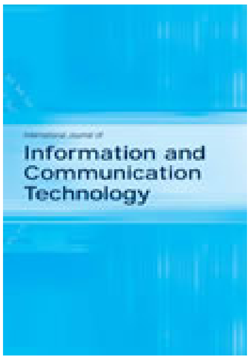


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

A wireless sensor network node redeployment method based on improved leapfrog algorithm

Bin Zhang, Xinhua Wang

DOI: 10.1504/IJICT.2022.10047449

Article History:

Received:	06 October 2021
Accepted:	28 October 2021
Published online:	05 December 2023

A wireless sensor network node redeployment method based on improved leapfrog algorithm

Bin Zhang*

Shangqiu Polytechnic,
Shangqiu 476000, China
Email: binzh@36haojie.com
*Corresponding author

Xinhua Wang

Henan Police College,
Zhengzhou 450046, China
Email: xhwang@36haojie.com

Abstract: In order to overcome the problems of large errors and low average coverage of nodes in traditional node redeployment methods, a node redeployment method based on improved frog jump algorithm is designed in this paper. The output node path of wireless sensor network is determined by constructing the distribution node deployment model. Then the leapfrog algorithm is improved by introducing virtual force algorithm, and the physical model of node deployment area is established, so as to optimise the node deployment process by using gravity and repulsion force and redeploy static and dynamic nodes. Experiments show that the minimum node redeployment error of this method is only 0.01. When the energy of some nodes is exhausted, it still has relatively good coverage quality performance, and the average coverage rate reaches 75%, which proves that it not only ensures the network coverage quality, but also reduces the number of working nodes.

Keywords: improved leapfrog algorithm; wireless sensor network; node redeployment; node path.

Reference to this paper should be made as follows: Zhang, B. and Wang, X. (2024) 'A wireless sensor network node redeployment method based on improved leapfrog algorithm', *Int. J. Information and Communication Technology*, Vol. 24, No. 1, pp.33–47.

Biographical notes: Bin Zhang received his Master's degree in Software Engineering from Sichuan University in 2010. Currently, he is an Associate Professor of Shangqiu Polytechnic. His research interest is in mathematical statistical analysis.

Xinhua Wang received his Master's degree in Thermal Power Engineering from Changsha University of Science and Technology in 2010. He is a Professor of Henan Police College. His research interests are in heat transfer enhancement and intelligent control technology.

1 Introduction

Through the wireless communication technology, the micro sensor nodes form the wireless sensor network through the topological association, which plays an important role in regional information perception and acquisition processing (Halakarnimath and Sutagundar, 2020; Alassery, 2019). In order to ensure the normal operation of wireless sensor network, the process of node deployment is very important. Sensor node deployment is the organic combination of front-end information collection and back-end information processing in wireless sensor network system, and it is the basic problem of wireless sensor network. Generally speaking, node deployment in sensor network can be divided into two parts, namely static deployment and dynamic deployment. The overall goal is to improve network sensing coverage (Huan et al., 2020). In most cases, the environment in which wireless sensor network exists is usually unknown and harsh, and the working environment is usually toxic or disaster-stricken area. Therefore, universities need to deploy algorithms to move sensors to ensure the coverage of wireless sensor nodes (Hajizadeh et al., 2020; Guo et al., 2019).

To this end, Lian (2019) proposes a wireless network node reorganisation method based on improved ant colony algorithm. By constructing a wireless network topology model, the initial nodes of the network are topologically distributed, the linear structure of the nodes is decomposed, and the characteristic quantity of the transmission channel of the wireless sensor network is extracted. Combining the decomposition of spectrum characteristics, the wireless sensor network topology is reorganised to realise the distributed optimisation of node deployment and improve the transmission performance of the wireless sensor network. This method can improve the stability of the wireless network, and the node deployment performance is improved, but there is a problem of large redeployment errors. He et al. (2019) propose an optimised deployment method for wireless sensor network nodes based on improved sine cosine algorithm. Combining the hyperbolic sine adjustment factor and the dynamic cosine weight coefficient, the global search of the sensor network is performed based on the balance algorithm. In order to ensure that the algorithm does not fall into the local optimal situation, the network search optimisation is carried out by combining Laplacian and Gaussian distribution. Bringing this method into node deployment optimisation can significantly improve node coverage, use fewer working nodes, and ensure the same deployment accuracy. However, there is a problem that the average node coverage is low. Sun et al. (2019) propose a multi-node joint deployment algorithm for sensor networks based on a probabilistic perception model. They introduce related research in the field of probability to reduce coverage blind areas in the deployment of sensor networks through the node dynamic conversion mechanism. They optimise the deployment of nodes to ensure that the network consumption is within a certain range and the network life cycle can be extended. Compared with the traditional network node coverage method, the coverage rate has been significantly improved to ensure the deployment performance of wireless sensor network nodes. However, there is a problem of a high number of working nodes.

In view of the shortcomings of traditional methods, this study designed a wireless sensor network node redeployment method based on the improved leapfrog algorithm. The design idea is as follows:

1 the path of output node of WSN is determined by constructing distribution node deployment model, which lays the foundation for subsequent node deployment

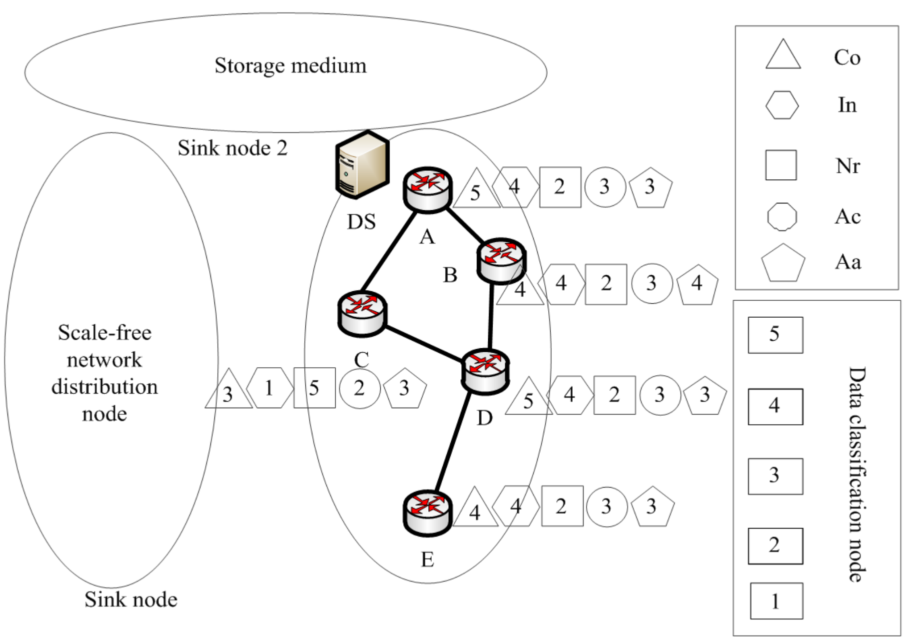
2 the virtual force algorithm is introduced to improve the leapfrog algorithm, and the physical model of the node deployment area is established by combining the improved algorithm, so as to optimise the node deployment process by using gravity and repulsion forces and fundamentally reduce the node redeployment error

3 in the sensor system of wireless network, static and dynamic nodes are redeployed based on the improved leapfrog algorithm.

2 Wireless sensor network node redeployment

In the process of wireless sensor network node redeployment, it is necessary to build a node redeployment model to perform distributed control on the nodes that have not been redeployed, and redeploy the nodes through the channel model and transmission protocol mechanism. Under different topological degrees, several robust coefficients of wireless sensor network neighbours are calculated (Nasri et al., 2020), and the topology structure of network node redeployment is shown in Figure 1.

Figure 1 Node redeployment model of wireless network sensor path (see online version for colours)



In Figure 1, the node redeployment method of the wireless sensor network mainly relies on the wireless sensor networking form, which controls the cluster routing topology of the nodes that have not been redeployed, and combines the associated feature detection method to schedule low-power cluster nodes. Realise optimised design during transmission (Du et al., 2019). The utility of nodes without redeployment is measured based on the node connectivity index. Assuming that the redeployment range the sensing position of wireless network sensor is a square, the side length is set to M, and there are N routing relay nodes without redeployed network nodes. The routing in the sensor network is selected through the distributed control protocol, and the routing nodes are adaptively distributed and controlled. For any sensor network without redeployment node i, the source and sink nodes can be obtained through compilation. Since the node transmission time interval T_f is limited, it is assumed that the output channel width of the node is $T_s = N_f T_f$, combined with the transmission bandwidth T_p of adjacent nodes in the network. Divide the frame of any cluster head node that has not been redeployed. There are a total of N_c chips. The energy consumption of the node is calculated. The specific expression is as follows:

$$T_c = ent\left(T_f / N_c\right) \tag{1}$$

In the process of node redeployment, if the redeployed node meets the conditions of $c_j T_c < T_f$, $\forall j \in [0, N_f - 1]$, in wireless network sensor lines, the node redeployment path is expressed based on fuzzy optimisation control, and the expression is as follows:

$$\begin{cases} x = \left(x_1, x_2, \ldots, x_n\right) \\ y = F(x) = \left(f_1(x), f_2(x), \ldots, f_m(x)\right)^T \\ g_i(x) \le 0, i = 1, 2, \ldots, q \\ h_i(x) = 0, i = 1, 2, \ldots, p \end{cases} \tag{2}$$

In the formula, $x = (x_1, x_2, \ldots, x_n$ represents the routing vector in the n-dimensional wireless sensor network without redeployment node scheduling.

There is a certain connection between the non-re-deployed node A and the associated node A, which can be expressed by the associated scale. The specific expression is as follows:

$$s(x) = \sum_i b_j \sum_{j=0}^{N_f - 1} p\left(t - iT_s - jT_d - c_j T_c\right) \tag{3}$$

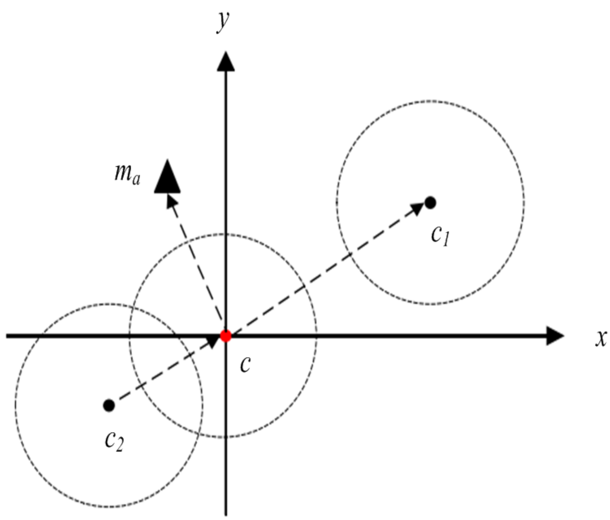
In the above formula, b_j represents the path loss of the collection and distribution nodes of different paths in sensors in wireless networks, T_s represents the attenuation value during the transmission process of the collection and distribution nodes in the wireless sensor network, and T_d represents the data decentralised control process of the sensors in wireless networks. The collection time of the distribution node, T_c represents the load balance value of by sub matching the nodes of sensors in wireless networks. Based on the sensors in wireless networks, combined with the extreme value of the convergence tree, the network routing is topologically controlled to ensure the accurate performance of network node redeployment to a certain extent.

3 Improved leapfrog algorithm

The leapfrog algorithm integrates the memetic-based algorithm of particle swarms, has good computing power, and has better search performance for the whole world. Due to the premature nature of the leapfrog algorithm itself, it is re-deployed in wireless sensor network nodes. It is easy to fall into local search, and it is difficult to solve the global optimal solution (Singh, 2019). This paper introduces virtual force algorithm to improve the leapfrog algorithm, establishes a physical model of node redeployment area, forms repulsion and gravity, and optimises node redeployment in the network.

The key of the virtual force algorithm is to imitate the force of physical charges. It is assumed that there is a certain gravitational or repulsive force between any node and other nodes in the redeployment area. Based on this interaction, the connection between the sensor network nodes is guaranteed and the nodes are effectively avoided. The distance is too far or too close to avoid the phenomenon of network nodes being separated or repeated coverage. The preset threshold d_{th} expresses the interaction force between network nodes, and the virtual force model is shown in Figure 2.

Figure 2 Node virtual force model (see online version for colours)



Sensors in wireless networks, the calculation method of the virtual force of any node *c* is as follows:

$$\vec{F}_c = \vec{F}_{c1} + \vec{F}_{c2} + S_{ma} \quad (4)$$

In the formula, $\vec{F}_{c1}$ and $\vec{F}_{c2}$ represent the gravitational and repulsive forces of the nodes $c1$ and $c2$, respectively, and S_{ma} represents the gravitational force received by the node in the node redeployment area.

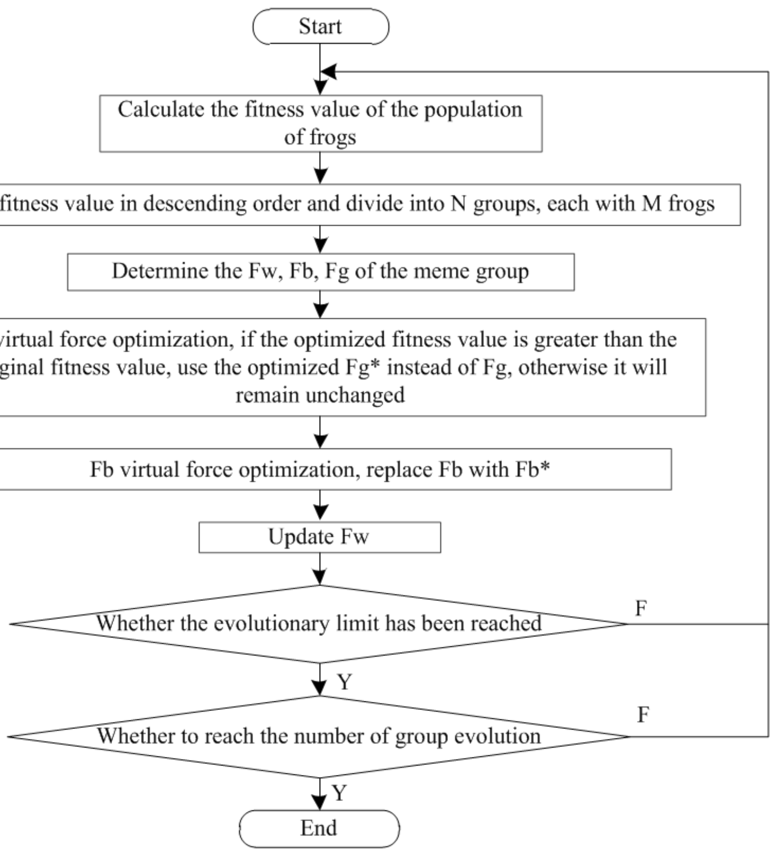
In the process of frog population evolution, virtual force evolution is carried out for the population optimal solution. Assuming that the progressive solution obtained after the evolution is better than the current solution, the current population optimal solution is replaced by the progressive solution, otherwise no replacement is used. Through this replacement, the convergence performance of the global optimal solution is guaranteed, the cycle in the redeployment zone is broken, and different node groups can quickly jump out of the local extreme value, ensuring the global search.

The specific expression of node redeployment position is as follows:

$$(x, y)_{new} = \begin{cases} (x, y)_{old}, & \text{if } |F_{xy}| \le F_{th} \\ (x, y)_{new} + \dfrac{(F_x, F_y)}{F_{xy}} \times MaxStep \times e^{\frac{-1}{F_{xy}}}, & \text{if } |F_{xy}| > F_{th} \end{cases} \tag{5}$$

In the formula, F_{xy} represents the virtual force received by the wireless sensor, F_x and F_y represent the virtual force components in the x and y axis directions, respectively, *MaxStep* represents the maximum displacement of the sensor movement, and F_{th} represents the virtual force threshold. The process of virtual force to improve leapfrog algorithm is shown in Figure 3.

Figure 3 Virtual force improved leapfrog algorithm



The detailed description of the improved leapfrog algorithm process is as follows:

1 Leapfrog algorithm parameters need to be initialised. The parameters mainly include the maximum number of population iteration *MaxIter*, the population size *N*, the number of subgroups *m*, the number of subgroup evolutions *M*, and the maximum distance of frog movement *DMax*.

2 The frog population *N* is generated randomly.

3 Using a grouping algorithm, divide the frog population into m subgroups, and place N frogs in the m memetic groups in descending order of appropriate values, and stop when the m frog enters the m group, and the $m + 1$. The first group of frogs today, the population distribution ends. F_w and F_b respectively represent the worst and best frogs in the group, and F_g represents the best frog in the population.

4 Update the local position of the subgroup in real time, and adjust the worst frog position. The specific adjustment method is as follows:

The distance the frog moves during the partial update:

$$D = rand\left(F_g\right)\times\left(F_b - F_w\right) \tag{6}$$

Assuming that compared with the original position, the updated position is not at the optimal position, continue the global update, and randomly update the worst frog position:

$$F_w = F_w + D\left(D_{\max} \geq D \geq -D_{\max}\right) \tag{7}$$

5 After M subgroup evolutions, all subgroups are mixed to form a new population;

6 Determine whether the optimal solution is output through several iterations. If it is the maximum number, output the optimal solution; otherwise, return to step 3.

4 Redeploy wireless sensor network nodes based on the optimised frog jump algorithm

Using the wireless sensor network architecture, combined with the appropriate distance exchange mechanism, the non-re-deployed nodes in the sensor network and the neighbouring nodes are grouped according to the positional relationship. To ensure that the nodes in the sensor network meet the minimum number of hops, the wireless sensor network structure system is shown in Figure 4.

In Figure 4, L_1, L_2 and L_3 represent three reference nodes with known positions, and the rest are unknown nodes. The unknown node A is located through the three reference nodes.

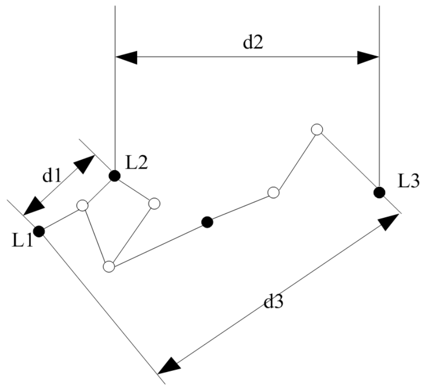
For any reference node in the sensor network, the distance per hop can be calculated through the node location information and the minimum hop count. Assuming that there is a node A to be redeployed, the distance between the node and the reference nodes L_1, L_2 and L_3 can be expressed as the product of the minimum number of hops and the average length of each hop. The specific formula is as follows:

$$\begin{cases} \hat{d}_1 = 3\times HS_1^D \\ \hat{d}_2 = 2\times HS_2^D \\ \hat{d}_3 = 3\times HS_3^D \end{cases} \tag{8}$$

Among them, the distance between unknown node A and reference target node L_1 is represented by $\hat{d}_1$, the distance between unknown node A and reference target node L_2 is represented by $\hat{d}_2$, and the distance between unknown node A and reference target node

L_3 is represented by $\hat{d}_3$; HS_1^D, HS_2^D and HS_3^D represent the distance from the reference node L_1, L_2 and L_3 are the base positions, and the average length of each hop is obtained.

Figure 4 Wireless sensor network architecture



4.1 Static node redeployment

Only a few nodes are dynamic in wireless sensor networks (Yang et al., 2020). All nodes need to be reasonably redeployed, through the GPS redeployment device, the unknown node location is redeployed based on the beacon node, and the static node is set. First determine the interval between the static node and each branch node, the interval value is the smallest. In the wireless sensor network, according to the number of nodes and the number of intervals between nodes, the average minimum error between unknown nodes is estimated to complete the redeployment of static nodes. Here are the specific steps:

Assuming that the coordinates of three nodes are known to be represented by (x_a, x_b), (x_b, y_b) and (x_c, y_c), respectively, and d_a, d_b and d_c are used to represent the distances from all nodes to node (x, y), respectively. The static redeployment process of the wireless sensor network is as follows:

$$\begin{cases} \sqrt{(x-x_a)^2+(y-y_a)^2}=d_a \\ \sqrt{(x-x_b)^2+(y-y_b)^2}=d_b \\ \sqrt{(x-x_c)^2+(y-y_c)^2}=d_c \end{cases} \tag{9}$$

When the static node redeployment of the wireless network induction is completed, the coordinates of the static node D are calculated according to formula (10) as:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2(x_a-x_c)2(y_a-y_c) \\ 2(x_b-x_c)2(y_b-y_c) \end{bmatrix}^{-1} \begin{bmatrix} x_a^2-x_c^2+y_a^2-y_c^2+d_c^2-d_a^2 \\ x_b^2-x_c^2+y_b^2-y_c^2+d_c^2-d_b^2 \end{bmatrix} \tag{10}$$

Therefore, it can be concluded that the traditional redeployment algorithm regards the original distance between static nodes as the actual distance between static nodes, and does not fully consider the probability effect between nodes, resulting in low redeployment accuracy (Domga et al., 2019). For this reason, it is necessary to find the difference between the original distance and the actual distance according to the original distance matrix to compensate for the redeployment error.

The algorithm re-deploys static nodes in wireless network induction, iteratively optimises known static nodes, and adapts the average distance to the increase of static nodes, thus realising the static re-deployment of wireless sensor networks. The following are the specific steps:

For the wireless sensor network static redeployment, it is assumed that $A(x, y)$ is the coordinate of the unknown static node, and the triangle coordinates $L_1(x_1, y_1)$, $L_2(x_2, y_2)$, $L_k(x_k, y_k)$ of the beacon node corresponding to the received A. The average values of the jumping distances of the beacon nodes corresponding to the received 666 are h_1, h_2 and h_k.

In order to solve the relationship between beacon nodes, hop_1, hop_2, …, hop_k represents the number of hops between beacon nodes, A represents the distance between nodes, and the distance between beacon nodes A to L_i can be as follows:

$$r_i = h_i \times hop_i \tag{11}$$

Assuming that the distance between the beacon node L_i and the static node d_i is known, the weight distance between the static nodes is as follows:

$$d_i = \left(\sum_{i=1}^{1-\frac{r_i}{k}} r_i \right) \times r_i \tag{12}$$

Compared to r_i, the distance between the static node A and the beacon node L_i in the wireless sensor network is d_i. Wireless network induction, by calculating the weight distance of static nodes, the coordinates of static nodes can be calculated as follows:

$$X = \left(A^T, A \right)^{-1} A^T b \tag{13}$$

The following formula can express the coordinate process of static node redeployment in wireless sensor network:

$$A = -2 \begin{Bmatrix} x_1 - x_k & \cdots & y_1 - y_k \\ \vdots & \ddots & \vdots \\ x_{k-1} & x_k & y_{k-1} \end{Bmatrix} \tag{14}$$

When redeploying static nodes in wireless sensor networks, the distribution of beacon nodes and static nodes in wireless sensor networks is random. According to the above formula, the static node can calculate the actual redeployment of the static node, thus realising the redeployment of the static node of the wireless sensor network.

4.2 Dynamic node redeployment

The existing dynamic redeployment methods of wireless sensor nodes almost use the node time spent in the operation of the Internet to control the time series of the target nodes. By establishing the model of node mobility compensation in wireless sensor networks, the network has a certain delay, which has an impact on the presumed node redeployment. Redeployment is prone to errors, which leads to the reduction of the accuracy of dynamic node redeployment, and the requirements for redeployment nodes cannot meet the required technology (Li and Tu, 2020).

Set the node target dynamic movement vector of wireless network sensor as:

$$x = [x, \dot{x}, \ddot{x}]^T \tag{15}$$

When the established standard value state is $\ddot{x}(t) = 0,$ combined with the wireless sensor network operation log to confirm whether the network state is an absolute constant value, (Wei and Xiu, 2020), when the determination result is no, the offset white noise is:

$$\ddot{x}(t) = w(t) \tag{16}$$

According to formula (16), it can be seen that the closer the offset acceleration is to the constant value, the smaller the variance *q* of *w*(*t*) will become. At this time, the time series calculation method between dynamic nodes of wireless sensor is as follows:

$$\begin{aligned} \dot{x}(t) &= \varphi w(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t), \\ \varphi &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{17}$$

w(*t*) in formula (17) is a parameter in covariance matrix *q*. The final calculation of this matrix can take a constant value. When the dynamic node target is redeployed, the angular travelling speed of target deviation can be expressed as $\sqrt{Q_{33}} = \sqrt{qt}.$ The calculation process of the offset angular speed of the dynamic target node is as follows:

$$\begin{aligned} x(k+1) &= Fx(k) + w(k) \\ z(k) &= Hx(k) + v(k) \end{aligned} \tag{18}$$

In the above formula, *x*(*k*) represents the dynamic node state of time *k*, *H* represents the redeployed measurement matrix, *z*(*k*) represents the dynamic measurement of the node offset angle of time *k*, *F* represents the matrix of state offset, and *w*(*k*) represents the matrix of covariance.

To sum up, this paper introduces the virtual force algorithm to improve the leapfrog algorithm, establishes a physical model of node redeployment area, and constructs new optimisations for node redeployment by gravity and repulsion. In the induction system of wireless network, static and dynamic nodes are re-deployed based on the improved leapfrog algorithm. The method in this paper improves the leapfrog algorithm and combines the interaction of virtual force models to ensure the connection between sensor network nodes and effectively avoid nodes. The distance between the nodes is too far or too close to avoid the phenomenon of network nodes being separated or overlapping coverage. It is introduced into the node redeployment process to ensure the node coverage.

5 Experimental study

5.1 Experimental environment

In order to verify the feasibility of the node redeployment method based on the improved leapfrog algorithm in wireless sensor network, the following experimental studies are carried out. The experimental environment is C compilation. Assuming that the current distribution range of the sensor network is a two-dimensional area of 200 m × 200 m. In this network, the initial coverage radius of the node to be deployed is 40 m, and the initial delay of the sensor network is 2.4 ms. The energy size of the wireless sensor SINK node is expressed as E_0 = 200 kJ, and the transmission bit rate is 20 KB/s. The deployment plan is designed based on the above experimental environment. Assuming that the range of nodes to be deployed in the network is controlled within a 5 km × 5 km area, and experimental research is conducted in this area, the number of deployed nodes is 100.

The specific control protocol of the experimental test platform is shown in Table 1.

Table 1 Specific control protocol of experimental test platform

Hexadecimal	*Decimal system*	*Control object*	*State*	*Hexadecimal*	*Decimal system*	*Control object*	*State*
01H	01	LED1	On/off	09H	09	LED1/3/5/7	Full open
02H	02	LED2	On/off	OAH	10	LED1/3/5/7	Full open
03H	03	LED3	On/off	OBH	11	LED2/4/6/8	Full open
04H	04	LED4	On/off	OCH	12	LED1/3/5/7	All pass
05H	05	LED5	On/off	ODH	13	All LED	Full open
06H	06	LED6	On/off	OEH	14	All LED	All pass
07H	07	LED7	On/off	OFH	15	On/off	Open/close
08H	08	LED8	On/off				

5.2 Research on performance indicators

1 Improved algorithm and traditional algorithm static deployment experiment on wireless sensor network nodes. The average deployment error of the two is compared. On average, the results of wireless sensor network node deployment errors are as follows:

$$E = \left(\sum_{i=1}^{n} \left\| x_{estimation}(i) - x_{real}(i) \right\|^2 \right) \Big/ (n \times r) \tag{19}$$

In formula (19), n indicates the number of nodes, and r represents the connection between the current node and the radius.

2 Coverage: it is an important indicator to measure the deployment results of wireless sensor network nodes. The expression is the ratio between the coverage area of the overlapping number of working nodes and the total deployment area. The specific calculation method is as follows:

$$S = \frac{L}{R} \times 100\% \tag{20}$$

In the above formula, L represents the number of sampling points that meet the test conditions; R represents the total number of sampling points.

3 Number of working nodes: indicates the number of active nodes from the end to the beginning of any work step during the node deployment process. The specific expression is as follows:

$$Q = l\left[\frac{1}{2}(m-i)(m-i+1)l + (m-i)x\right] \tag{21}$$

Among them, l represents the density of data transmitted by redundant nodes in the power communication network.

$$l = \frac{n}{L \times L} \tag{22}$$

5.3 Experimental results and analysis

5.3.1 Comparison results of node deployment errors

The actual performance of the proposed algorithm is verified by comparing the node deployment errors of the following different methods, as shown in Table 2.

Table 2 Comparison results of node deployment errors

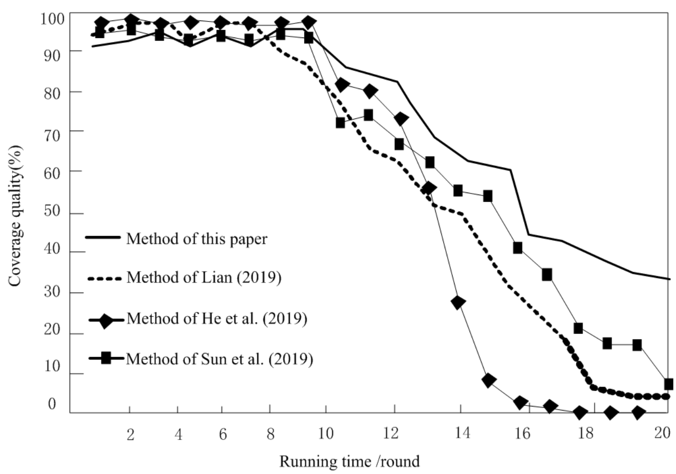
Business load/Erlang	*Node deployment error*			
	Method of this paper	*Method of Lian (2019)*	*Method of He et al. (2019)*	*Method of Sun et al. (2019)*
120	0.301	0.321	0.287	1.021
140	0.272	0.358	0.344	1.142
160	0.257	0.385	0.471	1.207
180	0.224	0.413	0.568	1.352
200	0.188	0.446	0.685	1.438
220	0.159	0.479	0.766	1.584
240	0.144	0.507	0.892	1.705
260	0.121	0.534	0.964	1.852
280	0.100	0.565	1.028	1.967
300	0.087	0.592	1.062	2.108
320	0.071	0.631	1.141	2.249
340	0.064	0.658	1.200	2.361
360	0.040	0.685	1.288	2.498
380	0.002	0.717	1.349	2.582
400	0.021	0.754	1.407	2.660
420	0.001	0.789	1.567	2.702
440	0.002	0.824	1.684	2.842
460	0.021	0.861	1.741	2.963

Analysis of the above experimental data shows that the node deployment error of the proposed algorithm has a more obvious downward trend than the other three methods, as low as 0.01, which fully illustrates the superiority of the proposed method. The main reason is that the proposed method analyses the occupancy of different link spectrum slots based on blockchain technology, optimises the allocation target, updates the frequency slot status of the occupied frequency slot of the link where the path is located, and completes network dynamic routing and spectrum allocation. The spectrum utilisation rate and bandwidth blocking rate are guaranteed, thereby reducing node deployment errors.

5.3.2 Comparative analysis of coverage rates

The comparison of coverage between the method in this paper and the method in literature is shown in Figure 5.

Figure 5 Comparison of coverage of different methods

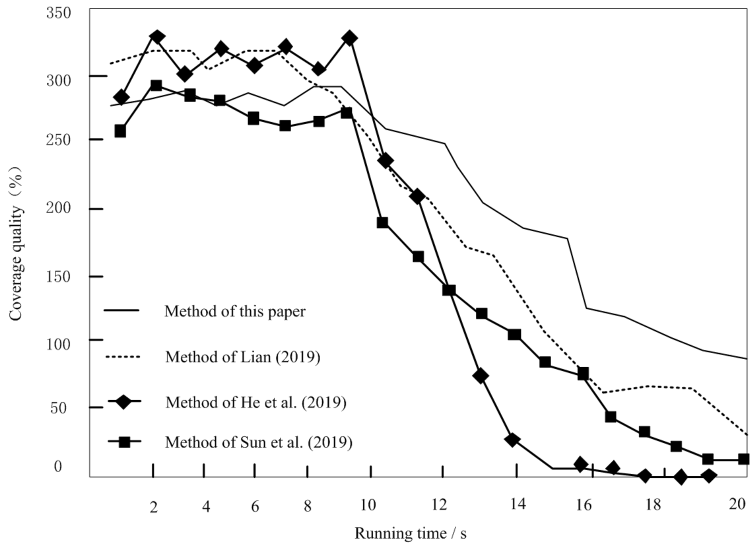


As can be seen from the above figure, the deployment method based on the improved frog leaping algorithm in this paper has a higher node coverage. After the running time reaches 8 minutes, the node coverage of different methods has decreased, but the method in this paper is still at a relatively high degree of coverage. The main reason is that the method in this paper improves the leapfrog algorithm, combined with the interaction of the virtual force model to ensure the connection between the sensor network nodes, effectively avoiding the distance between nodes too far or too close, and avoiding the phenomenon of network nodes being alienated or overlapping. Introducing it into the node deployment process ensures the degree of node coverage.

5.3.3 Comparison and analysis of the number of working nodes

Figure 6 shows the comparison of the number of working nodes.

Figure 6 Comparison results of the number of work nodes in different methods



By analysing the results shown in Figure 6, it can be seen that in the initial stage of operation, under the condition of high-quality coverage, the number of working nodes is also guaranteed. As the running time increases, the number of working nodes in this method is still in a highly active state, which can guarantee the deployment of wireless sensor network nodes. In this paper, the virtual force model is introduced to improve the leapfrog algorithm, which guarantees the coverage of node deployment and the number of working nodes, which proves that the method in this paper has high deployment performance.

6 Conclusions

1. In this paper, based on the improved leapfrog algorithm of wireless sensor network node deployment method, this method is introduced to leapfrog algorithm was improved virtual force algorithm, node deployment area physical model is established, combining with the deployment area of attraction and repulsion, optimise the node deployment, in wireless sensor network system, based on the improved leapfrog algorithm for static, dynamic node deployment.
2. Experimental results show that the method presented in this paper has the lowest redeployment error, and still has relatively good coverage quality performance under the condition of energy depletion of some nodes. While ensuring network coverage quality, the number of working nodes is reduced.
3. In the following research, it will be further optimised from the perspective of improving the timeliness of redeployment of the method in this paper.

Acknowledgements

This work was supported by Major Science and Technology Research Plan Project in Henan Province under Grant No. 142102110028.

References

Alassery, F. (2019) 'Convergence between virtual MIMO and node deployment strategy for high performance multi-hop wireless sensor networks', *IAENG International Journal of Computer Science*, Vol. 46, No. 2, pp.349–357.

Domga, R.K., Stanica, R., Tchuente, M. and Valois, R. (2019) 'Sensor deployment in wireless sensor networks with linear topology using virtual node concept', *Wireless Networks*, Vol. 25, No. 8, pp.4947–4962.

Du, R., Xiao, M. and Fischione, C. (2019) 'Optimal node deployment and energy provision for wirelessly powered sensor networks', *IEEE Journal on Selected Areas in Communications*, Vol. 37, No. 2, pp.407–423.

Guo, Z.L., Peng, J., Yin, B.Y., Wang, S.W., Yue, X.Z. and Liu, X.S. (2019) 'An enhanced gravitational search algorithm for node deployment in wireless sensor networks', *International Journal of Wireless and Mobile Computing*, Vol. 17, No. 1, pp.36–42.

Hajizadeh, N., Javidan, R., Shamsinejad, P. and Akbari, R. (2020) 'Node deployment in wireless sensor networks using the new multi-objective Levy flight bee algorithm', *IET Wireless Sensor Systems*, Vol. 10, No. 2, pp.78–87.

Halakarnimath, B.S. and Sutagundar, A.V. (2020) 'Multi-agent-based acoustic sensor node deployment in underwater acoustic wireless sensor networks', *Journal of Information Technology Research*, Vol. 13, No. 4, pp.136–155.

He, Q., Xu, Q.S. and Wei, K.Y. (2019) 'Deployment optimization of wireless sensor nodes based on improved sine cosine algorithm', *Computer Applications*, Vol. 12, No. 7, pp.2035–2043.

Huan, Y., Feng, L., Yu, D.X., Zou, Y.F. and Yu, J.G. (2020) 'Reliable data storage in heterogeneous wireless sensor networks by jointly optimizing routing and storage node deployment', *Tsinghua Science and Technology*, Vol. 26, No. 2, pp.230–238.

Li, W. and Tu, X. (2020) 'Quality analysis of multi-sensor intrusion detection node deployment in homogeneous wireless sensor networks', *The Journal of Supercomputing*, Vol. 7, No. 6, pp.1331–1341.

Lian, H.M. (2019) 'Topology reconfiguration of wireless communication network based on improved ant colony algorithm', *Journal of Taiyuan Normal University (Natural Science Edition)*, Vol. 18, No. 3, pp.66–71.

Nasri, N., Mnasri, S. and Val, T. (2020) '3D node deployment strategies prediction in wireless sensors network', *International Journal of Electronics*, Vol. 107, No. 5, pp.808–838.

Singh, S. (2019) 'A proficient node deployment mechanism using adjustable sensing range in wireless sensor networks', *Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, Vol. 43, No. 1, pp.191–199.

Sun, Z.Y., Liu, P.L. and Yan, B. (2019) 'Multi node joint network coverage algorithm based on probability perception model', *Computer Engineering and Applications*, Vol. 55, No. 5, pp.36–43.

Wei, Z.L. and Xiu, M.T. (2020) 'Quality analysis of multi-sensor intrusion detection node deployment in homogeneous wireless sensor networks', *Journal of Supercomputing*, Vol. 76, No. 2, pp.1331–1341.

Yang, G.Y., Xi, C.B., Gan, X. and Zheng, T. (2020) 'Energy efficient node deployment optimization for CTS based on cooperative beamforming in single-hop wireless sensor networks', *Automatic Control and Computer Sciences*, Vol. 54, No. 2, pp.147–155.