Research Article

A Reinforcement Learning-Based Dynamic Clustering Algorithm for Compressive Data Gathering in Wireless Sensor Networks

Xun Wang and Hongbin Chen

School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China

Correspondence should be addressed to Hongbin Chen; chbscut@guet.edu.cn

Received 15 January 2022; Revised 7 April 2022; Accepted 13 April 2022; Published 9 May 2022

Academic Editor: Jose M. Barcelo-Ordinas

Copyright © 2022 Xun Wang and Hongbin Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Compressive data gathering (CDG) is an effective technique to handle large amounts of data transmissions in resource-constrained wireless sensor networks (WSNs). However, CDG with static clustering cannot adapt to time-varying environments in WSNs. In this paper, a reinforcement learning-based dynamic clustering algorithm (RLDCA) for CDG in WSNs is proposed. It is a dynamic and adaptive clustering method aiming to further reduce data transmissions and energy consumption in WSNs. Sensor nodes act as reinforcement learning (RL) agents which can observe the environment and dynamically select a cluster to join in. These RL agents are instructed by a well-designed reward scheme to join a cluster with strong data correlation and proper distance. It is also a distributed and lightweight learning method. All agents are independent and operate in parallel. Additional overheads introduced by RL are lightweight. Computations of a linear reward function and a few comparison operations are needed. It is implementable in WSNs. Simulations performed in MATLAB validate the effectiveness of the proposed method and simulation results show that the proposed algorithm achieves the desired effect as well as fine convergence. It decreases data transmissions by 16.6% and 54.4% and energy consumption by 6% and 29%, respectively, compared to the two contrastive schemes.

1. Introduction

Wireless sensor networks (WSNs) are the perceptual layer of Internet of Things (IoT). With increasing applications of IoT, the scale of WSNs is expanded than ever before. Data transmissions in WSNs are growing exponentially. It is a severe challenge for resource-constrained WSNs. Fortunately, data compression technology can alleviate this dilemma. Data compression cuts down data transmissions and reduces energy consumption of sensor nodes and prolongs lifetime of WSNs.

Traditional data compression techniques, such as source coding and in-network coding [1–3], are usually computationally complex and need prior and global information of networks, which is hard to implement in WSNs. Compressive data gathering (CDG) combines compressive sensing (CS) [4] with data gathering and compresses sensing data while sampling. Nodes in WSNs execute compression encoding, which needs only a few simple multiplications and additions. Complicated decoding algorithms are carried out at the sink. Prior and global information about networks is not needed. So it is an excellent data compression solution for WSNs.

Luo et al. firstly developed the complete process of CDG in WSNs in [5]. Then, in [6], they reduced redundant data and improved the basic scheme. Reference [7] denoted CDG scheme in [5] as plain-CDG and proved that plain-CDG had limited effect on improvement of networks’ throughput. They proposed a scheme called hybrid-CDG which is similar to the framework in [6]. In order to further decrease data transmissions, reference [8] proposed a sparse-CDG scheme in which partial nodes in WSNs need to take part in data gathering. Plain-CDG, hybrid-CDG, and sparse CDG are three basic frameworks in subsequent references.
In the past decade, a large amount of literature committed to enhance the performance of CDG through various technologies of routing, clustering topology control, combination of metaheuristic algorithms or learning algorithms, etc. All of these works obtained performance gains which will be described in details in the next section. However, there are still further researches worth doing.

In the published works, most clustering and routing strategies in CDG are static, which cannot be adaptive with fluctuations of sensing data and environment variations in WSNs. Secondly, they are usually based on minimum distance which do not consider characteristics of the CS theory. Sensing data from nodes with minimum distance may not be most correlative. Thirdly, in the majority of previous methods, the required number of CS measurements used for data reconstruction, generally denoted as $M$, is constant. In reality, $M$ is relevant to sparsity of sensing data which is variable in time. Due to constant $M$, two opposite results are likely to introduce, i.e., data redundancy or data insufficiency, which will decrease transmission efficiency of networks or accuracy of data recovery.

To overcome the above problems, we propose a reinforcement learning-based dynamic clustering algorithm (RLDCA) for CDG in WSNs. We introduce reinforcement learning (RL) into the process of CDG. The learner of RL, i.e., the agent, observes the environment and takes actions and then receives rewards. The agent attempts to maximize its received rewards during the learning process. Because it learns from interaction, the agent can make adaptations according to variations in the surrounding environment. Meanwhile, different from other learning schemes, there are several lightweight operations in RL which are implementable in resource-constrained WSNs. The main contributions of this paper are summarized as follows:

(i) The proposed scheme first adopts CDG to cut down data transmissions and then combines the RL algorithm to enhance the performances of CDG. It takes sensor nodes as agents. Each agent dynamically selects a cluster to join in by a well-designed RL scheme with trade-off between data correlation and distance in order to minimize data transmissions and reduce energy consumption to prolong lifetime of the WSN.

(ii) The lightweight Upper Confidence Bound (UCB) algorithm is used as the guideline for agents to take actions. The corresponding state vector, action set, and reward function are designed based on intracluster data correlation and distances between nodes and CHs.

(iii) Dynamical CS measurement number $M$ is computed based on intracluster data sparsity in each round of data gathering assuring both data transmission efficiency and accuracy of data recovery.

The rest of this paper is organized as follows. Section II briefly introduces some related works about CDG and RL. Section III describes the system model and problem formulation. Section IV presents the proposed RLDCA for CDG in WSNs in details. Simulation results for performance evaluation are given in Section V. Finally, Section VI concludes this article. Table 1 lists notations and major abbreviations used in this paper.

| Notations | Description |
|-----------|-------------|
| CDG       | Compressive data gathering |
| CS        | Compressive sensing |
| RL        | Reinforcement learning |
| UCB       | Upper Confidence Bound algorithm |
| CH        | Cluster head |
| agent     | RL learner |
| $N$       | The set of all normal sensor nodes |
| $C$       | The set of cluster heads |
| $c_{hi}$  | The cluster head of $i$th cluster |
| $\rho$    | The Pearson correlation coefficient |
| $A$       | The action set |
| $S$       | The state vector |
| $R$       | The reward function |
| ASV       | Action selection value |
| $c$       | The control factor of exploration degree in the UCB algorithm |
| $\alpha$  | Balance factor in $R$ |

2. Related Works

In this section, a review of the research status about CDG technologies is presented. There are many studies on improvement of routing strategies for CDG. Reference [9] improved the Minimum-Spanning-Tree (MST) and took the sink as the root of the MST, which can reduce transmission overhead from projection nodes to the sink. Reference [10] proposed a Weighted Compressive Data Aggregation method which introduced the parameter of tree’s cost. Both hops and distances of routing paths are considered into the tree’s cost. References [11–14] adopted random walk (RW) routing schemes in CDG to reduce computations of precise routing paths and extra information exchanges and enhance network adaptability. The improved methodologies of routing strategies for CDG did not take data correlation into consideration. Our proposed method is designed based on data correlation among nodes. It can further cut down data transmissions of CDG.

A large amount of literature devoted to enhance the performance of CDG by various clustering technologies. Clustering technology is benefit for traffic load balance and energy efficiency of the network, especially in large-scale WSNs. References [15–17] designed proper clustering methods by improving criteria of CH selection. They considered a lot of elements, such as residual energy, distance to the sink, intra-to-inter distance ratio, node density, and compression ratio, into the criteria of CH selection.
References [18, 19] paid attention to the load balance problem in clustering-based CDG. They grouped nodes into clusters based on node distribution density or equal energy consumption. Reference [20] devoted to address the hot-spot area problem to achieve load balance. References [21–23] exploited data characteristics to devise clustering algorithms. Reference [21] exploited spatial correlation among sensors’ data to decrease data transmissions. Reference [22] considered event-driven WSNs and estimated the location of event source. Nodes around event source form a cluster. Reference [23] proposed a compressibility-based clustering algorithm. Reference [24] noticed that if sizes of some clusters are too small, it is inefficient to implement CDG. So they proposed a cluster size load balance technique for optimal utilization of CDG by keeping the minimum number of nodes in clusters. On the whole, most of these clustering methods are static and based on minimum distance that cannot be adaptive with fluctuations in sensing data or observed environments. Our proposed scheme is a dynamic clustering method based on RL algorithm with better adaptivity.

Besides routing design and clustering topology control, a lot of other technologies were introduced into CDG for further improvements of performances. Metaheuristic algorithms are good at some optimization problems. Reference [25] introduced a Grey Wolf optimization algorithm to search the best path from each CH to the sink with minimum energy consumption in clustering CDG. Reference [26] used the Bees algorithm to determine optimal results of CS recovery in clustering CDG. Reference [27] adopted multiple-objective genetic algorithm to calculate the optimal number of CS measurements and measurement matrix in CDG. Various learning algorithms can make CDG schemes more intelligent and adaptive. References [28, 29] used the dictionary learning method to obtain a sparse basis which has better sparse representation ability in CDG. References [30, 31] incorporated deep learning into CDG and designed a deep compressed sensing network to build a measurement matrix and reconstruct data from CS measurements. Reference [32, 33] adopted fuzzy logic in data aggregation. Reference [32] proposed a two-tier distributed fuzzy logic based protocol (TTDFP) for efficient data aggregation. The two-tier protocol includes a fuzzy clustering algorithm and a fuzzy routing procedure. Reference [33] used a fuzzy rule system-based RL algorithm to select data aggregator nodes. Mobile collectors and mobile sink can save energy consumption of transmission at node side. Taking Unmanned aerial vehicles (UAVs) as a mobile data collector has better transmission at node side. Taking Unmanned aerial vehicles (UAVs) as a mobile data collector has better transmission at node side. Taking Unmanned aerial vehicles (UAVs) as a mobile data collector has better transmission at node side. Taking Unmanned aerial vehicles (UAVs) as a mobile data collector has better transmission at node side.

3. System Model and Problem Formulation

3.1. Wireless Sensor Network Model. In this paper, we consider a two-dimensional WSN as shown in Figure 1. N sensor nodes, denoted by a node set \( N = \{ s_1, s_2, s_3, \cdots, s_N \} \), are randomly deployed in the plane area \( A \) for environmental monitoring, e.g., temperature, humidity, PM 2.5, and so on. A clustering topology control technology is adopted in the WSN. \( N \) sensor nodes are grouped into \( p \) clusters. Each cluster is managed by a cluster head (CH) node denoted as \( c_{ch} \), and these \( p \) CHs are denoted as a CH set \( C = \{ c_{ch_1}, c_{ch_2}, \cdots, c_{ch_p} \} \). All the nodes are static and periodically sense ambient data and then transmit to a CH through a single hop. CHs compress intracluster data and then send compressed data to the sink through a backbone tree among CHs. The sink is located at the center of the area \( A \), where the compressed data is recovered for further use.

Sensor nodes own equal initial energy denoted as \( E_0 \). The CH nodes in \( C \) are homogeneous and equipped with more energy, more computational and storage resources than normal sensor nodes in \( N \). The initial energy of CHs are denoted as \( E_c \). The maximum communication radius of normal sensor nodes and CH are denoted as \( r \) and \( r_{ch} \), respectively.

Sensing data in the network is time-varying. Hence, the adopted clustering scheme is based on RL algorithm to enhance adaption of the network. It is a dynamic clustering technology in which nodes will choose a cluster once again at the beginning of each round. All the nodes acted as RL agents independently perform the RL algorithm under the uniform scheduling. In each round of work, node \( s_i \) takes...
action and selects the jth cluster to join in and then sends its sensing data to the CH $c_{ji}$. Then, $s_i$ receives a reward of the current action. The reward is preserved in the memory of $s_i$ which is accumulated as learning experiences.

3.2. Data Transmission Model of CDG. Suppose N sensor nodes are divided into $p$ clusters and there are $L_i$ cluster members (CMs) in the ith cluster, i.e., $\sum_{i=1}^{p} L_i = N$. In the ith cluster, the sensing data from CMs to the CH is denoted as $x_i = [x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iL_i}]^T$. According to the CS theory, if $x_i$ is sparse or compressible and its sparsity is $K_i$, then $x_i$ can be compressed to $y_i$ with $M_i$ measurements, i.e.,

$$y_i = \Phi_i x_i = \Phi_i \Psi_i s_i = A_i s_i,$$  

(1)

where $\Phi_i$ is a Gaussian random matrix generated by the CH, and $\Psi_i$ is a $L_i \times L_i$ discrete cosine transformation basis used to convert a compressible signal into a sparse signal in the transformation domain. $A_i = \Phi_i \Psi_i$ is called sensing matrix or measurement matrix in CS. If $A_i$ satisfies the restricted isometry property (RIP) rule,

$$M_i \geq C \cdot K_i \cdot \log \frac{L_i}{K_i},$$  

(2)

where $C$ is a constant and $K_i$ denotes sparsity of $x_i$; then, $x_i$ can be recovered from $M_i$ measurements in $y_i$ with a probability close to 1. On the premise of $\Phi_i$ is a random matrix, $M_i = 3K_i \sim 4K_i$ is usually sufficient [4].

The data correlation can be characterized by the Pearson correlation coefficient, as follows:

$$\rho_{ij} = \frac{\sum (d_i - \bar{d}_i)(d_j - \bar{d}_j)}{\sqrt{\sum (d_i - \bar{d}_i)^2 \sum (d_j - \bar{d}_j)^2}},$$  

(3)

where $\rho_{ij}$ represents correlation coefficient of two data sets $d_i$ and $d_j$, and $\bar{d}_i$ and $\bar{d}_j$, respectively, represent the mean value of $d_i$ and $d_j$. Data sparsity is relevant to the correlation within the data set. Hence, those sensor nodes with strong correlation in the sensing data are grouped into a cluster. The intracluster data can be compressed more compactly, and then, data transmissions in the whole network can be greatly reduced.

3.3. Energy Consumption Model. The energy of nodes is mainly consumed at the data transmission stage. The objective of the proposed RLDCA is to cut down the energy consumption of nodes through decreasing data transmissions, thereby prolonging the lifetime of the WSN. We adopt the classical energy consumption model in [38]. When sending a data packet of $l$ bits, the energy consumption $E_{Tx}$ in a node mainly includes the energy consumption in processing circuits and transmitted amplifiers. When receiving a data packet of $l$ bits, the energy consumption $E_{Rx}$ mainly consists of the energy consumption in processing circuits, as follows:

$$E_{Tx}(l, d) = E_{Tx-elec}(l) + E_{Tx-amp}(l, d)$$

\[
= \begin{cases} 
    l(E_{elec} + \epsilon_{fs}d^2) & d < d_0 \\
    l(E_{elec} + \epsilon_{mp}d^2) & d \geq d_0,
\end{cases}
\]

$$E_{Rx}(l) = E_{Rx-elec}(l) = lE_{elec},$$  

\[
d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}},
\]

(4)

where $E_{elec} = 50nJ/bit$ is the energy consumed in processing circuits to transmit or receive 1 bit data, $d$ is the distance between a pair of sender and receiver, and $d_0$ is the threshold of distance in free space. When $d < d_0$, the channel model in free space is adopted and the amplifier consumes $\epsilon_{fs}d^2$ energy to send 1 bit data, where $\epsilon_{fs} = 10pJ/bit/m^2$. When $d \geq d_0$, the multipath fading channel model is adopted and the amplifier consumes $\epsilon_{mp}d^2$ energy to send 1 bit data, where $\epsilon_{mp} = 0.0013pJ/bit/m^4$.

3.4. Problem Formulation. In this paper, we devote to design a dynamic clustering algorithm for CDG based on RL and utilize characteristics of CDG to minimize data transmissions, balance traffic load, and prolong lifetime of the WSN.

3.4.1. Definition of the WSN’s Lifetime. The WSN’s lifetime is the duration of work and task execution by this WSN. First node dies (FND) is usually taken as the parameter of the WSN’s lifetime. FND is defined as the number of data transmission rounds accomplished by the WSN before the first node depletes its energy. A round means that all nodes finish a complete process of data sensing and transmission to the sink. Hence, reduction of energy consumption and balance traffic load are beneficial for extension of the lifetime.

3.4.2. Problem Formulation. According to the CS theory, if the correlation among original data is stronger, less CS measurements are required for accurate data recovery. Thus, in the proposed RLDCA, each node chooses the cluster with strong data correlation to join in. The objective function is to minimize data transmissions in the WSN thereby reduce the energy consumption and prolong lifetime of the network, as follows:

$$\min \sum_{i=1}^{p} \sum_{j=1}^{L_i} (L_i + M_j),$$  

(5)

where $L_i$ is the intracluster data transmissions in the ith cluster, $M_j$ is the data transmissions of the CH $c_{ji}$ in the ith cluster, and $r_{max}$ denotes the number of running rounds during lifetime of the WSN.

4. Algorithm Description

The proposed RLDCA will be introduced in detail in this section. Each normal sensor node is an agent and performs a RL procedure to take an action (select a cluster) and then
gets a reward stored in its memory. It is an on-line learning scheme which does not need training data. The computation of RLDCA is also lightweight at normal sensor nodes. Only a set of linear reward functions and comparing operations are needed. Thus, an agent is instructed to work as we expect by a reward stored in its memory. It is an on-line learning scheme which does not need training data. The computation of RLDCA is also lightweight at normal sensor nodes. Only a set of linear reward functions and comparing operations are needed.

All agents are independent and operate in parallel. They are scheduled by the sink. Triggered by a START message, each agent begins a round of work as shown in Figure 2. The process of learning and data transmission are alternate in a round. In the first and second phase, each agent selects an action (a cluster) and then performs this action (sends its sensing data to the CH of the selected cluster). In the third phase, the CH of the selected cluster returns a reward of data correlation to the agent. Together with a reward of distance to CHs saved locally, each agent computes an integrated reward and saves in its memory. In the last phase, CHs perform CDG and compress intracluster data and then transmit data to the sink though a backbone tree.

4.1. Reinforcement Learning. RL imitates human’s behavior which learns from interaction with ambient environments. It is a unsupervised learning method. A RL agent (learner) is not told which action to select but instead takes action autonomously and then receives a reward. It tries to find out which action yields the best reward. The objective of an agent is to maximize the total reward over the long run. Thus, an agent is instructed to work as we expect by a well-designed reward scheme.

The procedure of RL is that an agent observes the environment and gets state vector $\mathbf{S}$, then chooses an action from the action set $\mathbf{A}$, and at last receives rewards $r$ based on a reward function $R$. In order to maximize the total reward, the value function is used to measure the average reward of each action in $\mathbf{A}$. However, in most case, the value function is unknown and needs to be evaluated by continuously trying. An agent can exploit the evaluated value function to select the action with the highest value. But the evaluated value function may deviate from the true value. An agent also needs to explore those actions without the highest value. RL algorithms need to trade off exploitation and exploration during the process of learning. It is called the exploration-exploitation dilemma.

In this paper, the task of agents is to choose a cluster to join in. It is a multichoice problem which can be well solved by the Upper Confidence Bound (UCB) algorithm [39]. The UCB algorithm can properly balance the exploration-exploitation dilemma in RL. The strategy of action selection is based on the following:

$$A_t = \arg \max_a \left[ Q_t(a) + c \left( \ln t / N_t(a) \right)^{1/2} \right], \quad (6)$$

$$Q_t(a) = \frac{\sum_{i=1}^{t-1} r_i \cdot I}{\sum_{i=1}^{t-1} I}, \quad (7)$$

$$I = \begin{cases} 1 & \text{when the selected action is } a \text{ in } \mathbf{A} \text{ at time } t, \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

In Equation (6), $A_t$ denotes the action selected at the time step $t$, and $Q_t(a)$ is the evaluated value function of action $a$ which is defined in Equation (7). $c > 0$ is a constant used to control the degree of exploration, $\ln t$ denotes the natural logarithm of $t$, and $N_t(a)$ denotes the number of times that action $a$ has been selected prior to time $t$. In Equation (7), $r_i$ denotes the corresponding reward at time $i$.

4.2. Parameters Design of RLDCA. The RLDCA is implemented in a distributed way. Each sensor node rather than the sink in the WSN acts as a RL agent. This distributed way decreases the scale of state and action vector and reduces communication overheads from the sink to sensor nodes.

As described in Section III, there are $N$ sensor nodes in the WSN. These $N$ nodes are $N$ parallel and independent RL agents in the proposed RLDCA. Although they are independent, $N$ agents interact with the same environment and perform the same learning procedure. Thus, they own the same state vector $\mathbf{S}$, action set $\mathbf{A}$, and reward function $R$ described as follows.

4.2.1. The State Vector $\mathbf{S}$. An agent observes the environment and gets the state vector $\mathbf{S}$ based on which a rewards is calculated. The state vector of the RLDCA is $\mathbf{S} = [\rho_i, d_i]$, where $\rho_i$ represents the data correlation between agent and $i$th cluster which is calculated by Equation (3), and $d_i$ represents the distance from an agent to the CH of $i$th cluster.

4.2.2. The Action Set $\mathbf{A}$. According to the network model, there are total $p$ clusters in the network. At the beginning of each round, each node chooses a cluster to join in. So the action set $\mathbf{A}$ is designed as $\mathbf{A} = \{ch_1, ch_2, \ldots, ch_p\}$, where $ch_i$ is the cluster head of the $i$th cluster. An agent takes an action of $ch_i$ in $\mathbf{A}$ means that this node joins in $i$th cluster and transmits its sensing data to the cluster head $ch_i$.

4.2.3. The Reward Function $R$. The reward function $R$ considers two elements in the state vector $\mathbf{S}$ and is defined as

$$R = a r_d + (1 - a) r_c, \quad (9)$$

where $0 < a < 1$ is a constant used to modify the ratio of the two elements in $\mathbf{S}$. If the value of $a$ is higher, each node will place more emphasis on data correlation when selecting a cluster.
cluster to join in. \( r_c \) and \( r_d \) are rewards based on \( \rho_i \) and \( d_I \) in \( S_i \), respectively. In order to simplify nodes’ computational complexity, a linear and relative reward function model is adopted in this paper. First, \( p \) actions in the action set \( A \) are sorted, respectively, in descending order of \( \rho_i \) and in ascending order of \( d_I \). Then, the normalized reward value \( r_c \) and \( r_d \) is allocated to \( p \) actions in \( A \) according to the previous order. The reward values in \( r_c \) and range from 1 to -1, and the step between two adjacent values is \( \Delta \) calculated by.

\[
\Delta = \frac{2}{p - 1}.\tag{10}
\]

4.3. The RLDCA Procedure. The complete procedure of the proposed RLDCA for CDG is presented in this part. The pseudo codes of the RLDCA are listed in Algorithm 1. In initial phase, all nodes create the initial action set \( A \), reward Table \( R \), and action selection value (ASV) table \( A_t \). After receiving the START message, nodes start performing the RL strategy (lines 4-12). Lines 4-9 choose an action, and lines 10-11 get the reward and update the ASV table. Since there is no experience in the initial round, the reward and ASV table of all actions are zero. So each agent randomly chooses an action from \( A \). In the remaining rounds, the agent chooses the best action based on accumulated experiences preserved in the reward and ASV table. Lines 13-19 perform CDG by CHs. CHs calculate the sparsity of the intracluster sensing data and generate a Gauss random matrix to compress the intracluster sensing data and then send the compressed data to the sink. At last, the sink verifies the integrity of the received data and then broadcasts the START message of the next round.

It is an on-line learning algorithm. Considering WSN is a resource-limited network, the time and space complexity of the algorithm should be analyzed. We focus on the complexity of each round at normal node side. The extra required memory space introduced by learning processes at each normal node (agent) side is \( O(5 \times p) \) used to store values of \( N_i(a) \), \( Q_i(a) \), and \( A_i \) in Equations (6) and (7) and \( r_c \), \( r_d \) in Equation (9). Computation resources are mainly consumed at comparison operations of action selection and reward computation. The action selection is based on Equation (6), and it needs \( p \) comparison operations. In the computation of reward \( r_c \), \( p \) numbers need to be sorted and the computational overhead is \( O(p \log(p)) \). Besides, the computation complexity of Equation (7) is \( O(p) \). On the whole, the proposed RLDCA is a light-weight algorithm for normal sensor nodes.

5. Performance Evaluations

In this section, we present simulation results to evaluate the performance of the proposed RLDCA. Simulations are performed on the MATLAB platform. During the experimental evaluation, 100 sensor nodes are evenly distributed in a 70
Table 2: Parameters Setting

| Parameters                        | Value            |
|-----------------------------------|------------------|
| Node’s initial energy             | $E_n = 0.5 J$    |
| CH’s initial energy               | $E_c = 5 J$      |
| Communication range of normal nodes | $r = 100 m$    |
| Communication range of CHs       | $r_{ch} = 200 m$|
| Measurement matrix of CHs         | Gaussian random matrix |
| CDG method                        | Plain CDG        |
| Length of data packets            | 2000 bit         |
| Circuit energy consumption        | $E_{elec} = 50 mJ/bit$ |
| Amplifier energy consumption factor in free space ($d < d_0$) | $\epsilon_{fs} = 10 pJ/bit/m^2$ |
| Amplifier energy consumption factor in multipath channel ($d \geq d_0$) | $\epsilon_{mp} = 0.0013 pJ/bit/m^4$ |
| Threshold of distance             | $d_0 = 87.7 m$   |
| Learning parameter                | $c = 1$          |

m × 80 m rectangular area. These 100 nodes are divided into 6 clusters. Each cluster is managed by a fixed CH. So there are total 6 fixed CH nodes in the network. At the beginning of each round, sensor nodes dynamically select a cluster to join in based on the proposed learning algorithm. All the nodes are homogeneous and have the same initial energy. The 6 fixed CHs are super nodes with 10 times initial energy of normal sensor nodes. The sea surface temperature data set published by the Earth System Research Laboratory [40] is taken as the sensing data set. The parameters used in the simulation are listed in Table 2. We compare the proposed RLDCA with the Hybrid Clustering-based Data Collection Scheme (HCDCS) in [41] which is a static clustering CDG method without RL and the Minimum Distance-based Clustering (MDC) method without CS and RL. The parameters of data transmissions and energy consumption in the WSN are analyzed.

Figure 3 shows the WSN’s total data transmissions and energy consumption of the RLDCA, HCDCS, and MDC. We can see that the proposed RLDCA has less total data transmissions (Figure 3(a)) and energy consumption (Figure 3(b)) than the two contrastive algorithms. The HCDCS is also a data correlation-based clustering method but a static clustering method without RL. Simulation results confirm the advantage of the proposed algorithm. At the end of 500th round, the proposed RLDCA reduces total data transmissions by 16.6% and 54.4% and energy consumption by 6% and 29%, respectively, compared to the HCDCS and MDC. The drop of data transmissions is more obvious than that of energy consumption. The reason is that energy consumption is related to distance. Agents sometimes choose a far but more correlated cluster which will increase energy consumption to some extent. Both the proposed RLDCA and the HCDCS are CDG methods which obviously outperform the MDC method without CDG. The advantage of CDG is validated.

In order to analyze statistical significance of the obtained performance gain, we have repeated this comparison experiment 10 times with different data segments. The final performance gain is calculated by averaging values of 10 experiments (With the help of statistical analysis software Statistical Program for Social Sciences (SPSS), paired sample T test was used to analyze the statistical significance of 10 groups of experimental data. The results show that there are significant differences ($P < 0.05$) between the proposed algorithm and the two contrastive algorithms).

However, the advantage of RL does not appear from the very beginning. We notice that total data transmissions and energy consumption of the proposed RLDCS are larger than HCDCS in the early stage. We magnify the result of the first 100 rounds in Figure 3 as shown in Figure 4. It is shown that total data transmissions and energy consumption of the proposed RLDCA are higher than the HCDCS in the first 100 rounds. It is in accordance with characteristics of RL. At the beginning, RL agents own little experience and cannot choose a proper action. It leads to higher data transmissions and energy consumption. That is, it is needed a process with duration about 100 rounds to accumulate enough experience. The proposed RLDCA stands out after this process.

Convergence of the proposed algorithm is verified. Figure 5 shows the average reward of the sensor node $s_i$ with different values of the parameter $c$ in Equation (6). We can see that the proposed RLDCA is convergent. The average reward received by the agent rises steadily with learning rounds. The parameter $c$ in Equation (6) controls the degree of exploration in RL. Figure 5 shows how different values of $c$ influence the average reward. In the proposed method, when $c$ is equals to 1, the agent gets the maximum average reward. Hence, we choose $c = 1$ in the simulation.

Figure 6 shows the relationship between the energy consumption and the parameter $\alpha$ in the reward function of Equation (9) in the 500th round. During the process of clustering, $\alpha$ is used to balance the distance to CHs and data correlation. If the value of $\alpha$ is higher, an agent places more emphasis on data correlation when it chooses a cluster to join in. We can see from Figure 6 that the energy consumption of the network declines with the increase of the value of $\alpha$. It is testified that considering data correlation during the clustering phase can effectively reduce the energy consumption of the network. However, when the value of $\alpha$ exceeds 0.8, the energy consumption goes up. That means besides data correlation, the distances between nodes and CHs cannot be ignored in the proposed clustering algorithm. We find that the optimal value of $\alpha$ is 0.8 in the proposed RLDCA which results in the minimum energy consumption.

6. Conclusion

In this paper a reinforcement learning-based dynamic clustering algorithm (RLDCA) for CDG in WSNs has been proposed. It is a distributed method in which all the normal sensor nodes are independent RL agents. The well-designed RL algorithm instructs the agents to select a cluster with strong data correlation and proper distance. It is a dynamic clustering method because each agent selects a cluster to join in at the beginning of each round of data gathering. This dynamic scheme has well adaptivity to time-varying data in the network.
The fixed CHs receive intracluster data and perform CDG to cut down the data transmissions and energy consumption of the WSN. It is also a lightweight and distributed algorithm which is implementable in resource-constrained WSNs. The additional computational overhead at normal node side introduced by the RL algorithm is only a few comparison operations and linear computations.

Simulation results validate the effectiveness of the proposed RLDCA by comparing it with the HCDCS and MDC algorithms. The proposed RLDCA reduces total data transmissions by 16.6% and 54.4% and energy consumption by 6% and 29%, respectively, compared to the two contrastive algorithms. The convergence of the algorithm is also verified in the simulation. At last, due to characteristics of
natural science foundation (2020gxnsfba297097).

This research was supported by the national natural science foundation of china (62061009, 62161031) and the guangxi natural science foundation (2020gxnsfba297097).

Acknowledgments

The authors declare that there is no conflict of interests regarding the publication of this paper.

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] D. Slepian and J. Wolf, “Noiseless coding of correlated information sources,” IEEE Transaction on Information Theory, vol. 19, no. 4, pp. 471–480, 1973.

[2] R. Cristescu, B. Beferrull-Lozano, M. Vetterli, and R. Wattenhofer, “Network correlated data gathering with explicit communication: NP-completeness and algorithms,” IEEE/ACM Transaction on Networking, vol. 14, no. 1, pp. 41–54, 2006.

[3] H. Gupta, V. Navda, S. Das, and V. Chowdhary, “Efficient gathering of correlated data in sensor networks,” ACM Transaction on Sensor Networks, vol. 4, no. 1, pp. 1–31, 2008.

[4] E. J. Candès and M. B. Wakin, “An introduction to compressive sampling,” IEEE Signal Processing Magazine, vol. 25, no. 2, pp. 21–30, 2008.

[5] C. Luo, F. Wu, J. Sun, and W. C. Chen, “Compressive data gathering for large-scale wireless sensor network,” in Proceedings of the 15th annual international conference on Mobile computing and networking, pp. 145–156, Beijing, China, 2009.

[6] C. Luo, F. Wu, J. Sun, and W. C. Chen, “Efficient measurement generation and pervasive sparsity for compressive data gathering,” IEEE Transactions on Wireless Communications, vol. 9, no. 12, pp. 3728–3738, 2010.

[7] J. Luo and L. Xiang, “Does Compressed Sensing Improve the Throughput of Wireless Sensor Networks?,” in IEEE International Conference on Communications, Cape Town, South Africa, 2010.

[8] D. Ebrahimi and C. Assi, “Optimal and efficient algorithms for projection-based compressive data gathering,” IEEE Communications Letters, vol. 13, no. 8, pp. 1572–1575, 2013.

[9] D. Ebrahimi and C. Assi, “Compressive data gathering using random projection for energy efficient wireless sensor networks,” Ad Hoc Networks, vol. 16, pp. 105–119, 2014.

[10] S. A. Daresari and J. Abouei, “Toward cluster-based weighted compressive data aggregation in wireless sensor networks,” Ad Hoc Networks, vol. 36, pp. 368–385, 2016.

[11] H. F. Zheng, F. Yang, X. H. Tian, X. Y. Gan, X. B. Wang, and S. Xiao, “Data gathering with compressive sensing in wireless sensor networks: a random walk based approach,” IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 1, pp. 35–44, 2015.

[12] M. T. Nguyen and K. A. Teague, “Compressive sensing based random walk routing in wireless sensor networks,” Ad Hoc Networks, vol. 54, pp. 99–110, 2017.

[13] J. J. Huang and B. H. Soong, “Cost-aware stochastic compressive data gathering for wireless sensor networks,” IEEE Transactions on Vehicular Technology, vol. 68, no. 2, pp. 1525–1533, 2019.

[14] P. Zhang and J. X. Wang, “On enhancing network dynamic adaptability for compressive sensing in WSNs,” IEEE Transactions on Communications, vol. 67, no. 12, pp. 8450–8459, 2019.

[15] W. Osamy, A. M. Khedr, A. Aziz, and A. A. El-Sawy, “Clustertree routing based entropy scheme for data gathering in wireless sensor networks,” IEEE Access, vol. 6, pp. 77372–77387, 2018.

[16] R. Manchanda and K. Sharma, “Energy efficient compression sensing-based clustering framework for IoT-based heterogeneous WSN,” Telecommunication Systems, vol. 74, no. 3, pp. 311–330, 2020.
[17] Y. Song, Z. G. Liu, X. L. He, and H. Jiang, “Research on data fusion scheme for wireless sensor networks with combined improved LEACH and compressed sensing,” *Sensors*, vol. 19, no. 21, p. 4704, 2019.

[18] J. H. Qiao and Y. Y. Zhang, “Compressive data gathering based on even clustering for wireless sensor networks,” *IEEE Access*, vol. 6, pp. 24391–24410, 2018.

[19] Q. Wang, D. Y. Lin, P. F. Yang, and Z. Q. Zhang, “An energy-efficient compressive sensing-based clustering routing protocol for WSNs,” *IEEE Sensors Journal*, vol. 19, no. 10, pp. 3950–3960, 2019.

[20] Y. C. Yuan, W. Liu, T. Wang, Q. Y. Deng, A. F. Liu, and H. B. Song, “Compressive sensing-based clustering joint annular routing data gathering scheme for wireless sensor networks,” *IEEE Access*, vol. 7, pp. 114639–114658, 2019.

[21] U. S. Pacharaney and R. K. Gupta, “Clustering and compressive data gathering in wireless sensor network,” *Wireless Personal Communications*, vol. 109, no. 2, pp. 1311–1331, 2019.

[22] C. Zhang, X. Zhang, O. Li, Y. P. Yang, and G. Y. Liu, “Dynamic clustering and compressive data gathering algorithm for energy-efficient wireless sensor networks,” *IEEE Journal of Distributed Sensor Networks*, vol. 13, no. 10, Article ID 155014771773890, 2017.

[23] K. C. Lan and M. Z. Wei, “A compressibility-based clustering algorithm for hierarchical compressive data gathering,” *IEEE Sensors Journal*, vol. 17, no. 8, pp. 2550–2562, 2017.

[24] A. Aziz, K. Singh, W. Osamy, and A. M. Khedr, “Effective algorithm for optimizing compressive sensing in IoT and periodic monitoring applications,” *Journal of Network and Computer Applications*, vol. 126, pp. 12–28, 2019.

[25] A. Aziz, W. Osamy, A. M. Khedr, A. A. El-Sawy, and K. Singh, “Grey Wolf based compressive sensing scheme for data gathering in IoT based heterogeneous WSNs,” *Wireless Networks*, vol. 26, no. 5, pp. 3395–3418, 2020.

[26] A. Salim, W. Osamy, A. M. Khedr, A. Aziz, and M. A.Mageed, “A secure data gathering scheme based on properties of primes and compressive sensing for IoT-based WSNs,” *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5553–5571, 2021.

[27] M. A. Mazaideh and J. Levendovszky, “A multi-hop routing algorithm for WSNs based on compressive sensing and multi-objective genetic algorithm,” *Journal of Communications and Networks*, vol. 23, no. 2, pp. 138–147, 2021.

[28] J. Y. Chen, F. Q. Zhou, Z. S. Guo, and J. W. Wan, “Compressed data collection method for wireless sensor networks based on optimized dictionary updating learning,” *IEEE Access*, vol. 8, pp. 205124–205135, 2020.

[29] P. Zhang, J. X. Wang, and W. J. Li, “A learning based joint compressive sensing for wireless sensing networks,” *Computer Networks*, vol. 168, p. 107030, 2020.

[30] M. Q. Zhang, H. X. Zhang, D. F. Yuan, and M. G. Zhang, “Compressive sensing and autoencoder based compressed data aggregation for green IoT networks,” in *IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, 2019.

[31] M. Q. Zhang, H. X. Zhang, D. F. Yuan, and M. G. Zhang, “Learning-based sparse data reconstruction for compressed data aggregation in IoT networks,” *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11732–11742, 2021.

[32] S. A. Sert, A. Alchihabi, and A. Yazici, “A two-tier distributed fuzzy logic based protocol for efficient data aggregation in multihop wireless sensor networks,” *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 6, pp. 3615–3629, 2018.

[33] G. S. Gandhi, K. Vikas, V. Ratnam, and K. S. Babu, “Grid clustering and fuzzy reinforcement-learning based energy-efficient data aggregation scheme for distributed WSN,” *IET Communications*, vol. 14, no. 16, pp. 2840–2848, 2020.

[34] S. P. Tirani, A. Avokh, and S. Azar, “WDAT-OMS: a two-level scheme for efficient data gathering in mobile-sink wireless sensor networks using compressive sensing theory,” *IET Communications*, vol. 14, no. 11, pp. 1827–1838, 2020.

[35] D. Ebrahimi, S. Sharafeddine, P. H. Ho, and C. Assi, “UAV-aided projection-based compressive data gathering in wireless sensor networks,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1893–1905, 2019.

[36] M. Sekine and S. Ikada, “LACSLE: Lightweight and adaptive compressed sensing based on deep learning for edge devices,” in *IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, 2019.

[37] M. Sekine and S. Ikada, “Adaptive cooperative distributed compressive sensing for edge devices: a multiagent deep reinforcement learning approach,” in 19th IEEE International Conference on Pervasive Computing and Communications (IEEE Per Com), Kassel, Germany, 2021.

[38] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.

[39] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.

[40] “Sea surface temperature (SST) v2,” 2015, http://www.esrl.noaa.gov/psd/data/gridded/data.noaa.oisst.v2.html.

[41] C. Lin, G. J. Han, X. Y. Qi, J. X. Du, T. T. Xu, and M. M. Garcia, “Energy-optimal data collection for unmanned aerial vehicle-aided industrial wireless sensor network-based agricultural monitoring system: a clustering compressed sampling approach,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4411–4420, 2021.