Load balancing for mitigating hotspot problem in wireless sensor network based on enhanced diversity pollen

Trong-The Nguyen a, Jeng-Shyang Pan b, Thi-Kien Dao a and Shu-Chuan Chu c

aDepartment Information Technology, Hai Phong Private University, Hai Phong, Vietnam; bCollege of Information Sciences and Engineering, Fuzhou University of Technology, Fuzhou, People’s Republic of China; cSchool of Computer Science, Engineering, and Mathematics, Flinders University, Adelaide, Australia

ABSTRACT
This paper proposes a load balancing to mitigate the hot spot problem in wireless sensor network (WSN), based on enhancing diversity pollens in Flower pollination algorithm (FPA). The hotspot problem in the WSN is spots near base station (BS) that consume more energy and drain out energy more quickly than other nodes farther from the BS. The spots near BS are hotter than other places due to the heavy traffic from the cluster members and other cluster heads (CH) for relaying data to BS. Enhancing diversity pollens for FPA is one of the solutions to deal smoothly with trapping local extrema for solving the hotspot problem. To evaluate the proposed algorithm, we firstly use a set of benchmark functions to test performance quality, and secondly, we deal with the load balancing problem in WSN. The results compared with some metaheuristic approaches and other related clustering algorithms demonstrate that the proposed method performs better than the others regarding various performance metrics such as the load balancing, execution time, energy consumption, and convergence rate.

ARTICLE HISTORY
Received 20 April 2017
Accepted 12 September 2017

KEYWORDS
Enhanced diversity pollen; flower pollination algorithm; load-balanced clustering; wireless sensor network

1. Introduction
Rapid development in the field of integrated circuits and information technology leads to the development of cheap and compact size sensor nodes (Chang & Huang, 2016). Wireless Sensor Network (WSN) is made up of a set of such sensor nodes arranged in an ad-hoc fashion to monitor and interact with the physical world. Applications of WSNs have been expanding enormously due to the inclusion of new emergency areas (García-hernández, Ibargüengoytia-gonzález, García-hernández, & Pérez-díaz, 2007). For example, there are requests for environmental monitoring such as the landslide, earthquakes, rivers, and agricultural monitoring (Dao, Pan, & Nguyen, 2015; Nguyen, Pan, Chu, Roddick, & Dao, n.d.). In these use cases, sensor nodes primarily are energy constrained. An energy-efficient method in WSN deployment is clustering, where the sensor nodes are organized into groups termed as clusters (Nguyen, Dao, Horng, & Shieh, 2016; Zhao & Chen, 2017). A selected node in the cluster as a leader is Cluster Head (CH), and the other nodes in...
that cluster are node members (CM). The cluster members cannot send the data directly to the base station (BS), but it sends only to the CH and CH forwards its aggregated data to BS. The advantages of clustering are: reducing energy consumption, improving bandwidth utilization, reduced overhead, increased connectivity, stabilized network topology, decreased delay, efficient load balancing, and reduction in the size of the routing table. The CHs near the BS consumes more energy and drains out energy more quickly than the CHs farther from BS. CHs closer to BS are loaded with heavy traffic due to: intra-cluster traffic from its cluster members, data aggregation, and inter-cluster traffic from other CHs for relaying data to BS. This results in disruption of the network connectivity and coverage issues that happen to the clusters closer to BS. This issue is referred to as hot spot problem.

Clustering technique is a way to balance energy consumption, which can deal with the hotspot problem in WSN (Chohra, Kanaoui, Amarger, & Madani, 2014; Nguyen, Shieh, Horng, Ngo, & Dao, 2015). There are traditional approaches in irregular grouping including the probabilistic, deterministic, and heuristic. Meanwhile, recently, metaheuristic have been widely applied to solve balanced load clustering (Kulkarni, Förster, & Venayagamoorthy, 2011), e.g. Differential Evolution (DE) (Kuila & Jana, 2014), Genetic Algorithm (GA) (Hussain, Matin, & Islam, 2007), and Load-balanced clustering of wireless sensor networks (LBC) (Gupta & Younis, 2003).

Flower pollination algorithm (FPA) (Yang, 2012) is a new metaheuristic approach that is an effective population-based intelligent optimization algorithm. FPA is inspired by mimicking flower pollination process, and two different ways of pollination as the self-pollination and cross-pollination are considered as local pollination and the global pollination process. Compared with the other algorithms for various clustering, several advantages of FPA are as follows. It is easy to implement by using programming language. Balancing between exploring and exploiting of FPA is like to adjust parameters for these pollination processes. It can generate high quality of the solution, and it converges to a solution fast.

Due to the physical constraint optimization problem, the global best result cannot always be obtained. For interference phenomena among bound dimensions, it is sometimes easily trapped on a local extremum for solving complex constraint optimization problems. The advantage in collaboration diversity agents is exploitation through the local extrema to the global optimum. Three main factors can be used to construct a dynamic diversity method, including subpopulations, neighbourhood topology, and its best historical information (Pham, 2015). The subpopulations could be figured out by dividing the population into groups. The area topology technique could be made up by applying some Niching techniques, such as the crowding, the fitness sharing available resources, and the speciation. The useful historical information obtained by each group evolving optimization could be used to communicate with each other among the groups. This paper is an extended version of our conference paper (Pan, Dao, Nguyen, Chu, & Pan, 2016), which includes the following new elements. The first, several crowds of pollens are deployed in optimization algorithm of FPA by dividing the population into subpopulations that are suitable to the distributed nodes randomly in WSN. The second, the optimal group of sensors in WSN as cluster heads (CHs) and the optimized parameters as neighbourhood topology are modelled as the fitness function for optimization. The third, a formulating fitness function of the load balancing in WSN is
executed by applying the enhanced diversity pollen in FPA. The final, a load balancing among the CHs to mitigate hotspot issue in WSN is figured out and the results compared with others methods in the literature are carried out to evaluate the performance of the proposed algorithm.

The rest of this paper is organized as follows. Related work as problem statement of load balancing and a brief review of FPA are given in Section 2. A methodology of the enhanced diversity FPA is presented in Section 3. Applied dynamic diversity FPA for load balancing for mitigating hotspot issue is discussed in Section 4. Finally, the conclusion is provided in Section 5.

2. Related work

2.1. Energy consumption model

We envision that the traffic supported in WSNs is for periodic data acquisition. A WSN consists of \( N \) uniformly distributed sensor nodes over a monitoring area. There is one BS connected to the user via Internet. Throughout this article, we let \( S = \{ s_1, s_2, \ldots, s_{N-1}, s_N \} \) denote the set of nodes, where \( s_i \) is node \( i \) and \( |S| = N \). Sensor nodes have no geographical perception, as their positions can only be determined, via information exchanging in the network. Sensor nodes are identical but unable to recharge their energy. Nodes can control their transmission power level. In each round, all sensor nodes collect and transmit data packets.

WSN uses wireless radio transceivers depending on the various parameters, e.g. distance and energy consumption. The distance between the transmitter and receiver obeyed on the attenuated transceiving power decreased exponentially with the increasing distance (Heinzelman, Chandrakasan, & Balakrishnan, 2002). Here, we denote \((x_i; y_i)\) and \((x_j; y_j)\) are the coordinates of nodes \( i \) and \( j \). Suppose node \( i \) sends \( l \) bits to destination node \( j \), the energy consumption on this transmission over distance \( d \) is calculated by Equation (1):

\[
E_{Tx}(l, d) = E_{Tx-elec}(l) + E_{Tx-amp}(l, d) = \begin{cases} 
   l \times E_{elec} + l \times \epsilon_{fs} \times d^2, & d < d_0 \\
   l \times E_{elec} + l \times \epsilon_{mp} \times d^4, & d \geq d_0 
\end{cases},
\]

(1)

where \( E \) is the consumed energy; \( Tx \) is for transmitting, elec, amp indicate electronic and amplify for digital coding, modulating, filtering, and spreading signal. It can be observed from Equation (1) that when the distance between two nodes is larger than the threshold \( d_0 \), much energy is consumed. This demonstrates that the multi-hop communication is more effective in WSNs. \( d_0 \) is a threshold of space model,

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

(2)

where the power loss \( \epsilon_{fs}, \epsilon_{mp} \) are for free space and multipath models. The energy consumption for receiving the messages can be expressed as

\[
E_{Rx}(l) = E_{Rx-elec}(l) = l \times E_{elec}.
\]

(3)

WSN assumed implemented \( N \) nodes in the area of two dimensions \( M^2 \) with \( k \) clusters. Dealing with the hotspot problem in WSN is to balance the load among the CHs based on the utilized different clustering techniques. Unequal Clustering reduces the size of
the clusters closer to BS and the cluster size increases as the distance between the BS and CH increases (Soro & Heinzelman, 2005). The cluster members sense the real world parameters and transmit the sensed value to its CH. CH receives and aggregates data to remove redundant data and transmit aggregated data to BS directly or via intermediate CHs (Abbasi & Younis, 2007). In equal clustering, the size of the cluster is same throughout the network. However, in unequal clustering, the cluster size is determined based on the distance to BS (Nguyen, Shieh, Dao, Wu, & Hu, 2013). The consumed energy of a cluster for a data frame is modelled as:

\[ E_{\text{cluster}} = E_{\text{CH}} + \left( \frac{N}{k} - 1 \right) \times E_{\text{member}}, \]  

(4)

where \( N/k - 1 \) is average of member nodes in a cluster; \( E_{\text{CH}}, E_{\text{members}} \) are the dissipated energy for CH and members respectively. They could be calculated as follows:

\[ E_{\text{member}} = l \times E_{\text{elec}} + l \times E_{\text{fs}} \times d^2_{\text{toCH}}, \]  

(5)

\[ E_{\text{CH}} = l \times E_{\text{elec}} \times \left( \frac{N}{k} - 1 \right) + l \times E_{\text{DA}} \times \frac{N}{k} + l \times E_{\text{elec}} + l \times E_{\text{mp}} \times d^4_{\text{toBS}}. \]  

(6)

Therefore the consumed energy for a WSN in a generation could formulate from the energy process and energy transceiver as:

\[ E_{\text{total}} = E_p + E_{\text{frame}} = E_p + k \times E_{\text{cluster}}, \]  

(7)

where \( E_p \) is consumed energy for microcontroller and supply voltage of node. It does not affect the optimizing processes. Therefore, the consumed energy \( E_{\text{frame}} \) could optimize based on the distance for clustering optimization.

2.2. Flower pollination algorithm

A recently population-based algorithm is known as FPA (Yang, 2012). FPA is drawn inspiration from two pollination processes of the flowering plant, including self-pollination and cross-pollination. In the flowering plant, pollen are transported by pollinators according to the rules of Lévy flights, and they can self-pollinate randomly. There are two universal concepts of exploring and exploiting search space in the population-based algorithm as the guidance of optimal process. A self-pollination of the flowering plant is viewed as local pollination that expressed for exploitation in the search area.

Because the flower pollination processes can occur at both local and global, to imitate this feature, the proximity probability \( p \) (denoted \( p \in [0, 1] \)) is to switch between natural global pollination to intensive local pollination. It means to change between the exploring and exploiting phases in FPA to control characteristics of local and global pollination. Let’s \( x^*_j \), \( x^*_k \) be solution vectors of the pollen, i.e. pollen in the same plant or the flowers. We could model for the local pollination as follows:

\[ x^*_j^{t+1} = x^*_j + u \times (x^*_j - x^*_k), \]  

(8)

where \( u \) is a random variable with a distributed uniform \([0, 1]\). In cross-pollination considered global, gametes of flower pollen are carried by pollinators, e.g. insects. Insects can often fly and move longer range. A Lévy flight can express flying insects over long
distances with various length steps, and be used to mimic this characteristic efficiently. Let
\( L \) be a Lévy distribution with active as drawn formula.

\[
L = \frac{\lambda \Gamma(\lambda) \times \sin(\frac{\pi \lambda}{2})}{\pi \times s^{\lambda}}, \quad (9)
\]

where \( \Gamma(\lambda) \) is the gamma function, and this distribution is valid for large steps \( s > 0 \). \( L(\lambda) \) is called the step size like the parameter that corresponds to the strength of the pollination. Updating solution vectors learned from the cross-pollination for global pollination are given as

\[
x_{i}^{t+1} = x_{i}^{t} + \gamma \times L(\lambda) \times (x_{j}^{t} - g^{*}), \quad (10)
\]

where \( g^{*} \) is the current best solution found so far, \( \gamma \) is a scaling factor to control the step size and \( t \) is the current generation or iteration. In the evaluation for updating pollination, according to the fitness function, if \( x_{j}^{t} \) and \( x_{k}^{t} \) come from the same plants or the same selected population, \( u \) becomes a local random walk. A probability switch \( p \) is set to 0.8 as a preliminary parametric might offer to performance some applications well [14]. The basic steps of FPA are described as pseudo code in Figure 1.

### Step 1. Initialization

Initial pollen population \( x = (x_{1}, x_{2}, \ldots, x_{d}) \) is generated randomly. A
switch probability \( p \in \) arrange from 0 to 1. A stopping criterion is set.

### Step 2. The best solution \( g^{*} \) is calculated with initial population, \( F_{\min} \) is assigned to

fitness at \( g^{*} \).

### Step 3. For each pollen in the population

- if \( \text{rand} < p \),
  - A step vector \( L \) is computed as Lévy distribution Eq.(9)
  - Global pollination is updated via Eq.(10)
- else
  - Draw \( u \) from a uniform distribution in \([0,1]\)
  - Local pollination is processed via Eq.(8)
  - end if

### Step 4. Evaluate new solutions, the function value \( F_{\text{new}} \) is assigned to fitness(\( x_{t+1} \)). A new solution is accepted if the solution improves (\( F_{\text{new}} \) less than \( F_{\min} \)), by updating the best solution \( g^{*} \) to \( x_{t+1} \) and assign the minimum function \( F_{\min} \) to \( F_{\text{new}} \).

### Step 5. If the termination condition does not meet, go to Step 3.

### Step 6. Output the best found solution.

**Figure 1.** Pseudo code of FPA.

3. Enhanced diversity pollen for FPA

#### 3.1. Methodology of diversity pollen

We deploy several crowds in optimization algorithm of FPA by dividing the population into
subpopulations, and we use the neighbourhood topology to share the available fitness
resources. Because of such subpopulations could evolve themselves independently in regular iterations to locate for the better area in the search space (Liang & Suganthan, 2005). The information is exchanged among subpopulations whenever the communication strategy is triggered to achieve the benefit of cooperation and exploitation. To adjust the proportion of the global and local searching processes, a dynamic switching probability strategy in the exchanging period is used to make the dynamic diversity. Once an activated schedule is set with a stage iteration, a new configuration of small groups would be started searching the promising area in the global target (Yang, 2012).

In FPA, a switching probability \( p \in [0, 1] \) is used to control the proportion of local search and global search, and it is a constant value. To produce dynamics in an algorithm, we use a switching parameter for sharing global and local search processes in the exchanging period. Switching probability \( p \) can transform as following formula:

\[
p = \alpha - \beta \times \frac{\tau - \text{mod}(\text{iter}, \tau)}{\tau},
\]

where \( \tau \) and iter are the exchanging period and current iteration, \( \alpha, \beta \) are constants in arrange of \([0,1]\) and \( \alpha > \beta \). \( p \) is a range of \([0,1]\), \( \alpha \) is set to 0.55 and \( \beta \) to 0.1. The fitness sharing available resources is commonly used in the Niching technique. Each group in FPA has its pollen as known the search agent and finest agents according to the fitness evaluation function. The finest agents among the pollen in a group will be assigned to the poorer agents in other groups, replace the inferior agents or update the inadequate agents for each crew after running period.

An enhanced diversity FPA is figured out based on both FPA optimization and Niching technique. We consider two characters in region structure: small size and communicating. Small size is used as small groups to create the diversity in local search. FPA with small areas performs better on some constrained problems with multimodal. The small-sized groups could be employed by dividing the population in FPA into subgroups. Each subset uses its members to search for a better area of the local in the search space. The communicating character, the better-obtained information of evolving optimization in each cluster, is exchanged among them to achieve the cooperative individuals and exploitation through the local extremes to the global optimum. Since the small-sized groups are searching using their best historical information, they are easy to converge to a local optimum.

Let \( n \) be number pollen of the group. It is called subpopulation size of the group. Let \( G_j \) be the group, where \( j \) is the index of the group and \( j \) is set to \( 0, 1, 2, \ldots, m \). Here, \( m \) is the number of groups. The top \( k \) fitness pollen in the group \( G_j \) according to objection function evaluation will be copied to \( G_{j+1} \) to replace the worst fitness pollen with the same number of pollen during run time with \( t \cap R \neq \emptyset \). The new configuration of the groups will be started searching in every \( R \) generations with dynamic switching probability triggered. The useful information could be obtained by exchanging among the groups. Thus, simultaneously the diversity of the population could perform better on the complex multimodal problems.

The steps of the diversity agent can be described as shown in Figure 2.
3.2 Simulation results

Six selected testing functions (Das & Suganthan, 2011; Jamil & Yang, 2013) are used to validate the performance metric of the proposed algorithm of the enhanced diversity FPA (namely dFPA). The obtained evaluation output values of the testing functions are averaged over 25 runs with different random seeds. The optimization goal is to minimize the outcome of the benchmark functions. Let vector $X = \{x_1, x_2, \ldots, x_n\}$ be $n$-dimensional real-valued.

The population size is set the same for all the algorithms of the proposed algorithm and original ones in the experiments. The details of parameter settings of FPA can be found in Yang (2012). Table 1 lists the initialization of boundary, dimension (Dim) and total iterations (MaxIters) for the testing functions.

The setting parameters for FPA and dFPA are as follows. The total population size $N$ is set to 80 (for dFPA is the total population size $n \times m$ is set to $20 \times 4$). The initial probability $p$ is set to 0.55. $\lambda$ is set to 1.5, and the dimension $d$ is set to 30. $\alpha$ and $\beta$ are set to 0.55 and 0.1, respectively. The exchanging period is set to 20 and the dimension $d$ is set to 30. Let MaxIters be maximum iterations. MaxIters can be set to 500, 1000, 2000, or 10,000 for each

| Testing functions | Boundary | Dim | MaxIters |
|-------------------|----------|-----|----------|
| $F_1(x)$          | $\pm 500$ | 30  | 1000     |
| $F_2(x)$          | $\pm 5.12$ | 30  | 1000     |
| $F_3(x)$          | $0, \pi$  | 30  | 1000     |
| $F_4(x)$          | $\pm 20$  | 30  | 1000     |
| $F_5(x)$          | $0.12$    | 4   | 1000     |
| $F_6(x)$          | $0.12$    | 4   | 1000     |

Figure 2. Pseudo code of enhanced diversity pollen for FPA.
function and the number of runs can be configured to 25. The obtained final results of proposed dFPA are averaged of the outcomes from all runs and compared with those obtained for the FPA.

Table 2 shows the comparison of the performance quality and execution time of the proposed algorithm with that of FPA for a set of selected optimization problems. The calculated value columns of RD (ratio deviation) and metric comparison in Table 2 are the percentage deviation of the proposed approach compared with the original method regarding running time and performance metric, respectively. The results of the proposed algorithm on all of these cases of testing problems show that the proposed algorithm provides a better performance than that obtained from original algorithm. The rate for a maximum case of derived from the proposed method increases greater than those obtained from FPA is up to 43%. However, the value for the minimum case increases by only 5%. The obtained average values for all testing cases of the proposed algorithm offer 24% increase in comparison with those obtained from the FPA. However, the execution time of them is similar.

Table 2. Comparison of quality performance metric and execution time of the proposed algorithm with FPA for a set of testing problems.

| Testing functions | Execution times (minutes) | Performance | Metric % comparison |
|-------------------|---------------------------|-------------|---------------------|
|                  | FPA | Proposed | RD % | FPA | Proposed | RD % |
| $F_1(x)$        | 0.836 | 0.8354 | 0% | $-6.08E + 03$ | $-6.39E + 03$ | 5% |
| $F_2(x)$        | 0.83273 | 0.8502 | 2% | $2.08E + 02$ | $1.69E + 02$ | 23% |
| $F_3(x)$        | 2.4783 | 2.4961 | 1% | $2.09E + 00$ | $1.46E + 00$ | 43% |
| $F_4(x)$        | 0.7776 | 0.7784 | 0% | $1.60E-03$ | $1.20E-03$ | 33% |
| $F_5(x)$        | 0.8918 | 0.9072 | 2% | $-3.24E + 00$ | $-3.89E + 00$ | 17% |
| $F_6(x)$        | 0.9593 | 0.9793 | 2% | $-9.15E + 00$ | $-1.07E + 01$ | 15% |
| Avge            | 1.1276 | 1.1296 | 1% | $-9.84E + 02$ | $-1.11E + 03$ | 24% |

Figure 3. Comparison the experimentally obtained results of the proposed dFPA with the FPA (the original FPA – oFPA) for the first function (F1).
Figures 3–5 indicate the experimentally obtained results of the proposed algorithm (dFPA) and FPA (oFPA) for the first three testing problems with the same setting condition, e.g. MaxIter is set to 1000 iterations. Four subfigures in the figures of Figures 3–5 included the function graph, semiology, convergence, and execution time. The plotted semiology measures indicated the performance through using a base 10 logarithmic scale with

Figure 4. Comparison the experimentally obtained results of the proposed dFPA with the FPA (the original FPA – oFPA) for the second function (F2).

Figure 5. Comparison the experimentally obtained results of the proposed dFPA with the FPA (the original FPA – oFPA) for the third function (F3).
their index containing real numbers. The plotted semiology measures indicated the performance through using a base 10 logarithmic scale with their index containing real numbers. The convergence criteria are the measure through using the convergence, and the time consumption is the execution time of dFPA and FPA methods for the particular optimization problems.

Apparently, the proposed dFPA performance quality for all of the cases of testing functions is higher than those obtained from FPA regarding the accuracy and convergence speed. However, the time requirement of two these methods are equal.

4. Load balancing for mitigating hotspot in WSN problem

The causes of the hotspot problem in WSN with multi-hop communications are unbalanced energy consumption. The higher energy consumed nodes are close to the BS because of the heavy traffic flows. In this section, the applied dFPA selects the optimal group of sensors as CHs and utilizes optimized parameters for objective function to form clusters with supposing of preventing unbalanced energy consumption. The optimized total communication distances from the cluster members to CHs and CHs to BS in WSNs can lead to an increase in saving energy. The experiments consists of the following steps: modelling objective function, describing proper pollen representation as mapping solution to CHs, optimizing CH, and comparing results.

4.1. Objective function

We assumed a WSN model with all randomly deployed sensor nodes along with a few CHs in the desired area. In communication range of a cluster, a node can assign to any CH within their coverage communicated range. Therefore, a particular sensor node can be configured to some pre-specified CHs. Thus each sensor node has a list of CHs, but it should be allocated to only one CH amongst them. The sensor nodes collect the local data and send it to their corresponding CHs. On receiving the data, the CHs aggregate them to reduce the redundant data within their cluster. Then CHs send the aggregated data to the BS directly or through the intermediate CHs.

Terminologies used in this section of balanced load clustering are as follows:

- A set of CHs denoted by symbol \( H = \{h_1, h_2, \ldots, h_m\}, n > m \).
- A set of sensor nodes denoted by symbol \( S = \{s_1, s_2, \ldots, s_n\} \).
- \( l_i \) indicates the load provided traffic of a sensor node \( s_i \), \( s_i \in S \).
- \( H_j \) indicates the set of CHs to which sensor node \( s_j \) may be assigned, where \( s_j \in S \) and \( H_j \subseteq H \).
- Let \( L_i \) be the CHs load \( h_i \). The overall load of each CH is calculated as \( L = \max\{L_i \mid \forall h_i \in H \} \).

The load traffic of the sensor nodes can be estimated prior with CHs of the clustering in WSN. However, the load balancing problem could be addressed with the primary objective to minimize the overall load of the CHs. Let \( a_{ij} \) be a Boolean variable such that \( a_{ij} = 1 \), if the sensor node \( s_j \) is assigned to the CH \( h_j \) and \( a_{ij} = 0 \), if it is not. Then the optimization problem of load balanced clustering (LBC) concerning Integer Linear Programming can be
formulated as follows:

$$\text{Min } L = \max \{L_i \mid \forall h_i \in H\}$$

Subject to

$$\sum_{h_j \in H} a_{ij} = 1 \mid \forall s_i \in S,$$

and

$$\sum_{s_j \in S} l_j \times a_{ij} \leq L \mid \forall h_i \in H_j,$$  \hspace{1cm} (13)

The constraints in Equation (12) indicate that a sensor node can be assigned to one and only one CH, and Equation (13) shows that load of CH with the burden of all the attached sensor nodes must not exceed the overall maximum traffic load of the CH. The average load and the standard deviation of the CH load are optimized to balance load networks by applying the clustering evaluation model for measuring the performance.

Table 3 shows an example of 4 CHs and 10 sensor nodes in a WSN, i.e. $H = \{h_1, h_2, h_3, h_4\}$, and $S = \{s_1, s_2, ..., s_{10}\}$. Thus, the dimension of the pollen position vector is the same as the number of sensor nodes, i.e. $D = 10$. All the sensor nodes and their possible CHs to which the sensor nodes can be assigned. The initial population of pollen vector is generated randomly. For example, random vector is as indicated in the first column in Table 3, $X^i_j = [0.45, 0.18, 0.38, 0.66, 0.86, 0.63, 0.23, 0.41, 0.34, 0.72]$. It should be noted that the pollen in clustering solution because the entire 10 sensor nodes are assigned to their corresponding CH. In this example, suppose $s_{01}$ is possible to connect to one of CH amongst $h_1, h_2$, and $h_3$. Similarly, $s_{02}$ selects $h_3$, or $h_2$ in $H_2$ and so on. The first element of pollen vector $x$ is 0.45, and Ceiling(x|Count(H_1)) = 2; therefore, the second CH from $H_1$, i.e. $h_2$, is selected for assigning $s_1$ as shown in Table 3. In the same way, all the sensor nodes are allocated to a CH using the randomly generated vector. The final assignment of the sensor nodes to their corresponding CHs is as shown in the last column of Table 3.

We build a fitness function to evaluate the pollen of the initial population as follows. It is noted that the load balancing of the CH not only minimizes the maximum CH load but also concentrates on the load distribution among all the CHs.

$$\mu(\text{avge load}) = \sum_{i=1}^{n} \frac{l_i}{m},$$  \hspace{1cm} (14)

| Pollen vector $X^i_j$ | Sensor node(s) ($s_i$) | Possible CH (H_i) | Count (H_i) | Ceiling ($X^i_j \times$ count (H_i)) | Assigned CH |
|------------------------|------------------------|-------------------|-------------|----------------------------------|-------------|
| 0.45                   | $s_{01}$               | $H_1 = \{h_1, h_2, h_3\}$ | 3           | 2                                | $h_2$       |
| 0.18                   | $s_{02}$               | $H_2 = \{h_1, h_2, h_3\}$ | 2           | 1                                | $h_1$       |
| 0.38                   | $s_{03}$               | $H_3 = \{h_2, h_3\}$ | 2           | 1                                | $h_3$       |
| 0.66                   | $s_{04}$               | $H_4 = \{h_1, h_2, h_3, h_4\}$ | 4           | 3                                | $h_3$       |
| 0.86                   | $s_{05}$               | $H_5 = \{h_2, h_3, h_4\}$ | 3           | 3                                | $h_3$       |
| 0.64                   | $s_{06}$               | $H_6 = \{h_1, h_2, h_3, h_4\}$ | 4           | 3                                | $h_3$       |
| 0.23                   | $s_{07}$               | $H_7 = \{h_2\}$ | 1           | 1                                | $h_2$       |
| 0.42                   | $s_{08}$               | $H_8 = \{h_1, h_2, h_3, h_4\}$ | 3           | 2                                | $h_2$       |
| 0.35                   | $s_{09}$               | $H_9 = \{h_1, h_3\}$ | 2           | 1                                | $h_1$       |
| 0.72                   | $s_{10}$               | $H_{10} = \{h_2, h_3, h_4\}$ | 3           | 3                                | $h_3$       |
where \( m \) and \( n \) are the number of CHs and sensor nodes, respectively. Therefore, we construct the fitness function by the standard deviation (s) of the CH load, which gives an even distribution of the load per cluster. The standard deviation of CH load is computed by

\[
\sigma(\text{CH load}) = \text{SQRT} \left( \frac{\sum_{j=1}^{m} (\mu - wL_j)^2}{m} \right),
\]

where \( wL_j \) is the overall load of the CH \( h_j \). Apparently, the less the standard deviation, the higher is the fitness value. Therefore, a chosen fitness function is with the reciprocal of the standard deviation of the CH load.

\[
\text{Fitness} = \frac{1}{\sigma(\text{CH load})}.
\]

### 4.2. Experimental results

Simulation of the network with \( N \)-node (\( N = 100,200.. \)) is distributed in a two-dimensional problem space [0:100,0:100]. In the target network, there are \( N \) deployed nodes in a \( n \times n \) grid space test platform is established, where nodes were randomly distributed between \((x = 0, y = 0)\) and \((x = n, y = n)\) with the Sink with \( N \) set to 100, 200, 300, and 400 nodes.

The initial values of communication energy parameters, \( E_j = 0.5 \) J, \( E_{elec} = 50 \) nJ/bit, \( e_{mp} = 0.0013 \) pJ/bit/m\(^4\), \( e_{fs} = 10 \) pJ/bit/m\(^2\), \( E_{DA} = 5 \) nJ/bit/signal, a number of initial CHs \( k = 10–20 \), \( l = 1024 \) bit (Heinzelman et al., 2002). The test case with considered various grid sizes can be effect to solution search rate. That means the number of sensor nodes was increased or decreased accordingly to different \( N \) values were verified to evaluate the effectiveness, time-liness, and reliability. Table 4 illustrates the cases of \( N \) from 100 to 400. It shows that by increasing the density of node in a cluster, the chances for nodes to become CH will be higher. The number of CHs is about 10.0% of the total number of nodes; the percentage may vary if nodes are unevenly distributed.

We figure the performance metric judge of the load balancing out, by calculating the standard deviation of the CH loads and the impact mean of the number of sensor nodes. The standard deviation of the CH load gives an even distribution of the load per CH. The fitness function in Equation (16) is repeatedly run the generations of 2000 with 25 runs. The final obtained results are taken as the average of the outcomes from all runs and then compared with the related clustering approaches, i.e. FPA (Yang, 2012), DE (Kuila & Jana, 2014), GA (Hussain et al., 2007), and LBC (Gupta & Younis, 2003). The algorithms are run for the same load of the sensor nodes by varying the sensor nodes from 100 to 300 and the number of CHs for 15 and 20. It is observed that our proposed method produces better load balancing for the equal load of the sensor nodes than others, as shown in Figure 6(a,b).

### Table 4.

| Number of nodes | Convergence after the generations | CH (%) | Solution search rate (%) |
|-----------------|----------------------------------|--------|--------------------------|
| 100             | 180                              | 10.0   | 97                       |
| 200             | 350                              | 10.0   | 99                       |
| 300             | 450                              | 10.0   | 98                       |
| 400             | 640                              | 11.2   | 96.2                     |
Figure 7 compares the proposed method of load balancing base on different clustering in WSN with the FPA, DE, GA, and LBC methods regarding convergence. The proposed method performs well for the uneven load for this case too.

Figure 8. Comparison of the convergence rate and power consumption of the proposed method with FPA, DE, GA, and LBC approaches. (a) The convergence rate. (b) The average residual energy.
Figure 8(a) shows the comparison of the convergence rate of our proposed method with FPA, DE, GA, and LBC approaches. We ran our algorithm for 200 sensor nodes and 20 CHs. Apparently, the proposed method shows the faster convergence than the other algorithms. Figure 8(b) demonstrates the comparison of the average residual energy of the proposed approach with FPA, DE, GA, and LBC approaches for a network with 200 sensor nodes and 10 percentages of it sensor nodes. Apparently, the proposed method offers the bigger number of the rounds than other methods.

Figure 9 shows the comparison of a number of rounds at first CH die, and the execution time of the proposed method with FPA, DE, GA, and LBC methods. The proposed method produces the better performance regarding the execution time and the number round at first CH die than other methods.

5. Conclusion

In this paper, we proposed an enhanced diversity pollens in FPA for the load balancing in WSN and the global optimization. The significance of the dynamic diversity population is to avoid the easy loss of the global optimum nearby, rather converge to a local optimum of the optimal algorithms for solving the complex constrained optimization problems. Several evolution subpopulations from FPA and the neighbourhood topology technique are used to model the diversity pollen. The new promising area is created after triggered communication among subgroups.

A dynamic switching between global and local search in each exchanging period provides the diversity of searching process. In experiment sections, a balanced load clustering for mitigating hotspot issue in WSN and a set of selected optimization problems are used to evaluate the accuracy, and the computational time of the proposed method. According to the experimental results for the testing functions, the proposed method is 24% faster in regarding the convergence than FPA. For the load balancing clustering issue, the proposed algorithm offers the better performance than the other based clustering algorithms, such as GA, DE, FPA, and LBC algorithm regarding balanced load of the CHs for the equal as well as the unequal clusters in WSN.
Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by Flinders University, Fujian University of Technology, Hai Phong Private University.

ORCID

Trong-The Nguyen http://orcid.org/0000-0002-6963-2626
Jeng-Shyang Pan http://orcid.org/0000-0002-2354-5834
Thi-Kien Dao http://orcid.org/0000-0002-2805-652X

Notes on contributors

Trong-The Nguyen received the Doctorate degree in Communication Engineering from the National Kaohsiung University of Applied Sciences, Taiwan in 2016. He is currently a lecturer with the Department of Information Technology, Haiphong Private University, Vietnam. His research interests include Computational Intelligence and Sensor Networks.

Jeng-Shyang Pan obtained his B.S. degree in Electronic Engineering from the National Taiwan University of Science and Technology in 1986, the M.S. degree in Communication Engineering from the National Chiao Tung University, Taiwan in 1988, and the Ph.D. degree in Electrical Engineering from the University of Edinburgh, U.K. in 1996. Currently, he is a Dean for College of Information Science and Engineering, Fujian University of Technology, also a professor of Fuzhou University of International Studies and Trade and a director of Innovative Information Industry Research Center, Harbin Institute of Technology Shenzhen Graduate School, China. He joined the editorial board of LNCS Transactions on Data Hiding and Multimedia Security, Journal of Computers, Journal of Information Hiding and Multimedia Signal Processing and so on. His current research interests include soft computing, information security and signal processing.

Thi-Kien Dao obtained her bachelor and master degree from the Faculty of Information Technology, College of Technology of Vietnam National University, Hanoi in 2001 and 2007, respectively. She is currently a Ph.D. student in the Department of Electronic Engineering, National Kaohsiung University of Applied Sciences, Taiwan. Her research interests include Computational Intelligence and Robot Sensor.

Shu-Chuan Chu achieved her Doctorate degree from Flinders University, Australia. Currently, she is a Research fellow with School of Computer Science, Engineering, and Mathematics, Flinders University, Adelaide, Australia. Her current research interests include Pattern Recognition and Data Mining, Computational Intelligence, and Information Hiding.

References

Abbasi, A. A., & Younis, M. (2007). A survey on clustering algorithms for wireless sensor networks. Computer Communications, 30(14–15), 2826–2841. doi:10.1016/j.comcom.2007.05.024
Chang, F.-C., & Huang, H.-C. (2016). A survey on intelligent sensor network and its applications. Journal of Network Intelligence, 1(1), 1–15.
Chohra, A., Kanaoui, N., Amarger, V., & Madani, K. (2014). Hybrid intelligent diagnosis approach based on soft computing from signal and image knowledge representations for a biomedical application. Vietnam Journal of Computer Science, 1(3), 155–167. doi:10.1007/s40595-014-0017-6
Dao, T., Pan, T., & Nguyen, T. (2015). A compact artificial bee colony optimization for topology control scheme in wireless sensor networks. *Journal of Information Hiding and Multimedia Signal Processing, 6*(2), 297–310.

Das, S., & Suganthan, P. N. (2011). Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real-world optimization problems. *Electronics, 1*–42.

García-hernández, C. F., Ibargengoytia-gonzález, P. H., García-hernández, J., & Pérez-díaz, J. A. (2007). Wireless sensor networks and applications: A survey. *Journal of Computer Science, 7*(3), 264–273. doi:10.1109/MC.2002.1039518

Gupta, G., & Younis, M. (2003). Load-balanced clustering of wireless sensor networks. IEEE international conference on communications. ICC '03., 3, 1848–1852. doi:10.1109/ICC.2003.1203919

Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications, 1*(4), 660–670. doi:10.1109/TWC.2002.804190

Hussain, S., Matin, A. W., & Islam, O. (2007). Genetic algorithm for hierarchical wireless sensor networks. *Journal of Networks, 2*(5), 87–97. doi:10.4304/jnw.2.5.87-97

Jamil, M., & Yang, X.-S. (2013). A literature survey of benchmark functions for global optimization problems. *International Journal of Mathematical Modelling and Numerical Optimisation, 4*(2), 150–194. doi:10.1504/IJMMNO.2013.055204

Kuila, P., & Jana, P. K. (2014). A novel differential evolution based clustering algorithm for wireless sensor networks. *Applied Soft Computing, 25*, 414–425. doi:10.1016/j.asoc.2014.08.064

Kulkarni, R. V., Förster, A., & Venayagamoorthy, G. K. (2011). Computational intelligence in wireless sensor networks: A survey. *IEEE Communications Surveys and Tutorials, 13*(1), 68–96. doi:10.1109/SURV.2011.040310.00002

Liang, J. J., & Suganthan, P. N. (2005). Dynamic multi-swarm particle swarm optimizer with local search. *2005 IEEE Congress on Evolutionary Computation, 1*(1), 522–528. doi:10.1109/CEC.2005.1554727

Nguyen, T.-T., Dao, T.-K., Horng, M.-F., & Shieh, C.-S. (2016). An energy-based cluster head selection algorithm to support long-lifetime in wireless sensor networks. *Journal of Network Intelligence, 1*(1), 23–37.

Nguyen, T.-T., Pan, J.-S., Chu, S.-C., Roddick, J. F., & Dao, T.-K. (n.d.). Optimization localization in wireless sensor network based on multi-objective firefly algorithm. *Journal of Network Intelligence, 1*(4), 130–138.

Nguyen, T.-T., Shieh, C.-S., Dao, T.-K., Wu, J.-S., & Hu, W.-C. (2013). *Prolonging of the network lifetime of WSN using fuzzy clustering topology*. Proceedings – 2013 2nd international conference on robot, vision and signal processing, RVSP 2013. doi:10.1109/RVSP.2013.10

Nguyen, T.-T., Shieh, C.-S., Horng, M.-F., Ngo, T.-G., & Dao, T.-K. (2015). Unequal clustering formation based on bat algorithm for wireless sensor networks. In V.-H. Nguyen, A.-C. Le, & V.-N. Huynh (Eds.), *Knowledge and systems engineering: Proceedings of the sixth international conference KSE 2014* (pp. 667–678). Cham: Springer International.

Pan, J.-S., Dao, T.-K., Nguyen, T.-T., Chu, S.-C., & Pan, T.-S. (2016). *Dynamic diversity population based flower pollination algorithm for multimodal optimization*. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics) (Vol. 9621). doi:10.1007/978-3-662-49381-6_42

Pham, H. A. (2015). Reduction of function evaluation in differential evolution using nearest neighbor comparison. *Vietnam Journal of Computer Science, 2*(2), 121–131. doi:10.1007/s40595-014-0037-2

Soro, S., & Heinzelman, W. B. (2005). Prolonging the lifetime of wireless sensor networks via unequal clustering. Proceedings – 19th IEEE international parallel and distributed processing symposium, IPDPS 2005 (Vol. 2005). doi:10.1109/IPDPS.2005.365

Yang, X. S. (2012). *Flower pollination algorithm for global optimization*. Lecture Notes in Computer Science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics) (Vol. 7445 LNCS, pp. 667–678). Cham: Springer International.

Zhao, L., & Chen, N. (2017). An effective clustering routing protocol for heterogeneous wireless sensor networks. *Journal of Information Hiding and Multimedia Signal Processing, 8*(3), 723–733.