# Multidimensional Scaling-Based Localization Algorithm for Wireless Sensor Network with Geometric Correction

Yingqiang Ding, Dongsheng Yang, and Gangtao Han
College of Information Engineering, Zhengzhou University, Zhengzhou 450001, PR China
Email: cnpowerfoot@gmail.com

***Abstract*—Position is an extremely important parameter for each node in most of applications using wireless sensor work (WSN). In this paper, a new distributed localization (DMDS-GC) algorithm is proposed based on the multidimensional scaling (MDS) technique and the geometric correction technique. In DMDS-GC, the entire network is split into several localization clusters (LCs) with sufficient common nodes between neighbor clusters by a heuristic search algorithm. Then, based on the geometric relationships between nodes, the shortest path distance matrix is corrected in each LC. During the computation and optimization of relative local maps for LCs, the SMACOF algorithm, which is an improved version of MDS, is implemented. Finally, with the help of common nodes, the global map of entire network is achieved through coordinate transformation and mergence of LCs. Simulation results show that the DMDS-GC algorithm has better performances on localization precision, communication load, energy efficiency and robustness to range error, which can meet the needs of nodes running in WSN.**

***Index Terms*—Wireless Sensor Network; Multidimensional Scaling; Localization; Mergence; Geometric Correction**

## I. INTRODUCTION

Wireless sensor network (WSN) is defined as an ad hoc network consisting of a large number of dispersed sensor nodes, embedded computation, distributed information processing, and radio-link. In recent years, sensor network begins to show its potential in many application fields such as transportation, military, biomedicine, emergency, *etc* [1, 2]. In constructing most of these applications, functionality for the nodes to determine their positions is of extreme importance. From the emergence of Active Badge developed by AT&T Laboratories Cambridge on, plenty of work is devoted to finding the appropriate strategy for localization. Thus, how to determine the sensor's location becomes a topic of active research in wireless sensor network [3-7].

With large number of sensor nodes, the localization problem is to extract the location, which is represented by relative or absolute coordinates, from the complex relationships of networks. In WSN, these relationships can be represented by a weighted and undirected graph, where edge means the radio-link and weight means the distance between two nodes [8-10]. Usually, the relationship set of a node is the high-dimensional data and at least consists of the distances between itself and its neighbors. Therefore, some dimension reduction and optimization method should be implemented for position extraction. Multidimensional scaling (MDS) is a general method that is used to make the high-dimensional data metrical visualization in metric geometry [11]. Through MDS technique, the positions of sensors can be obtained from the complex relationships of networks. Shang firstly advocated the MDS-based localization algorithm [12]. Subsequently, more studies were done by Biaz, Cheung, Ji, etc [13-15]. In these studies, the MDS-MAP [12], MDS-MAP(P) and MDS-MAP(P,R)[16] are the most famous algorithms. These algorithms can make full use of all distance estimations between sensors to simultaneously compute the positions of multi-sensors. In these popular MDS-based algorithms, several feasible frameworks were established, but lots of problems remained. Firstly, all the shortest path distances between nodes are not available due to deployment, which can result in the failure of MDS implementation. Secondly, the shortest path distance is very different from the actual distance and can deteriorate the performance of algorithm badly. Thirdly, because MDS is a centralized algorithm, the implementation to entire network may lead to much energy consumption and memory requirements [17-19].

In this paper, a new distributed localization algorithm, based on MDS algorithm and geometric correction, is presented and named as DMDS-GC. The algorithm constructs the localization clusters (LCs) to split network into local maps, where the shortest path distance is corrected through geometric technique. With the aids of common nodes in neighbor LCs, all local maps are merged to obtain the global map of network, where each node is already localized.

## II. NETWORK MODEL

Here, it is supposed that $N$ nodes are randomly deployed in the $m$ dimensional (m-D) space and the number of anchors is $M$. The positions of all nodes are modeled as $X=(x_1, x_2, \cdots, x_N)$ , where $x_i = (x_{i,1}, x_{i,2}, \cdots, x_{i,m})^T$ is the position of node $i$. The distance between sensor $i$ and sensor $j$ is defined as


doi:10.4304/jnw.9.3.582-587

$$d_{i,j}=\sqrt{\sum\nolimits_{k=1}^{m}(x_{i,k}-x_{j,k})^2}=\|x_i-x_j\|_F$$

where $\|\cdot\|_F$ denotes the Euclidean distance. Additionally, the network is assumed to have the following features:

(1) Each sensor has a unique identifier (ID).

(2) The anchors are position aware.

(3) The signal channel is symmetrical and the maximum communication radius of sensor is $R$. If $d_{i,j}\le R$, node $i$ and node $j$ are neighbors and $d_{i,j}$ can be measured directly

## III. THE DMDS-GC ALGORITHM

In DMDS-GC algorithm, rather than building a local map for each sensor in MDS-MAP(P), the whole network is split into LCs, where the distance estimations of all pairs of nodes are collected by the cluster head and the relative localization map of the LC is calculated through MDS technique. Then the relative global map of network can be obtained by merging all relative local maps of LCs together. The definition of LC is

$$\mathrm{LC}(k,h)=\{j \mid hop_{k,j} \le h\} \quad (1)$$

where $hop_{k,j}$ is the hop number from cluster head $k$ to node $j$ in LC($k$,$h$). The DMDS-GC algorithm can be summarized as:

(1) Let the value of $h$ be equal to $m$ and partition all nodes into several LCs. During the construction of LC, the shortest path distances between each pair of sensors are computed through the Dijkstra algorithm [16, 20] and collected by the head.

(2) In each LC, the shortest path distances are corrected by the geometric method to get the actual distance.

(3) According to the corrected shortest path distance, the relative positions of nodes in each LC are computed using classical MDS algorithm and SMACOF algorithm [21].

(4) Based on the relative local maps of all LCs, perform mergence to get the relative global map of network or absolute coordinates with the help of anchors on a powerful node (such as gateway node) or a computer.

### *A. The Construction of LCs*

In a network, the distribution of LCs must be taken into account carefully. If each LC includes too many nodes, the loads of head nodes must be heavy and the management of cluster is also a complex task. On the contrary, a cluster can not include few nodes. This is because that there should be some redundant nodes for the merging of relative local maps of LCs. Therefore, the value of $h$ is set as $m$ in DMDS-GC, and a heuristic search algorithm is employed to construct the LCs. The steps for LC construction can be summarized as:

*Step1*. Initialize the counter ($Lct_i$ = 0), timer ($Tm_i$ = 0), node table ($Nt_i$ = $\phi$) and maximum timeout ($T_{max}$) for node $i$. At the beginning, $Tm_i$ is closed.

*Step2*. The process of LC construction starts from anyone node in network. This node sets $Lct_{\mathrm{ID}}$ as 1, broadcasts the construction request LC_B(ID, $h_{\mathrm{ID}}$) and starts the $Tm_{\mathrm{ID}}$ at the same time, where $h_{\mathrm{ID}}$ is equal to 1.

*Step3*. When the node $i$ receives a message from node $j$, the algorithm runs as follows:

*(1)* If node $j$ is not included in $Nt_i$, node $i$ stores it, meaning that node $i$ belongs to LC($j$, $h$). Three are three cases where the algorithm can switch to.

*Case 1*. If $h_j$ is less than $m$, node $i$ let $h_j$ and $Lct_i$ increase by one. Then it broadcasts construction request LC_B($j$, $h_j$), meanwhile, restarts timer with $Tm_i$ = 0.

*Case 2*. If $h_j$ is equal to $m$ and $Lct_i$ is less than 2, node $i$ let $Lct_i$ increase by one and restarts timer with $Tm_i$ = 0. Then it broadcasts construction request LC_B($i$, $h_i$), where $h_i$ is equal to 1.

*Case 3*. If $h_j$ is equal to $m$ and $Lct_i$ is greater than or equal to 2, node $i$ let $Lct_i$ increase by one and restarts timer with $Tm_i$ = 0.

*(2)* If node $j$ has *been* already included in $Nt_i$, node $i$ ignores this message.

*Step4*. Node $i$ keeps on monitoring the construction request of network. If $Tm_i$ is less than or equal to $T_{max}$ and node $i$ receives a construction request, the algorithm switches to *Step3*. Other, if $Tm_i$ is more than $T_{max}$, There are two cases:

*Case 1*. If Node $i$ does not belong to any LC, it broadcasts construction request LC_B($i$, $h_i$), where $h_i$ is equal to 1. Then the algorithm ends.

*Case 2*. If node $i$ has been already included in anyone of LCs, the algorithm ends.

Through these steps, the number of common nodes between neighbor LCs is not less than ($m$+1), indicating that all local maps of LCs in m-D space can be merged together successfully.

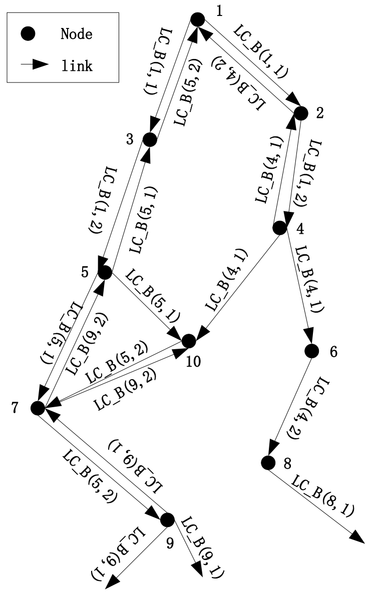


Figure 1. Process of localization clusters construction

In the process of construction of LCs, the variable $h$ determines the minimum number of common nodes between neighbor, and the variable $Lct$ is used to decrease the number of LCs in the network. Here, the 2-D ($m$=2) case is taken as the example for principle explanation as shown in Fig. 1. Node 1 starts the construction of LCs and broadcasts message LC_B(1,1) with $Lct_1$=1. Node 2 and node 3 receive this message and broadcast message LC_B(1,2) with $Lct_2$=1 and $Lct_3$=1. In this scenario, node 4 and node 5 can get message LC_B(1,2). Then, LC(1,2) is built including node 1, node 2, node 3, node 4 and node 5. Because the parameter *h* in LC(1,2) is 2, node 4 and node broadcast the message LC_B(4,1) and LC_B(5,1), respectively. Through message transferring, LC(4,2) and LC(5,2) can be built with the member of LC(4,2)={4,2,6,10,1,8} and LC(5,2)={5,3,7,10,1,9}. It is clear that LC(1,2) and LC(4,2) are neighbors and have common node {1,2,4}. In 2-D space, three common nodes guarantee the success of coordinate transformation between LC(1,2) and LC(4,2). In additional, node 10 can get message LC(9,2), but it can't start to built LC(10,2) with the variable $Lct_{10}$ larger than 2. Thus, this restrictive condition can effectively reduce the number of LCs.

### *B. The Shortest Path Distance Correction Algorithm*

In phase of LC's construction, the shortest path distances between all pairs of nodes are collected. In graph theory, the shortest path problem is the problem of finding a path between two nodes such that the sum of the weights of its constituent edges is minimized. In WSN, the weight of constituent edge is equal to the distance between neighbors. Therefore, the difference between the shortest path distance and the actual distance is significant due to the deployment of nodes in network. In DMDS-GC, parts of these shortest path distances can be corrected by the geometric algorithm according to its position. Here, the 3-D ($m$=3) case is taken as the example for principle explanation as shown in Fig. 2. Node B, C, D and E are within a direct neighborhood, indicating that the distances *BC*, *BD*, *CD*, *BE*, *CE* and *DE* are known. Without loss of generality, assume that node A can be one hop or multiple hops away from the other nodes in Fig. 2. When the distances *AC*, *AD* and *AB* are also known, the distance between node A and E can be computed by solving Eq. (2). If node B, C and D are within the neighborhood of node A, the distance *AC*, *AD* and *AB* can be directly measured. Otherwise, these distances can be estimated iteratively from neighbor nodes which are close to node A.

$$\left\| x_i - x_j \right\|_F = L \tag{2}$$

where *j* is different from *i* and *L* is equal to *BC*, *BD*, *CD*, *BE*, *CE*, *DE*, *AC*, *AB* or *AD*. Unfortunately, Eq. (2) has two possible solutions with the positions of E and $E^*$, which are mirror symmetry about plane BCD. According to these known conditions, node E cannot make reliable decision on which of *AE* and $AE^*$ is the correct distance estimation.

To solve this problem, other information will be necessary. As shown in Fig. 2, if the node F exists which is the neighbor node of node B, D and E, another equation set like Eq. (2) will be obtained. Thus, two possible results of the distance between node A and E are also available and denoted as *AE* and $AE^{**}$, in which one will correspond to the correct distance *q*. Considering that *J* equation sets with the style of Eq. (2) can be build and distance measurement error exists, the solution sets $\{q_i, q_i^*\}$ maybe have no common solution for *q* estimation. In this situation, the distance estimation *q* can be calculated through Eq. (3)

$$q = \{x \mid \min[\sum_{i=1}^{J} \min(\left|x - q_i\right|, \left|x - q_i^*\right|)]\} \tag{3}$$

where $\|\bullet\|$ denotes the absolute value

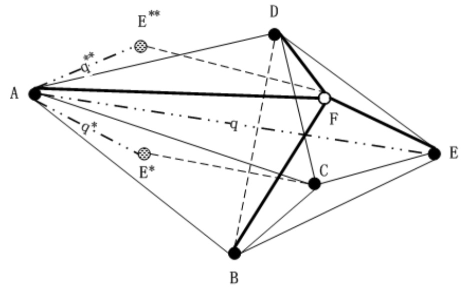


Figure 2. Solving alias case byadding another neighbor node

### *C. Calculating the Relative Local Map of LC*

Assume that matrix $X^{(k)} = (x_1^{(k)}, x_2^{(k)}, \cdots, x_{N_k}^{(k)})$ is the positions of nodes in LC(*h*, *k*), where $x_i^{(k)} = (x_{i,1}^{(k)}, x_{i,2}^{(k)}, \cdots, x_{i,m}^{(k)})^T$ is the coordinates of node *i*. In the process of position calculation, a cost function (stress function) is defined as

$$Stress_k(X^{(k)}) = \sum_{i<j\le N_k} w_{i,j}\,(p_{i,j}^{(k)} - \left\|x_i^{(k)} - x_j^{(k)}\right\|_F)^2 \tag{4}$$

where $p_{i,j}^{(k)}$ is the shortest path distance between node *i* and *j*, $w_{i,j}$ is the weight of $p_{i,j}^{(k)}$ and $N_k$ is the number of nodes in LC(*h*, *k*). To decrease the localization error, $w_{i,j}$ is defined as

$$w_{i,j} = \begin{cases} 2^{1-hop_{i,j}} & \text{, if } p_{i,j} \text{ is not calibrated} \\ 1 & \text{, if } p_{i,j} \text{ is calibrated} \end{cases} \tag{5}$$

In LC(*h*, *k*), the relative coordinates of nodes are determined by minimizing the Eq. (4). Here, this problem is solved by iteratively using quadratic majorizing functions as in SMACOF [21]. Define the relative positions of nodes at iteration *s* as $X^{(k,s)} = (x_1^{(k,s)}, x_2^{(k,s)}, \cdots, x_{N_k}^{(k,s)})$ . Then the update procedure for $X^{(k,s)}$ can be get as

$$X^{(k,s+1)} = X^{(k,s)} \cdot (\sum_{i<j\le N_k} w_{i,j}\, p_{i,j}^{(k)} \left\|x_i^{(k,s)} - x_j^{(k,s)}\right\|_F^{-1} A^{(i,j)}) \cdot V^{-1} \tag{6}$$

where $V = [v_{i,j}]$ is a $N_k \times N_k$ matrix with $v_{i,i} = \sum w_{t,u}\ (t < u \le N_k)$ and $v_{i,j} = -w_{i,j}$, and $A^{(i,j)} = [a_{t,u}]$ is also a $N_k \times N_k$ matrix with $a_{i,i} = a_{j,j} = 1$, $a_{i,j} = a_{j,i} = -1$ and all other elements are zeros. If $V^{-1}$ doesn't exist, it should be replaced by $V^+ = (V + e \cdot e^T / N_k)^{-1} - e \cdot e^T / N_k$ , where *e* is a

$N_k$-column identity vector. The iteration includes two steps:

*Step1*. If all the shortest path distances between nodes are known, the result of MDS is used as $X^{(k,0)}$. Otherwise, it is randomly initialized.

*Step2*. According to Eq. (6), add $s$ by one and recompute the $X^{(k,s)}$ until it converges

*D. Mergence of Relative Local Maps*

In 3-D space, the rotation matrices for point $(x,y,z)^T$ is

$$T_{\alpha,\beta,\gamma}=\begin{pmatrix}\cos\alpha & -\sin\alpha & 0\\ \sin\alpha & \cos\alpha & 0\\ 0 & 0 & 1\end{pmatrix}\times\begin{pmatrix}1 & 0 & 0\\ 0 & \cos\beta & -\sin\beta\\ 0 & \sin\beta & \cos\beta\end{pmatrix}\times\begin{pmatrix}\cos\gamma & 0 & \sin\gamma\\ 0 & 1 & 0\\ -\sin\gamma & 0 & \cos\gamma\end{pmatrix} \quad (7)$$

By $T_{\alpha,\beta,\gamma}$ the $x$-$y$ plane, $y$-z plane and $z$-$x$ plane can be rotated counterclockwise through angle α, β and γ, respectively. The reflection matrix for plane (A$x$+B$y$+C$z$=0) is

$$T_s=(A^2+B^2+C^2)^{-0.5}\times\begin{pmatrix}-A^2+B^2+C^2 & -2AB & -2AC\\ -2AB & A^2-B^2+C^2 & -2BC\\ -2AC & -2BC & A^2+B^2-C^2\end{pmatrix} \quad (8)$$

Suppose that there are $N_C$ common sensors in LC($k$, $h$) and LC($u$, $h$). The positions for these common sensors are $(\overline{x}_1^{(k)},\overline{x}_2^{(k)},\cdots,\overline{x}_{N_C}^{(k)})$ and $(\overline{x}_1^{(u)},\overline{x}_2^{(u)},\cdots,\overline{x}_{N_C}^{(u)})$. If $\overline{x}_1^{(k)}$ and $\overline{x}_1^{(u)}$ are used as origin, the method for converting the coordinates of common nodes in LC($u$, $h$) to those in LC($k$, $h$) is given by

$$\min\sum_{i=1}^{N_C}\left\|(\overline{x}_i^{(k)}-\overline{x}_1^{(k)})-T_{\alpha,\beta,\gamma}T_s(\overline{x}_i^{(u)}-\overline{x}_1^{(u)})\right\|_F^2 \quad (9)$$

If $G=T_{\alpha,\beta,\gamma}T_s$ can meet Eq. (9), the coordinate transformation from LC($u$, $h$) to LC($k$, $h$) can be obtained as

$$x_i^{(u\to k)}=G(x_i^{(u)}-\overline{x}_1^{(u)})+\overline{x}_1^{(k)} \quad (10)$$

where $x_i^{(u\to k)}$ is the relative coordinate $x_i^{(u)}$ in LC($k$, $h$). If sufficient anchors exist, the absolute position of node also can be obtained through this method

## IV. Simulation and Analysis

To validate the DMDS-GC algorithm, we conducted simulations using OMnet++ 4.0. The simulations were executed on a network consisting 200 sensor nodes, which were randomly distributed within a $10r\times10r\times10r$ cube. Each case was repeated 100 times and the mean values of results were taken to be the final result. To evaluate the performance of DMDS-GC, system parameters and performance parameters are listed below:

(1) To ensure that the network is connected, the initial radio radius ($R$) is 1.5$r$, resulting in average connectivity 2.3.

(2) The directly measured distance $d^*$, is modelled as

$$d^*=d\times(1+N(0,\delta)) \quad (11)$$

where $d$ is the true distance and $N(\cdot,\cdot)$ is a Gaussian random number with mean 0 and variance $\delta$.

(3) The average localization error ($ALE$) is defined as

$$ALE=\sum_{i=1}^{N}\left\|X_{est,i}-X_{real,i}\right\|_F/(N\times r) \quad (12)$$

where $X_{est,i}$ and $X_{real,i}$ are the estimation position and true position of node $i$, respectively.

Fig. 3 shows the performance of DMDS-GC as a function of R when the number of anchors is 4. The range of $R$ is from 1.5$r$ to 2.75$r$, which leads to the average connectivity from 2.3 to 12.6. For all radio radiuses, the results of DMDS-GC outperform those of other algorithms. Especially, the ALE curves of DMDS-GC are much lower than the ones of MDS-MAP(P,R), especially when the $R$ becomes bigger. This is because that more neighbor nodes will be involved in geometric correction with the increase of $R$, which also implies that the distance correction is more feasibly preferable than the refinement techniques in performance improvement. Furthermore, it can be particularly observed that the curve of DMDS-GC descends slowly and then even tends to be flat when $R$ is larger than 2.0r, implying that the involvement of more equation sets like Eq. (2) can not significantly improve the distance estimation error.

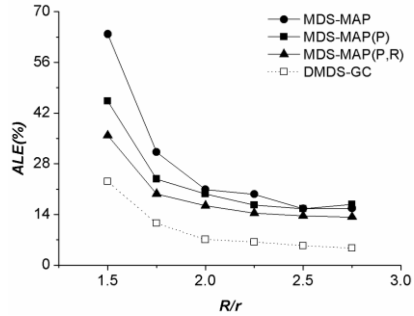


Figure 3. ALE changes with R when the number of anchors is 4.

Fig. 4 shows the performance of DMDS-GC as a function of the number of anchors with R=2.0r. It is clear that $ALE$ is large when a small number of anchors are dispersed. When the number of anchors reaches 10, the variation of *ALEs* tends to be slowed down although there is a little fluctuation and still a trend to decrease with the increase of number of anchors. In addition, it should be noted that the precision of localization almost remains the same with the increase of number of anchors when a threshold reaches. This behavior is explained by the fact that the distance correction algorithm is based on geometric technology and one-hop distance measurement, so more anchors hardly result in more accurate distance estimation when the density of node is enough to perform the distance calibration.

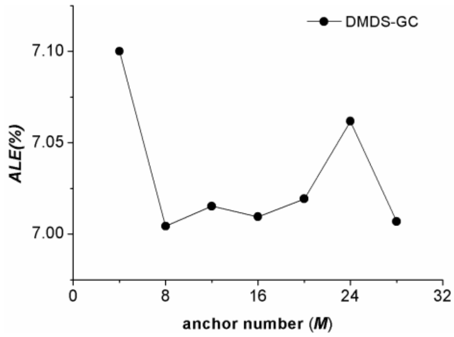


Figure 4. ALE changes with the number of anchors when R is 2.0r.

The sensitivity of *ALE* to range error is a major concern for range-based localization algorithm. Fig. 5 shows the effects of increasing of range error on *ALE* when *R* is 1.75*r* with 8 anchor nodes in network. It can be seen that the localization accuracy consistently degrades as the range error gradually increases including all localization algorithms. When the range error ranges from 0 to 0.2, the *ALE* almost keeps invariant except MDS-MAP algorithm. This is because the MDS-MAP algorithm is simply applied the MDS technique without other optimization technique. When the value of range error is larger than 0.3, the performances of DMDS-GC algorithm is still good. This is attributed to that the geometric correction and SMACOF algorithm can effectively inhibit the increases of localization error. Among these algorithms, the slope of *ALE* versus range error for DMDS-GC is much gentler than those for the other three, suggesting that the DMDS-GC is more robust to the range error, mainly due to that Eq. (3) make the distance estimation more insensitive to the range error.

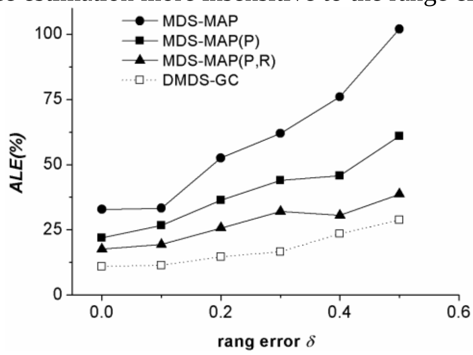


Figure 5. The effects of range error on ALE when R is 1.5r with 4 anchors

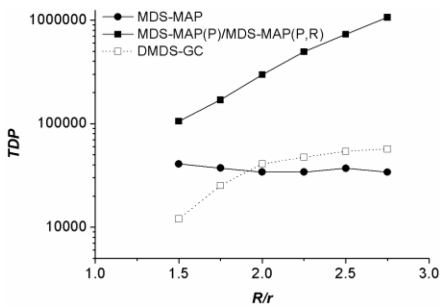


Figure 6. The number of data packets transmitted changes with R..

In the MDS-based localization algorithm, the energy consumption of distances measurement equals to each other. Therefore, in the process of localization, the energy consumption is dominated by the communication cost. Here, the number of transmitted data packets (TDP) during localization is used for analysis of communication cost. The value of parameter TDP of these algorithms is depicted in Fig. 6. It can be seen that the communication cost of MDS-MAP(P)/(P, R) algorithm increases rapidly as the increase of the number of communication radius. This is because the increase in the network linking-density results in rapid increase of the number of one-hop neighbor nodes, which causes the increase of energy consumption. Compared with MDS-MAP(P), the TDP of DMDS-GC is much lower because only small number of LCs need to be constructed. Although the TDP of DMDS-GC is a little higher than that of MDS-MAP when *R*>2.0*r*, the former has much smaller localization error. In other words, the DMDS-GC has higher energy efficiency. Additionally, it is also can be concluded that the communication radius should decrease for energy saving under the premise of meeting the requirement of localization precision.

## V. SIMULATION AND ANALYSIS

In this paper, the DMDS-GC algorithm, which is based on MDS and geometric correction, is presented. Through the construction of localization cluster, entire network is split and the shortest path distance is corrected. After the calculation of relative local map for each LC by MDS technique, the task of node localization is achieved by mergence operation. Simulation results show that the DMDS-GC algorithm has the advantages of precision, strong robustness and high energy efficiency, which is suitable for node localization in WSN.

## ACKNOWLEDGMENT

This study is supported by the Key Program for Science and Technology Development of Henan Province (No.102101210600) and the Doctoral Foundation Program of Zhengzhou University.

## REFERENCES

[1] I. F. Akyildiz, W. L. Su, Y. Sankarasubramaniam and E. Cayirci, “A Survey on Sensor Network” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102-114, 2002.

[2] N. Bulusu, J. Heidemann and D. Estrin, “GPS-less Low Cost Outdoor Localization for Very Small Device” *IEEE Personal Communications*, vol. 7, no. 5, pp. 28-34, 2000.

[3] S. Yi, R. Wheeler, and P. J. F. Markus, “Positioning using Local Maps,” *Ad Hoc Networks*, vol. 4, no. 2, pp. 240–253, 2006.

[4] M. Shahrokhzadeh, A. T. Haghighat, F. Mahmoudi and B. Shahrokhzadeh, “A Heuristic Method for Wireless Sensor Network Localization.” *Procedia Computer Science*, vol. 5, pp. 812-819, 2011.

[5] Y. Zhang, S. Liu, X. Zhao and Z. Jia, “Theoretic Analysis of Unique Localization for Wireless Sensor Networks.” *Ad Hoc Network*, vol. 10, no. 3, pp. 623-634, 2012.

[6] G. Wu, S. Wang, B. Wang, Y. Dong and S. Yan, “A Novel Range-free Localization Based on Regulated Neighborhood Distance for Wireless Ad Hoc and Sensor

Network.” *Computer Networks*, vol. 56, no. 16, pp. 3581-3593, 2012.
[7] X. H. Kuang, H. H. Shao and R. Feng, “A New Distributed Localization Scheme for Wireless Sensor Networks.” *Acta Automatica Sinica*, vol. 34, no. 3, pp. 344-348, 2008M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.
[8] P. Kulakowski, J. V. Alonso, E. E. Lopez, W. Ludwin and J. G. Haro, “Angle-of-arrival Localization Based on Antenna Arrays for Wireless Sensor Networks.” *Computers & Electrical Engineering*, vol. 36, no. 6, pp. 1181-1186, 2010.
[9] M. Gholami, N. Cai and R. W. Brennan, “An Artificial Neural Network Approach to the Problem of Wireless Sensors Network Localization.” *Robotics and ComputerIntegrated Manufacturing*, vol. 29, no. 1, pp. 96-109, 2013.
[10] H. Lv, Y. N. Wu and P. Bao, “A Localization Method of Wireless Sensor Network Based on Two-Hop Focus.” *Procedia Engineering*, vol. 15, pp. 2021-2025, 2011.
[11] B. Eric, “Graph Multidimensional Scaling with Selforganizing Maps.” *Information Sciences*, vol. 143, no. 1-4, pp. 159-180, 2002.
[12] S. Yi, W. Ruml and Y. Zhang, “Localization from Mere Connectivity in Sensor Networks,” *In Proceedings of the 4th ACM Int’l Symp on Mobile Ad Hoc networking & computing, New York, USA*, pp. 201-212, June 2003.
[13] S. Biaz and Y. M. Ji, “Precise Distributed Localization Algorithm for Wireless Networks,” *In Proceeding of the 6th IEEE International Symposium on WoWMoM, Taormina, Italy*, pp. 388-394, June 2005.
[14] K. W. Cheung and H. C. So, “A Multidimensional Scaling Framework for Mobile Location Using Time-of-Arrival.” *IEEE Transaction on Signal Procesing*. vol. 53, no. 2, pp. 460 -470, 2005.
[15] X. Ji and H. Zha, “Sensor Positioning in Wireless Ad-hoc Sensor Network Using Multidimensional Scaling,” *In 23th Annual Joint Conference of the IEEE Computer and Communications Societies, Hong Kong, China*, pp. 2652-2661, 2004.
[16] S. Yi and W. Ruml, “Improved MDS-Based Localization.” *In Proceedings of the 23rd Conference of the IEEE Communications Society, Hong Kong, China*, pp. 2640-2651, 2004.
[17] G. Q. Mao, B. Fidan and B. D. O. Anderson, “Wireless Sensor Network Localization Techniques.” *Computer Network*, vol. 51, no. 10, pp. 2529-2553, 2007.
[18] S. H. Bae, J. Qiu and G. Fox, “Adaptive Interpolation of Multidimensional Scaling.” *Procedia Computer Science*, vol. 9, pp. 393-402, 2012.
[19] G. Aloor and L. K. Jacob, “Distributed Wireless Sensor Network Localization Using Stochastic Proximity Embedding.” *Computer Communication*, vol. 33, no. 6, pp. 745-755, 2010.
[20] F. B. Zhan, “Three Faster Shortest Path Algorithms on Real Road Networks.” *Journal of Geographic Decision Analysis*, vol. 1, no. 1, pp. 69-82, 1997.
[21] J. D. Leeuw, “Applications of Convex Analysis to Multidimensional Scaling.” *Recent developments in statistics, North-Holland Publishing Company, Amsterdam*, pp. 133-146, 1977.

**Yingqiang Ding** is lecture at college of information engineering, Zhengzhou University. He has received PhD degree from Tianjin University, and his major is electronic engineering. His research interests include wireless sensor network, digital signal processing, information theory.

**Dongsheng Yang** is a graduate student in college of information engineering, Zhengzhou University. He has received bachelor degree from Zhengzhou University, and his major is circuit design. His research interests include RF development, optimization technique and machine-to-machine technique.

**Gangtao Han** is a doctor candidate student in college of information engineering, Zhengzhou University. He has received bachelor degree from Zhengzhou University, and his major is signal processing. His research interests include digital signal processing and localization technique.