHS algorithm. Simulation results demonstrated that DCDF is more efficient than IRDF in terms of energy consumption and energy balancing. These advantages conserve the energy of readers and prolong the lifetimes of networks. However, the CHR algorithm incurs additional energy consumption and compromising solutions require further investigation.

ACKNOWLEDGMENTS

This work was supported by the Strong Research Group Program of Hue University.

ORCID

Viet Minh Nhat Vo https://orcid.org/0000-0002-0686-5529
Van Hoa Le https://orcid.org/0000-0001-6621-8617

REFERENCES

1. H. Liu et al., *Taxonomy and challenges of the integration of RFID and wireless sensor networks*, IEEE Netw. **22** (2008), no. 6, 26–35.
2. X. Zhu, S. K. Mukhopadhyay, and H. Kurata, *A review of RFID technology and its managerial applications in different industries*, J. Eng. Technol. Manag. **29** (2012), no. 1, 152–167.
3. S. Zhang and H. Zhang, *A review of wireless sensor networks and its applications*, in Proc. IEEE Int. Conf. Autom. Logistics (Zhengzhou, China), Aug. 2012, pp. 386–389.
4. L. Wang et al., *Data cleaning for RFID and WSN integration*, IEEE Trans. Industr. Inform. **10** (2014), no. 1, 408–418.
5. S. S. Park, *An IoT application service using mobile RFID technology*, in Proc. Int. Conf. Electron., Inf., Commun. (Honolulu, HI, USA), Jan. 2018, pp. 1–4.
6. M. M. Afsar and M.-H. Tayarani-N, *Clustering in sensor networks: A literature survey*, J. Netw. Comput. Appl. **46** (2014), 198–226.
7. W. Choi and M. Park, *In-network phased filtering mechanism for a large-scale RFID inventory application*, in Proc. Int. Conf. Inf. Technol. Applicat. (Harbin, China), Jan. 2007, pp. 401–405.
8. D.-S. Kim et al., *Energy efficient in-network phase RFID data filtering scheme*, in Ubiquitous Intelligence and Computing, vol. 5061, Springer, Berlin, Germany, 2008, pp. 311–322.
9. A. K. Bashir et al., *Energy efficient in-network RFID data filtering scheme in wireless sensor networks*, Sensors **11** (2011), no. 12, 7004–7021.
10. A. K. Bashir et al., *In-network RFID data filtering scheme in RFID-WSN for RFID applications*, in Intelligent Robotics and Applications, vol. 8103, Springer, Berlin, Germany, 2013, pp. 454–465.
11. D. Shin and S. Park, *An energy efficient scheme for detecting redundant readings in cluster-based model of integrated RFID and wireless sensor networks*, Int. J. Appl. Eng. Res. **12** (2017), no. 14, 4708–4722.
12. Q. Ma et al., *MRLIHT: Mobile RFID-based localization for indoor human tracking*, Sensors **20** (2020), no. 6, 1–19, doi:10.3390/s20061711.
13. M. Darianian and M. P. Michael, *Smart home mobile RFID-based internet-of-things systems and services*, in Proc. Int. Conf. Adv. Comput. Theory Eng. (Phuket, Thailand), Dec. 2008, pp. 116–120.
14. A. Motroni, A. Buffi, and P. Nepa, *Localization of a mobile device equipped with an RFID reader*, in Proc. IEEE Int. Conf. RFID Technol. Applicat. (Warsaw, Poland), Sept. 2017, pp. 74–79.
15. C. H. Hsu et al., *Alleviating reader collision problem in mobile RFID networks*, Pers. Ubiquitous Comput. **13** (2009), no. 7, 489–497.
16. J. B. Eom, S. Bin Yim, and T. J. Lee, *An efficient reader anticollision algorithm in dense RFID networks with mobile RFID readers*, IEEE Trans. Ind. Electron. **56** (2009), no. 7, 2326–2336.
17. B. C. Chen et al., *Mutual authentication protocol for role-based access control using mobile RFID*, Appl. Sci. **6** (2016), no. 8. 1–10, https://doi.org/10.3390/app6080215.
18. S. Jin, M. Zhou, and A. S. Wu, *Sensor network optimization using a genetic algorithm*, in Proc. World Multi-conf. Systemic, Cybernetics Inform. (Orlando, FL, USA), July 2003, pp. 1–6.
19. R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*, DEC Technical Report TR301, vol. cs.NI/9809, no. DEC-TR-301, 1984.
20. E. Hossain, R. Palit, and P. Thulasiraman, *Clustering in mobile wireless ad hoc networks*, in Wireless Communications Systems and Networks, Springer, Boston, MA, 2004, pp. 383–424.

AUTHOR BIOGRAPHIES

Viet Minh Nhat Vo received his PhD in Cognitive Informatics from the University of Quebec in Montreal, Canada, in 2007. He is currently an associate professor at Hue University, Vietnam. His research interests include optical packet/burst-based switching networks, mobile RFID/sensor systems, quality of service, optimization, and traffic engineering.

Van Hoa Le received his PhD in Computer Science from Hue University, Vietnam, in 2020. He is currently a lecturer at Hue University, Vietnam. His research interests include optical packet/burst-based switching networks, mobile RFID/sensor systems, fairness, quality of service, and traffic engineering.