Global and Stochastic Analysis
Vol. 10 No. 3 (December, 2023)

OPTIMIZATION OF WSN NETWORKS CLUSTARIZATION BASED ON STOCHASTIC ANALYSIS

ANNA MAKAROVA, VLADIMIR GORLOV, ALLA MAKAROVA

Abstract. The main idea of Wireless Sensor Networks (WSN - Wireless Sensor Network) is to fully automate the operation of the network through participation in the collection of human data flow, for example, this may be due to the fact that the sensor is significantly removed from the engineering team or frequent access to the sensor is complicated, as well as if it is necessary to collect data systematically on for an extended period of time. Based on the algorithms obtained and the model for finding the most efficient network partitioning presented in this publication, it is possible to develop protocols and implement an improved SDN network.[1, 2]

Introduction

Clustering in wireless MESH networks It was previously noted that wireless MESH sensor networks have a significant drawback: they have limited resources, in particular, the most important of them is the charge and the ability to replenishment. Also, such networks usually have low or medium speeds. data transmission. All this imposes restrictions on the development of protocols and algorithms implemented in WSN networks, and at the same time opens a vast area for research to increase the sustainability of work, data transfer rates in such networks. [2]

From this we can conclude that all developed and developed algorithms and protocols for sensory wireless networks should show high performance indicators with limited network resources. It is worth noting that when we talk about touch wireless network, we should not forget the fact that a relatively large a period of time in such networks, the nodes are in sleep mode and do not transmit information continuously. Therefore, when developing algorithms and protocols of WSN networks, it is required to apply the principles of self-organization and self-configuration of the network without direct human intervention. The central object of study in the creation of WSN networks is the process of clustering. Formation of a clustered WSN networks in most cases depends on the way the main node is selected, at the same time, one main requirement is imposed on it in the functioning cycle: ensure maximum stability, durability network operation period, as well as network coverage.Note that in a clustered WSN, the way to choose a master host is almost equivalent to a packet routing protocol. For the head node selection method, you

Date: Date of Submission March 11, 2023 ; Date of Acceptance June 7, 2023 , Communicated by Yuri E. Gliklikh.

2010 *Mathematics Subject Classification.* Primary 60H15; Secondary 60H30; 37N35; 49J20.

Key words and phrases. SDN; WSN; efficiency; network; probability, mathematical modelling; quality of the partition.

can define a series requirements and formulate them as follows: - the algorithm needs to be scalable and function efficiently in clusters and networks with a different number of nodes; - the algorithm needs to reduce the number of calculations to form networks thereby increasing autonomy; - the algorithm needs to be simple enough to calculate and there should be no differences in the speed of calculations on all nodes of the net-work; - the algorithm needs to reduce the number of data transfers by extending thus the period of operation of the network nodes. To date, polygamy has been researched and developed protocols and algorithms for choosing the main node of the cluster, but, in practice, they are all designed for WSNs with fixed nodes. It is the study of WSN with non-stationary nodes that is the most promising and advanced with allow significantly improve the performance of such networks, since protocols and algorithms developed on these principles for WSN networks allow to integrate them with mobile cellular networks for improved clustering and fault tolerance. Note that for the efficient functioning of the clustered WSN network requires a sufficiently high density placement of nodes on the area under consideration. In this regard, a parallel task arises to develop a methodology placement of network nodes, optimally locating all network nodes to ensure full coverage of the area under consideration. Thus, it can be unequivocally said that research conducted in the field of developing protocols and routing algorithms, in particular, problems of placing nodes in space and problems selection of the main node in the cluster - are relevant and will be relevant for over an extended period of time. Let's summarize the information about wireless sensor networks (WSN - Wireless Sensor Network). Such networks are distributed, self-configuring, self-organizing, consist of a large number of nodes placed with high density and combined using radio channels.[3]

Advantages of WSN networks: - ability to self-healing and self-organization; - the ability to transmit data over long distances with enough low power transmitters. For this purpose, the method relaying when transmitting packets in the network; - rather low cost of the nodes themselves and their compact dimensions; - low energy consumption and the ability to receive energy from autonomous sources of electricity; - the absence of wires and cables when forming a network, as well as simplicity of practical engineering implementation; - the ability to integrate WSN networks into already functioning networks without additional complex technical work; - low cost of maintenance. There are two main types of wireless sensor networks. varieties: homogeneous (homogeneous) and heterogeneous (heterogeneous) WSN networks.

In homogeneous wireless networks, all nodes have the same energy, i.e. have the same power source, and perform computationally homogeneous operations, i.e. have approximately the same hardware. In turn, heterogeneous wireless sensor networks do not have homogeneity of network nodes and have two or more types of nodes in terms of energy and hardware components. There are three main types of resource heterogeneity in nodes in WSN networks: - computational inhomogeneity. Computational inhomogeneity is the heterogeneity with respect to the computing power of the network node. There are advanced nodes in the network that have an improved process. rubbish, large amounts of RAM and large permanent storage device. Such nodes in the network can act as as the centers of the main calculations of the task, and the usual nodes transmit the collected data to data centers. Thus, it is decided the task of complex information processing and its long-term storage. - heterogeneity of network opportunities. A number of nodes in the network have a more advanced transmitter and receiver, i.e. give higher throughput. Such a heterogeneous node is designed to provide more reliable transmission of information on the network. - heterogeneity of energy. Advanced nodes in such a network have a more powerful power source or possibly connected to stationary power source. Such nodes are most often become the main nodes in WSN networks with energy heterogeneity and designed to provide data aggregation in the network. Application of heterogeneity in the creation of wireless touch networks gives us a number of advantages:

1) Reducing data transmission delay. In particular, if we talk about computational inhomogeneity, then its application allows significantly reduce data transmission delays for nearby nodes. Network heterogeneity also reduces the waiting time packet in the queue for transmission, which also has a positive effect on network efficiency.

2) Increasing the sustainability of the network and network lifetime indicator for one period. Much the average energy consumption required for transmission is reduced messages from normal nodes.

3) Increasing the reliability of data transmission. Network application heterogeneity allows you to build a more reliable WSN network, since advanced nodes have a more advanced transmitter and receiver.

1. The main result

1.1. "Peacock tail" clustering algorithm for wireless sensor MESH networks. To find the most efficient partitioning of the network, consider partitioning $M \in \Phi$ - splitting n vertices into m clusters. Φ is the set of all possible partitions of the set of vertices of the graph V.

Let's define some random variable Q taking values from 1 to m with probabilities q_i, where $i = 1..m$. For all clusters of the i network, we define some random variable P^i, taking values from 1 to n_i with probabilities p_i^k, where $k = 1..n_i$. As a result of the study, we obtain an expanded understanding of the quality indicator of the $L(M) partition$ is the upper bound of the length of the codeword that determines the quality of the partition

$$L(M) = \sum_{i=1}^{m} q_i \ln(\sum_{i=1}^{m} q_i) - 2\sum_{i=1}^{m} q_i \ln(q_i) - \sum_{\alpha=1}^{n} p_\alpha \ln(p_\alpha) + \\ + \sum_{i=1}^{m} (q_i + \sum_{\alpha \in i} p_\alpha) \ln(q_i + \sum_{\alpha \in i} p_\alpha)$$

The general scheme of the clustering algorithm is as follows:

Input: $G(V, E)$ - graph, where V is the set of all vertices, and E is the set of all edges, $n = ||V||$, $m = ||E||$.

Sequence of steps:

1) Initially, each vertex is considered a separate cluster. Counts and the partition quality index $L(M)$ is stored. Transition to step 2.

2) Random displacement results in a sequence peaks. Go to step 3.

3) Next, we take into account the frequency of occurrence of vertices, subsets are formed vertices of the resulting sequence. A cluster is formed from vertex data. Go to step 4.

4) For the resulting partition, the partition quality index is calculated $L(M)$ then the partition M is preserved and go to step 2 (functioning algorithm continues). Otherwise, if the value of the quality score partition $L(M)$ has not decreased, go to step 5.

5) As a result of the algorithm, the partition M is taken as the result of the clustering process in the network. Go to step 6. 6) End of the algorithm.

In addition to the developed scheme, an algorithm for constructing a geometric model based on a graph with distinguished communities, based on the method of physical analogies.

Algorithm 1. Placement of clusters of one connected component based on the geometric model of the method of physical analogies. Input data: $G(V, E)$ - graph, where V - set of all vertices, E - set all edges, $n = ||V||$, $m = ||E||$. G is a grouped set of clusters one connectivity component according to the scheme discussed above.

Sequence of steps:

1) Each allocated cluster is placed using the algorithm .quick peacock tail.. Go to step 2.

2) After placing the clusters separately for each cluster the dimensions (width and height) of the bounding rectangles are calculated. Go to step 3.

3) Next, a new graph is built in which the vertices are grouped clusters (Algorithm 2). Go to step 4.

4) Automatic placement is applied to the new graph. fast peacock tail., resulting in a visual placement grouped clusters, centered relative to the origin coordinates. Go to step 5.

5) With the help of binary search, the coefficient by which the coordinates of the vertices of the cluster graph will be multiplied, since when placing the cluster graph, the sizes of the vertices were not taken into account in any way grouped clusters. The coefficient will be equal to such a minimum vertex placement scale, at which, after ungrouping the framing rectangles of each cluster will not intersect between themselves. Go to step 6.

6) Compress the placement obtained in step 5. Go to step 7.

7) End of the algorithm.

Note that the placement obtained in step 5 is not enough in order to get the final idea. Cons of this accommodation is that some clusters are very far from center, while they did not intersect with other clusters. This is due to the fact that the scale in step 5 is applied to all vertices. Therefore, step 6 produces placement compression. It happens in the following way. All vertices of the community graph are sorted in order increasing distance to the center of coordinates. Then successively for each such vertex, using a binary search, search for a coefficient that shows how many times you can maximize reduce the distance to the center, while guaranteeing the absence of intersections between clusters.

In fact, this procedure at each step fixes all vertices, except for one, and tries to move it as far as possible to the beginning coordinates in such a way that after ungrouping the clusters will not cross.

1.2. Node Clustering Algorithm for Wireless sensor MESH networks. As part of the solution of the task, the subtask on development of an algorithm for a general scheme for selecting clusters.

Consider the set Φ of all possible partitions of the set vertices of the graph V. A partition $M \in \Phi$ is a partition of n vertices into m clusters. For the partition M, we define some numerical value $L(M)$ - the upper bound on the length of the code word that determines the quality of the partition M. Now let $L(M)$ be the quality index of the partition M. In order to find the index $L(M)$ for a certain partition M apply Shannon's theorem. It is formulated as follows: when n is used code words to describe n states of a random variable X, which occur with frequencies p_i, the average length of a codeword cannot be less than the entropy of a random variable X, represented as:

$$H(X) = -\sum_{i=1}^{n} p_i \log(p_i)$$

The partition quality index $L(M)$ will be determined based on last formula. Let us define for the network a fixed partition M into some clusters, we also define some random variable Q taking values from 1 to m with probabilities q_i, where $i = 1..m$. For all clusters i of the network, we define some random variable P^i that takes values from 1 to n_i with probabilities p_i^k, where $k = 1..n_i$. The calculation of the partition quality index $L(M)$ depends on the entropy random variables Q and P^i defined above as follows:

$$L(M) = qH(Q) + \sum_{i=1}^{m} p_i H(P^i)$$

were $q = \sum_{i=1}^{m} q_i$ probability of transition between clusters on each step of the random movement, q_i - the probability of leaving the cluster i, p_α - probability of hitting vertex α, $p_i = \sum_{\alpha \in i} p_\alpha + q_i$ - probability stay in the cluster i.

$$H(Q) = -\sum_{i=1}^{m} \frac{q_i}{\sum j = 1^m q_j} \log(\frac{q_i}{\sum j = 1^m q_j})$$

$H(Q)$ - entropy of transitions between clusters.

Let us write the entropy $H(p^i)$ of displacement inside the i-th cluster, defining a lower bound on the average length of a codeword for naming of vertices in i-th cluster:

$$H(P^i) = \frac{q_i}{q_i \sum_{\beta \in i} p_\beta} \log(\frac{q_i}{q_i + \sum_{\beta \in i} p_\beta}) - \sum_{\alpha \in i} \frac{p_\alpha}{q_i + \sum_{\beta \in i} p_\beta} \log(\frac{p_\alpha}{q_i + \sum_{\beta \in i} p_\beta})$$

From the last two formulas we obtain an extended understanding partition quality index $L(M)$:

$$L(M) = \sum_{i=1}^{m} q_i \log(\sum_{i=1}^{m} q_i) - 2\sum_{i=1}^{m} q_i \log(q_i) - \sum_{\alpha=1}^{n} p_\alpha \log(p_\alpha)$$
$$+ \sum_{i=1}^{m}(q_i + \sum_{\alpha\in i} p_\alpha) \log(q_i + \sum_{\alpha\in i} p_\alpha).$$

It is worth noting here that $\sum_{\alpha\in i} p_\alpha \log p_\alpha$ does not depend on partition networks into clusters. In this connection, during the operation of the algorithm in order to find the most efficient network partitioning, it is required to save all received changes: q_i - the probability of random movement of the input and leaving clusters, and $\sum_{\alpha=1}^{m} p_\alpha$ - time spent in each cluster with random movement. The last formula is calculated quickly by storing more parts of the results of intermediate calculations included in it terms.

Algorithm 2. General scheme of the cluster selection algorithm.

Input data: $G(V, E)$ - graph, where V - set of vertices, E - set edges, $n = ||V||$, $m = ||E||$.

Sequence of steps:

1) At the beginning of the algorithm, each vertex is separate cluster. The partition quality index is calculated and recorded $L(M)$. Go to step 2.

2) Random displacement forms a sequence of vertices. Go to step 3.

3) Next, we take into account the frequency of occurrence of vertices, subsets are formed vertices of the resulting sequence. A cluster is formed from vertex data. Go to step 4.

4) For the resulting partition, the partition quality index is calculated $L(M)$. If this value becomes smaller as a result, then the partition M is preserved and go to step 2 (functioning algorithm continues). Otherwise, if the value of the quality score partition $L(M)$ has not decreased, go to step 5.

5) As a result of the algorithm, the partition M is taken as the result of the clustering process in the network. Go to step 6.

6) End of the algorithm.

In further work, when using the algorithm and formula for calculation $L(M)$, the probability of network failures decreases, the efficiency indicator increases and the "lifetime" of the network increases as a whole.

References

[1] A.V. Makarova, V.A. Gorlov, " Stochastic analysis methods in SDN networks modelling", *Communications on Stochastic Analysiss*, Vol. 14 No. 1-2 (2020), 13-18.

[2] A.V.Samsonov, S.V.Roslyakov, A.Yu.Vanyashin, M.Yu.Grebeshkov: *Internet of Things*, PGUTI, Samara, 2014.

[3] G. Kiran M. Kamal K. Nitin. Wireless Sensor Network: A Review on Data Aggregation *International Journal of Scientific and Engineering Research*,Vol. 2 (2011), 1-6.

Anna Makarova: Military Medical Academy of S.M. Kirov, Saint Petersburg, Russian Federation

E-mail address: annaghm@mail.ru

Vladimir Gorlov : SBERBANK PJSC, Voronezh, Russian Federation
E-mail address: gorlov.v.a@gmail.com

Alla Makarova: N.E. Zhukovsky and Y.A. Gagarin Air Force Academy, Voronezh, Russian Federation
E-mail address: allagm@mail.ru