Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 410363, 7 pages
http://dx.doi.org/10.1155/2014/410363

Research Article

Optimal Placement of Actors in WSANs Based on Imposed Delay Constraints

Chunxi Yang,[1] Xinyu Wu,[1] Jian Huang,[2] and Lingyun Huang[3]

[1] *Faculty of Chemical Engineering, Kunming University of Science and Technology, Kunming 650500, China*
[2] *Department of Control Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China*
[3] *Faculty of Land Resource Engineering, Kunming University of Science and Technology, Kunming 650500, China*

Correspondence should be addressed to Lingyun Huang; hly0510@126.com

Received 20 November 2013; Accepted 2 February 2014; Published 23 March 2014

Academic Editor: Huaicheng Yan

Wireless Sensor and Actor Networks (WSANs) refer to a group of sensors and actors linked by wireless medium to probe environment and perform specific actions. Such certain actions should always be taken before a deadline when an event of interest is detected. In order to provide such services, the whole monitor area is divided into several virtual areas and nodes in the same area form a cluster. Clustering of the WSANs is often pursued to give that each actor acts as a cluster-head. The number of actors is related to the size and the deployment of WSANs cluster. In this paper, we find a method to determine the accurate number of actors which enables them to receive data and take actions in an imposed time-delay. The k-MinTE and the k-MaxTE clustering algorithm are proposed to form the minimum and maximum size of cluster, respectively. In those clustering algorithms, actors are deployed in such a way that sensors could route data to actors within k hops. Then, clusters are arranged by the regular hexagon. At last, we evaluate the placement of actors and results show that our approach is effective.

1. Introduction

Networked control systems (NCSs) have attracted a lot of interest for its low cost, reduced system wiring, simple system diagnosis, and maintenance [1–3]. As special NCSs, Wireless Sensor and Actor Networks (WSANs) [4] have started to attract growing interest from the research and engineering communities in recent years. WSANs can increase the effectiveness of numerous applications such as home automation, infrastructure health monitoring, intelligent buildings, and sewer management [5, 6]. Such networks employ a large quantity of miniaturized sensor nodes and a few number of actor nodes [7]. Sensor nodes are small and inexpensive, usually with limited power and limited data processing capabilities, while actor nodes are more capable nodes with relatively more onboard energy supply and richer computation and communication resources. Sensors probe their surroundings and report their detective parameters to the actor nodes, which process the collected sensor's reports and respond to emerging events of interest, shown in Figure 1.

Real-time requirement is one of the most important design goals in time critical applications. Certain actions should be taken quickly enough after sensors detect an event. However, a minimal transmission delay may not guarantee a valid data that meets certain applications' requirements. Some applications may impose a time deadline on sensed data to be delivered to an actor for timely acting [8]. For example in forest monitoring applications, actors such as robots need to be engaged in a short interval to control a fire accident and prevent it from spreading.

In order to fulfill the real-time requirement, the whole monitor area is divided into several virtual areas and nodes in the same area form a cluster [9]. Clustering of the WSANs is often pursued to give that each actor act as a cluster-head and takes certain actions based on the received information from the sensors within its cluster [3, 10]. The larger areas a cluster of WSANs covers, the more hops that sensed data need to pass through correspondingly. In the case of that, the end-to-end delay and action time would be extended.

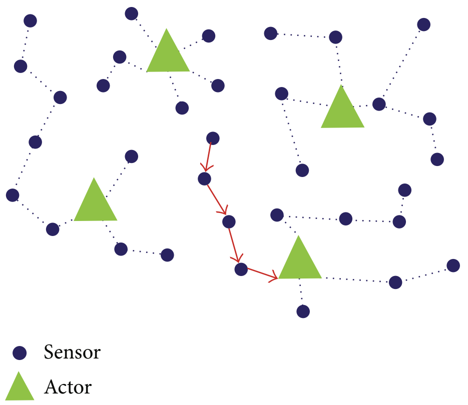

Figure 1: A wireless sensor and actor network.

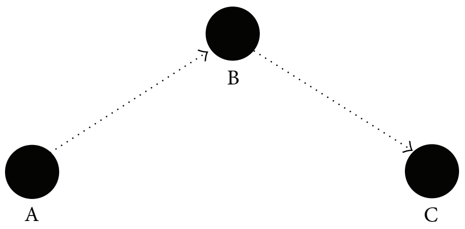

Figure 2: MTE routing path.

In this paper, we find a method to determine the accurate number of actors needed to cover the monitoring area where they receive data and take actions in an imposed time-delay. The number of actors is related to the size and the deployment of WSANs clusters. Different clustering algorithms lead to the different clusters' size. Based on the classical MTE routing protocol [11], we propose a k-MinTE clustering algorithm to minimize the size of cluster, while a k-MaxTE clustering algorithm is presented to maximize the size of clusters in WSANs. By statistical analysis of simulation experiments, we determine both maximum and minimum size of WSANs clustered by both clustering algorithms. According to the size of WSANs clusters, the minimal number of actors needed in both algorithms can be determined.

This paper is organized as follows. Section 2 summarizes the related work and distinguishes our work from the previous studies. In Section 3 we provide system modeling and problem description. Section 4 presents the details of k-MinTE and k-MaxTE clustering algorithm. The simulation of the approach is in Section 5. Section 6 concludes the paper with a summary.

2. Related Work

2.1. Maximum Allowable Response Time. In [12], it revealed the relationship between the maximum acceptable response time for actors and the number of hops that collected data passed through. The maximum acceptable response time $R_{\max}$ can be described by

$$R_{\max} = \sum_{i=1}^{N_h} T_{i,i+1} + T_h \times (N_h + 1) + T_a, \qquad (1)$$

where $T_{i,i+1}$ is the time cost in transmitting or receiving data by any corresponding couple of hops i and $i+1$. T_h is the time spent in each node for data process and congestion, including actor node. N_h is the number of hops from the source sensor to the destination actor. T_a is the time to initiate the first action by the destination actor. It is assumed that each cluster's coverage is equal in our context; we regard T_a as the maximum acting time from center to the edge.

In the application of WSANs, it is usually given an imposed delay which could be regarded as the maximum acceptable response time. Thus, according to (1), the maximum hops between sensor and actor in a cluster could be determined. Therefore, information delivered to an actor could be guaranteed to meet the delay constraints.

2.2. Regular Hexagonal Node Coverage Model. Recently, there has been a lot of works to improve the coverage of WSANs through intelligent actor deployment. The main idea in [10] is to apply repelling forces among neighboring actors, similar to molecular particles in Physics. Their actor deployment algorithm is based on triangular grid, which means that uncovered areas still exist.

The work in [13] presented a Distributed Actor Deployment Algorithm for Maximum Coverage (DA^{2}MC), which divides the monitored area into several virtual regular hexagon areas and nodes in each regular hexagon area form a cluster. Actors are also selected as cluster-heads. They proved that regular hexagonal node coverage model can reach maximum coverage without uncovered areas. Because the overlaps between clusters are minimum, DA^{2}MC covers the area with the least actors deployed. However, they regarded actor's radio range as the radius of circle. Actors may not be guaranteed to take actions in an imposed deadline, since the action range (cluster's size) is always smaller than radio range.

Considering these situations, we focus on determining the cluster's size in WSANs which could guarantee the end-to-end delay within an acceptable span (Figure 11).

2.3. MTE Routing Protocol. The main idea of minimum transmission energy (MTE) protocol is to minimize the transmit amplifier energy. Routes from each sensor to the actors are chosen such that each sensor's next-hop neighbor is the closest node that is in the direction of the actors [11]. In this case, node A would transmit a message through node B to node C, shown in Figure 2.

Since each sensor chooses the closest node to route data, the size of cluster formed by MTE protocol is minimum. However, since the MTE protocol could not guarantee to meet delay requirement, we present a k-MinTE clustering algorithm which minimizes the size of WSANs cluster and enables data route to actors within k hops to meet the delay constraints at the same time. Corresponding to the k-MinTE

```
k-MinTE clustering algorithm
// for each actor $I$ perform the following
Broadcast message (“cluster-head $I$”, “location $I(x, y)$”) within its cluster
// for each sensor $j$ in the cluster $I$ perform the following
Receive message from cluster-head $I$
Calculate the distance to the cluster-head $D\text{-}I\text{-}j$
Broadcast message (“sensor $j$”, “distance $D\text{-}I\text{-}j$”, “hops $H\text{-}j$”) within the cluster $I$
Receive message from neighbors
Generate a maximum acceptable hop $k$
    while $D\text{-}I\text{-}j <$ sensor’s radio range $R$
        then connect to the actor $I$ directly
            hops $H\text{-}j = H\text{-}j + 1$
    while $D\text{-}I\text{-}j \geq$ sensor’s radio range $R$
        then find a minimum distance $D\text{-}j\text{-}z$ to sensor $z$
            if $D\text{-}j\text{-}z < R$ and $H\text{-}z > 0$ and $H\text{-}z < k$
                then connect to the sensor $z$
```

Algorithm 1: The k-MinTE clustering algorithm pseudocode.

```
k-MaxTE clustering algorithm
// for each actor $I$ perform the following
Broadcast message (“cluster-head $I$”, “location $I(x, y)$”) within its cluster
// for each sensor $j$ in the cluster $I$ perform the following
Receive message from cluster-head $I$
Calculate the distance to the cluster-head $D\text{-}I\text{-}j$
Broadcast message (“sensor $j$”, “distance $D\text{-}I\text{-}j$”, “hops $H\text{-}j$”) within the cluster $I$
Receive message from neighbors
Generate a maximum acceptable hop $k$
    while $D\text{-}I\text{-}j <$ sensor’s radio range $R$
        then connect to the actor $I$ directly
            hops $H\text{-}j = H\text{-}j + 1$
    while $D\text{-}I\text{-}j \geq$ sensor’s radio range $R$
        then find a maximum distance $D\text{-}j\text{-}z$ to sensor $z$
            if $D\text{-}j\text{-}z < R$ and $H\text{-}z > 0$ and $H\text{-}z < k$
                then connect to the sensor $z$
```

Algorithm 2: The k-MaxTE clustering algorithm pseudocode.

algorithm, a k-MaxTE clustering algorithm is presented to maximize the size of WSANs clusters.

3. System Modeling and Problem Description

3.1. Network Initialization. We assume that a set of sensors are spread randomly throughout a field of interest. All the sensor nodes are homogeneous and stationary, which is typical for WSANs.

The actors will be placed in the same area to collect information from sensors and take necessary actions based on received information. It is assumed that actors cannot sense information and can only take actions based on sensors' information. We also divide the monitored area into several virtual regular hexagon areas and nodes in each regular hexagon area form a cluster. The range of a cluster is a limited circle circumscribed about the regular hexagon and assumed to be equal. Finally we assume that both the sensors and actors know their locations through mechanisms like GPS or other means.

3.2. Problem Description. We define the problem as follows. “Given a set of abundant sensors initially deployed randomly in an area of interest and an unknown number of actors, we are interested in determining the minimum number of actor nodes and the maximum cluster size of WSANs such that transformed time delay could be limited in an imposed period and actors can cover the whole area without uncovered areas.”

4. K-MinTE and k-MaxTE Clustering Algorithm

4.1. Description of k-MinTE Clustering Algorithm. We also divide the monitored area into several virtual regular hexagon areas and actors are placed in the center of each regular hexagon as cluster-heads.

Based on the MTE routing protocol, we present a k-MinTE clustering algorithm. Here, the parameter k shows the number of hops for a sensor to reach its dominator and thus its cluster-head. If k is infinity, the k-MinTE becomes classic MTE algorithm. While k is a certain positive integer, the cluster will be formed such that a sensor reaches its actor within k hops routes. The main idea is that each sensor connects its closest next-hop neighbor, while the route from sensors to the actor is guaranteed within k hops. The completing k-MinTE clustering algorithm pseudocode is given in Algorithm 1.

As shown in line 1, each actor in the area broadcast a CLUSTER-HEAD and LOCATION message within its action range. Then sensors wait to receive the message from cluster-head within its cluster at line 2. In lines 3–5, sensors would calculate the distance away from cluster-head and transmit the message to neighbors. Based on the information of sensors, the nodes begin to connect with each other.

Lines 6–16 are the procedure for a sensor to choose a route. A sensor would choose its closest neighbor and route its data to the actor within k hops. If the route is more than k hops, the sensor would choose another neighbor until fulfilling the requirement.

4.2. Description of k-MaxTE Clustering Algorithm. Differed from the k-MinTE algorithm, a k-MaxTE clustering algorithm is presented to maximize the size of cluster. Here, the parameter k is also the number of hops for a sensor to reach its cluster-head. The main idea is that each sensor connects its next-hop neighbor within its communication span which has the shortest path to actor node, while the route from sensors to the actor is guaranteed within k hops. The complete k-MaxTE clustering algorithm pseudocode is given in Algorithm 2.

Remark 1. In k-MaxTE algorithm, if we adopt that each sensor connects with its farthest next-hop neighbor, the phenomenon that some nodes connect with its farthest next-hop node instead of connecting to actor directly happened. As a result, the given hop k with largest communication radius cannot be found.

Remark 2. k-MinTE algorithm and k-MaxTE algorithm have the minimum radius and the maximum radius of cluster unit, respectively, in all clustering algorithms. Consequently, these two radiuses can be looked as boundary conditions in clustering algorithms design.

5. Experimental Evaluation

5.1. Determination of the Cluster's Size. Since the sizes of all the clusters are equal, we only need to determine one cluster's size so that it could be applied to other clusters. We would determine a cluster's size by the following steps.

(1) Deploy a number of sensors in a big enough circle area and put an actor which acts as cluster-head in the center of the circle. Then, cluster nodes naturally by running k-MinTE and k-MaxTE algorithms, respectively, in which k is set to be large enough. This step is to form a big cluster with enough hops.

(2) Record the distribution of each hop's range with radius in the cluster.

(3) Set k to a certain positive integer and run both clustering algorithms in a circle with the radius of k hops nodes. Sensors would be isolated if they cannot reach actors within k hops. This step is to form a k-hop cluster.

(4) Deduce the cluster's range gradually and search a radius in which sensors of certain hop could be sufficiently included.

(5) Simulate the deployment of WSANs cluster with the radius founded and thus determine the number of actors.

Table 1: The relative parameters.

Maximum acceptable response time $R_{\max}$	10 s
Action time T_a	5 s
Receiving and transmitting time T_h	1 s
Computation time $T_{i,i+1}$	1 s

5.2. Simulation Results. According to (1) and the assumed values listed in Table 1, we have

$$10 = \sum_{i=1}^{N_h} 1 + 1 \times \left(N_h + 1\right) + 5. \qquad (2)$$

The number of hops is $N_h = 2$. We take the clusters run by 2-MinTE and 2-MaxTE algorithm as an example to determine its size. Both clustering algorithms are realized under a visualized emulator wrote by MATLAB.

At first, 500 sensors nodes are placed randomly in a big circle area where radius is 50 meters and an actor is placed in the center. It is set that the communication range of sensors is 10 m. The nodes form a cluster naturally by running the k-MinTE and k-MaxTE clustering algorithm, respectively, in which k is set to be large enough in this case, as shown in Figures 3 and 4.

Remark 3. It can be seen from Figures 3 and 4 that nodes choose the nearest neighbor to cluster in k-MinTE algorithm, while nodes link next-hop neighbor which is as far as possible in k-MaxTE algorithm. Therefore, the number of hops from the same source sensor to cluster-head using k-MaxTE algorithm is much less than that of using k-MinTE algorithm.

Remark 4. If we define a side where a sensor is connected with its neighbor as degree 1, then the number of sensors with bigger degree using k-MaxTE algorithm is much more than those of k-MinTE algorithm. As a result, the life-time of WSANs created by k-MaxTE algorithm becomes much shorter than that of using k-MinTE algorithm.

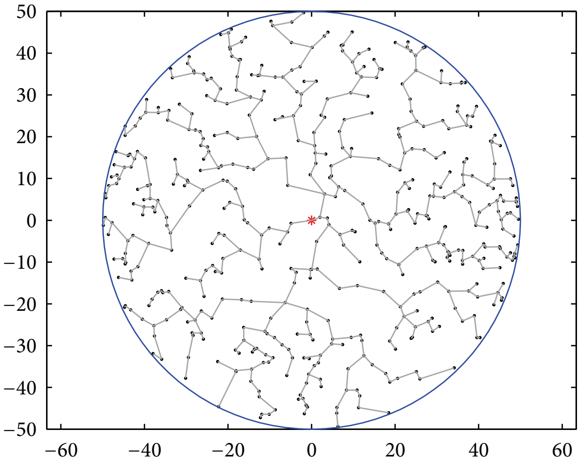

Figure 3: A cluster formed by running the k-MinTE algorithm.

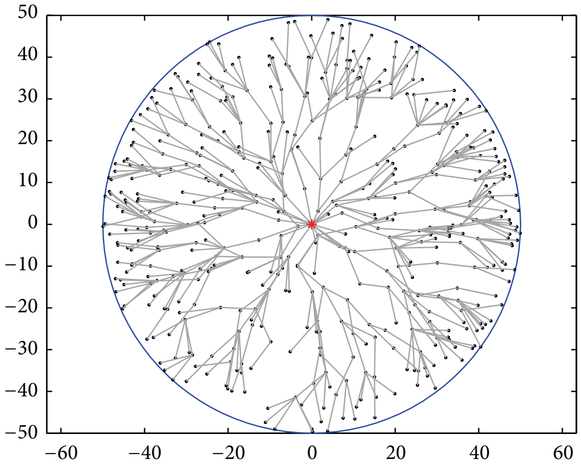

Figure 4: A cluster formed by running the k-MaxTE algorithm.

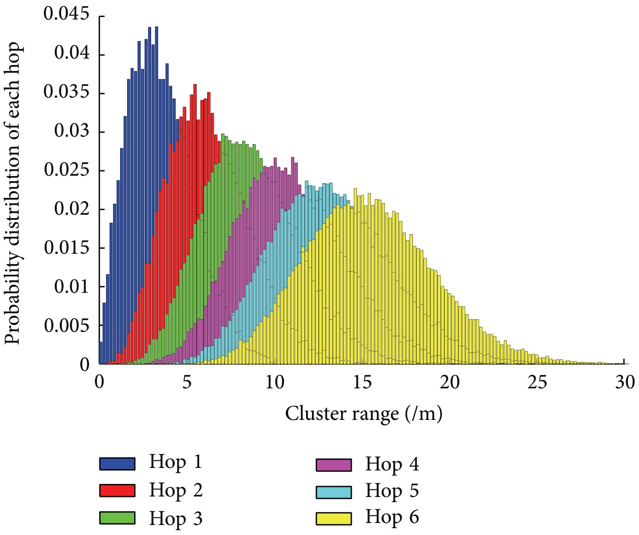

Figure 5: The hop's range distribution in k-MinTE algorithm.

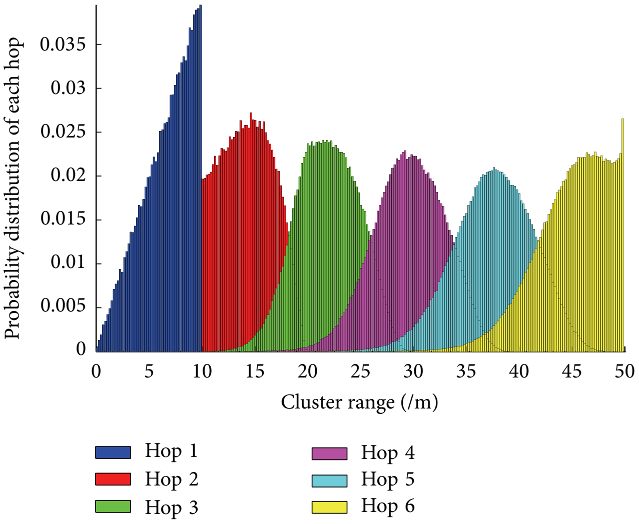

Figure 6: The hop's range distribution in k-MaxTE algorithm.

We record each hop's range distribution in the cluster with radius in Figures 5 and 6. Different color stands for different number of hops. From the bar chart Figure 5, it could be seen that the radius of hop 2 range approximately from 0.4 m to 15 m in k-MinTE algorithm. And in Figure 6, it shows that the radius of hop 2 in k-MaxTE algorithm ranges from about 10 m to 20 m.

The range of cluster is searched that 2-hop sensors could be sufficiently included. Figure 7 shows the distribution of the sensors with 2-hop in k-MinTE and k-MaxTE clustering algorithm. It can be seen that, with radius increased in both algorithms, the number of 2-hop sensors is decreasing. 90% of sensors in 2-hop clustered by k-MinTE are deployed within an approximate range of 4 m, while less than 10% deploy within 10 m. And in k-MaxTE clustering algorithm, the percentage of sensors is decreasing from nearly 90% in 15 m to 0% in 20 m. We would have searched the accurate clusters' radius in the dramatically changed ranges that the 2-hop sensors could be sufficiently included.

The radius is reduced gradually to determine this accurate range by running 2-MinTE and 2-MaxTE algorithm. Some sensors would be isolated if they cannot reach actors within 2 hops. We observe that nearly 97% of 2-MinTE clustered sensors clustered in the radius of nearly 5 m and 99% of sensors were included in 4 m as seen in Figure 8. In Figure 9, it can be seen that about 97% of 2-MaxTE clustered sensors clustered in the radius of nearly 16 m and more than 95% of sensors deployed in 16.5 m.

Finally, we simulate the placement of WSANs with the clusters' radius where 97% of sensors can be included in. In the experiment of 2-MinTE clustering algorithm, the area of

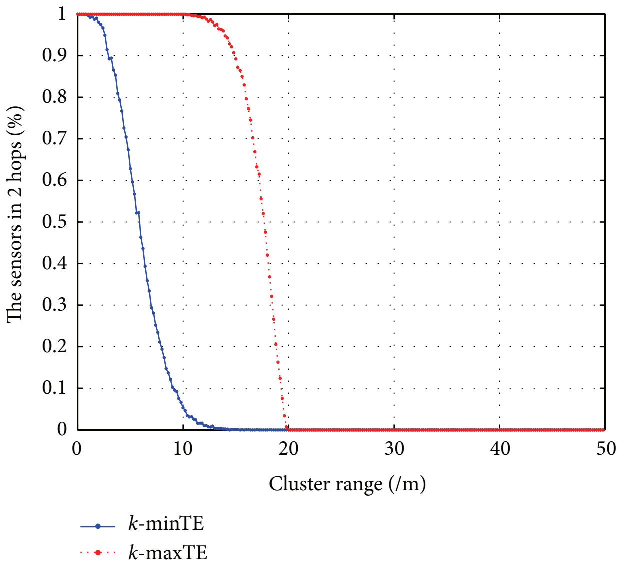

FIGURE 7: The distribution of sensors in 2 Hops.

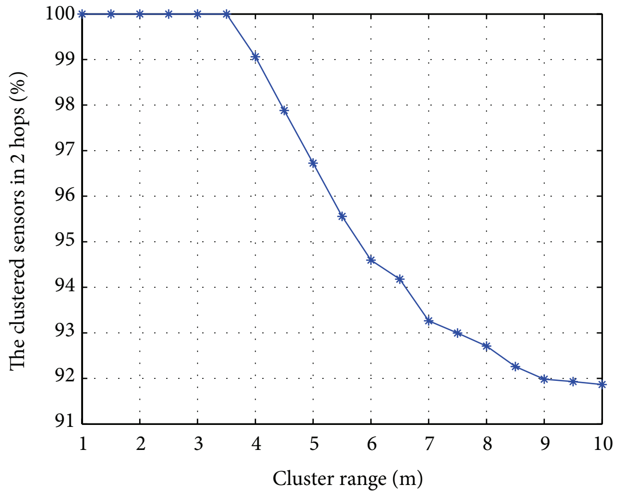

FIGURE 8: The distribution of sensors in 2 hops by 2-MinTE.

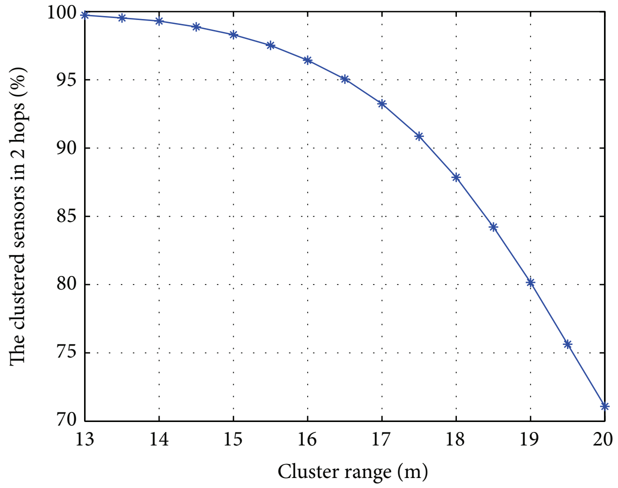

FIGURE 9: The distribution of sensors in 2 hops by 2-MaxTE.

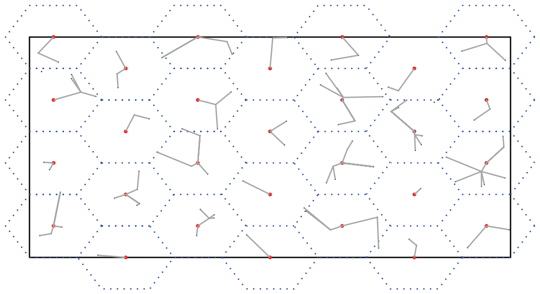

FIGURE 10: WSANs clustered by 2-MinTE.

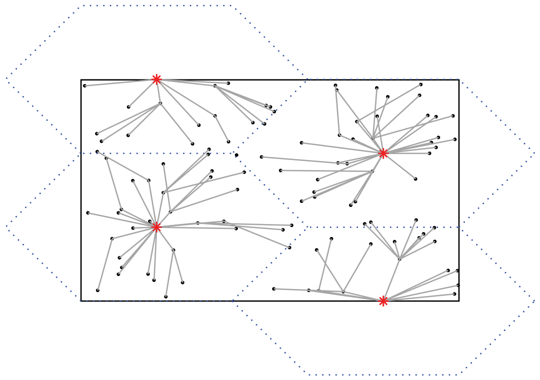

FIGURE 11: WSANs clustered by 2-MaxTE.

a rectangle is 50 × 30 m^2, and 95 nodes of the same density as before are randomly deployed. Since the radius range of the WSANs cluster determined above is 5 m, the whole area is divided into 28 regular six hexagons correspondingly and thus 28 actors are deployed shown in Figure 10. The result shows that above 97.8% of sensors are clustered, which means more than 97.8% of sensors could reach actors in 2 hops. In the simulation of 2-MaxTE clustering algorithm, the area is 40 × 41.5 m^2, and 105 nodes of the same density as before are randomly deployed. The cluster's radius is 16 m determined above and thus the whole area is divided into 4 six hexagons. The simulation result shows that about 99.05% of sensors are clustered.

Tables 2 and 3 show the relationship between the minimal number of actors needed and the density of sensors clustered by 2-MinTE and 2-MaxTE clustering algorithms. Table 2 illustrates the minimal number of actors needed to cover the 50 × 30 m^2 area in different density of sensors clustered by 2-MinTE algorithm. It can be clearly seen that more actors are needed when the density of sensors is increasing, while Table 3 presents that less actors are needed in 2-MaxTE algorithm when the density of sensors is increasing in the area of 135 × 110 m^2.

6. Conclusion

WSANs are gaining more and more interests in numerous fields. Actors need to collect sensor's data and take certain actions in an imposed deadline to perform tasks. According

Table 2: The minimal number of actors in different density in 2-MinTE.

The density of sensors in 2-MinTE algorithm (/m^2)	The minimal number of actors
0.05	21
0.10	45
0.15	55
0.20	78
0.25	91
0.30	97
0.35	105
0.40	136
0.45	161
0.50	180

Table 3: The minimal number of actors in different density in 2-MaxTE.

The density of sensors in 2-MaxTE algorithm (/m^2)	The minimal number of actors
0.025	35
0.050	30
0.075	27
0.100	27
0.125	24
0.150	24

to the relationship of maximum imposed response time and the number of hops between the source sensor and the destination actor, we turn to search the maximum size of WSANs cluster with certain hops. Based on the size of WSANs cluster determined by k-MaxTE and k-MinTE algorithm, the minimum number of actors could be determined. At last, the simulation results prove that our approach is effective.

Conflict of Interests

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

Acknowledgments

This work is supported by National Natural Science Foundation (NNSF) of China under Grants 61004033 and 61364002 and 2012 Jin Chuan school-enterprise cooperation.

References

[1] W. Zhang, M. S. Branicky, and S. M. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–97, 2001.

[2] L. X. Zhang, H. J. Gao, and O. Kaynak, "Network-induced constraints in networked control systems—a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, 2013.

[3] H. Zhang, H. C. Yan, F. W. Yang, and Q. J. Chen, "Quantized control design for impulsive fuzzy networked systems," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 6, pp. 1153–1162, 2011.

[4] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Ad Hoc Networks*, vol. 2, no. 4, pp. 351–367, 2004.

[5] H. Salarian, K. Chin, and F. Naghdy, "Coordination in wireless sensor-actuator networks: a survey," *Journal of Parallel and Distributed Computing*, vol. 72, pp. 856–867, 2012.

[6] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[7] C. Yang, X. Wu, C. Li, and J. Huang, "Dynamic characters of a distributed fusion algorithm in wireless sensor and actor networks," in *Proceedings of the 31st Chinese Control Conference*, pp. 26–28, Hefei, China, 2012.

[8] K. Akkaya, F. Senel, and B. McLaughlan, "Clustering of wireless sensor and actor networks based on sensor distribution and connectivity," *Journal of Parallel and Distributed Computing*, vol. 69, no. 6, pp. 573–587, 2009.

[9] X. Ren and Z. Cai, "A distributed actor deployment algorithm for maximum connected coverage in WSAN," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 283–287, Tianjin, China, 2009.

[10] K. Akkaya and S. Janapala, "Maximizing connected coverage via controlled actor relocation in wireless sensor and actor networks," *Computer Networks*, vol. 52, no. 14, pp. 2779–2796, 2008.

[11] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Siences (HICSS '33)*, Honolulu, Hawaii, USA, 2000.

[12] M. H. Alaiwy, F. H. Alaiwy, and S. Habib, "Optimization of actors placement within wireless sensor-actor networks," in *Proceedings of the 12th IEEE International Symposium on Computers and Communications (ISCC '07)*, pp. 179–184, Aveiro, Portugal, 2007.

[13] K. Akkaya and M. Younis, "Coverage and delay aware actor placement mechanisms for wireless sensor and actor networks," *International Journal of Sensor Networks*, vol. 3, pp. 152–164, 2008.