Cooperative Coverage of Mobile Robots with Distributed Estimation and Control of Connectivity

Xiaoli Li[a,b], Shuguang Zhao[a], and Haiqin Xu[a]
[a] School of Information Science and Technology, Donghua University, Shanghai 201620, China
[b] Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai 200240, China
skill-li@163.com, xlli@dhu.edu.cn

***Abstract*—This paper deals with the discrete-time connected coverage problem with the constraint that each robot of group can only sense and communicate in the local range. In such distributed framework, the algebraic parameter of connectivity, that is, the second smallest eigenvalue of topology Laplacian, is estimated by introducing the minimal-time consensus algorithm to guarantee the high cooperation efficiency. Since no certain edges are imposed to be preserved, the method of keeping the second smallest eigenvalue positive reserves a sufficient degree of freedom for the motion of robots in the connected group. Furthermore, a self-deployment algorithm is developed to disperse the robots with the precondition that the resulting second smallest eigenvalue keeps positive at each time-step. At last, we prove that the proposed algorithm steers each pair of neighbor robots to reach the largest objective distance from each other. It implies that the distributed optimal coverage is achieved under the connectivity constraint.**

***Keywords*—connected coverage; distributed cooperative control; minimal-time consensus; eigenvalue estimation**

I. INTRODUCTION

Connected coverage control has attracted much attention due to its extensive applications including search and rescue, surveillance, emergency management, weather monitoring, and reconnaissance[1]. In order to sense a large area or to extract much information about the environment in interest, the system of robots need spread out as dispersedly as possible, while the whole group should be restricted in a limited region such that the extracted information can be transmitted to the outer station.

Coverage and connectivity maintenance have been considered as an overall static optimal problem in [2]-[3] or incremental optimal localization problem by deploying the mobile devices one-at-a-time in [4]. For distributed coverage control, many strategies are also developed in the literature. The work in [5]-[6] adopts local dispersion and the distributed self-deployment algorithm to induce a global coverage. Under the underlying assumption that the communication and sensing radius of device are controllable, optimal coverage is studied in [7]-[8] based on Voronoi partition and solved in a distributed style. The distance and orientation control is used to reach ideal coverage and reliable connectivity in [9]. In [10], the maximal coverage problem is considered with the constraint that each agent keeps at least k neighbors. This scheme may prevent the destruction of connectivity but with the lost of coverage performance. In order to improve the coverage performance, the core topology, a special spanning subgraph of the topology, is developed in [11], which keeps the connectivity of the communication topology and reserves the degree of freedom for the neighbors to traverse over the distance approaching the largest value with each other.

In fact, connectivity can be measured by the second smallest eigenvalue λ_2 of the topology Laplacian [12]. Hence, a more relaxed connectivity condition can be obtained by keeping λ_2 larger than zero. However, the global property of λ_2 prevents its extensive applications in cooperative control, which motivates the research of the distributed estimation algorithm for λ_2. For example in [13], λ_2 is estimated by utilizing the infinite-time consensus course.

This paper aims at solving the distributed connected coverage problem without compulsively preserving some certain edges. Connectivity is maintained by estimating and controlling λ_2 in a pure distributed manner. Compared to our previous work [11], the freedom degree of robot is further relaxed to support a better coverage performance of group. The distributed minimal-time cooperation algorithm adopted for λ_2 estimation guarantees that connectivity is judged with faster velocity than that in [13] so that the real-time property of system is improved. Based on the estimated λ_2, a distributed self-deployment algorithm is developed to steer the robots away from their neighbors on the premise of $\lambda_2 > 0$.

Notation: The set of connected graphs is denoted as $\mathbb{C}$. Let $\mathbb{R}_+$ denote the set of positive real numbers, $\mathbb{N}$ denote the set of natural numbers, and $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. For any $x \in \mathbb{R}^2$, $\bar{B}(x,r)$ is the set of points whose distances to x are not larger than r, Let $\|\cdot\|$ denote the Euclidean vector norm, and $e_r^T = \left[0,\cdots,0,1_{r\text{-}th},0,\cdots,0\right] \in \mathbb{R}^{1\times N}$.

II. PRELIMINARY

This paper discusses a group of mobile robots roving with limited motion ability. The discrete-time motion model of individual robot is characterized by

$$x_i(k+1) = x_i(k) + u_i(k) \tag{1}$$

where $x_i(k) \in \mathbb{R}^2$ denotes the position of the i-th robot at time k, $i \in \{1,2,...,N\}$, $u_i(k)$ is its corresponding control

This work is supported by the National Natural Science Foundation of China (Grant No.61203073, 61271114), the Research Fund for the Doctoral Program of Higher Education (Grant No.20120075120008), the Fundamental Research Funds for the Central Universities (Grant No. 12D10412, 12D10423), and the Foundation of Key Laboratory of System Control and Information Processing, Ministry of Education, P.R. China.

978-1-4673-5769-2/13/$31.00 ©2013 IEEE

law at that time, which always satisfies $\left\|u_i(k)\right\| \le \chi$ given the maximum step length of robot χ. Suppose that each robot is equipped with some sensing and communication devices to detect the environment around it and to communicate within some certain range. Let r_c and r_s denote the communication and sensing radius of robot, respectively. The communicating neighbors of robot i is thus naturally achieved by

$$N_i = \left\{ j \mid \left\| x_j - x_i \right\| \le r_c, j \ne i \right\} \quad (2)$$

which implies the information from the neighbors in N_i is also acquirable for robot i.

Cooperative coverage aims at coordinating the position of each robot i based on information from its neighbors in N_i, so that the whole steered group can extract information as much as possible from the environment in interest. The interactions in such distributed system can thus be modeled by the communication topology $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ represents the N robots and $\mathcal{E}$ denotes all their communication connections

$$\mathcal{E} = \left\{ (i,j) \in \mathcal{V} \times \mathcal{V} \mid i \in N_j, j \in N_i, j \ne i \right\}$$

Since N_i depends on the relative position of the robots from (2), $\mathcal{E}$ is evolving during the motion process. So the distributed control law $u_i(k)$ should also guarantee the connectivity requirement of time-variant communication topology, i.e., $G(k) \in \mathbb{C}$.

According to the above discussion, the control objective of distributed connected coverage problem is to spread out the mobile robot group to the largest possible sensing coverage over the environment via the self-deployment of robots under the precondition that $G \in \mathbb{C}$ is not destroyed. It is worth mentioning that both coverage and connectivity maintenance should be considered in a distributed manner and with the physical limitation that each individual robot can only obtain local information from its neighbors. From each robot's local perspective, coverage objective is said to be achieved, if any pair of neighboring robots has the largest possible distance (the communication radius r_c defined above) from each other, whereas connectivity is intrinsically a global property and is not facile for the individual robot. Therefore a purely distributed estimation of connectivity will be studied before presenting the distributed connected coverage self-deployment algorithm.

III. FINITE-TIME DISTRIBUTED ESTIMATION OF GROUP CONNECTIVITY

Given the undirected communication topology G, the induced Laplacian matrix is symmetric and denoted by $L = L(G) = \left[l_{ij} \right] \in \mathbb{R}^{N \times N}$, where $l_{ij} = -1$ and $l_{ii} = |N_i|$. Let $0 = \lambda_1(L) \le \lambda_2(L) \le \cdots \lambda_N(L)$ be the eigenvalues of $L(G)$ in increasing order. The eigenvector respect to $\lambda_i(L)$ is denoted by $v_i(L)$. It is known that some important structure characteristics of G can be captured by its algebraic representations.

Lemma 1[14]. Given an undirected topology G,

i) $\lambda_1 = 0$, and $\lambda_2 > 0$ if and only if $G \in \mathbb{C}$.

ii) $\left\langle v_i(L), v_j(L) \right\rangle = v_i(L)^T v_j(L) = 0$ for any $1 \le i, j \le N$, $i \ne j$. Especially, $v_1 = 1_N$ where 1_N is one vector with dimension N.

$\lambda_2(L)$ serves as the measure of group connectivity from Lemma 1. Hence, if the parameter $\lambda_2(L)$ can be estimated with only local information and be controlled to be larger than zero by the local cooperation of robot, the objective of connectivity preservation is achieved for the cooperative group. At the same time, higher degree of freedom is remained for the motion of each robot compared with our previous research of imposing strict motion constraint on some certain edges in $\mathcal{E}$ [11]. In this article, the distributed estimation of $\lambda_2(L)$ is developed by utilizing the following technique of minimal-time computation of consensus value.

A. *Distributed minimal-time average consensus value computation*

Suppose each robot perform the following standard discrete-time consensus scheme about arbitrary information

$$\varsigma_i(\tau+1) = \varsigma_i(\tau) + \varepsilon \sum_{j \in N_i} \left(\varsigma_j(\tau) - \varsigma_i(\tau) \right)$$

Let $d_{\max} = \max\left\{ |N_i|, i = 1, \ldots, N \right\}$, ε is the sampling time satisfying $0 < \varepsilon < \dfrac{1}{d_{\max}}$. Aggregate the information of each robot at time τ into a vector $\varsigma(\tau) = col(\varsigma_1(\tau), \varsigma_2(\tau), \ldots, \varsigma_N(\tau))$. The associated discrete-time consensus dynamics on the whole group is given by

$$\varsigma(\tau+1) = (I_N - \varepsilon L)\varsigma(\tau) \triangleq A\varsigma(\tau) \quad (3)$$

where I_N denotes the identity matrix of dimension N, L is the Laplacian matrix of G.

From the viewpoint of each robot, the measurable local information of a robot labeled r is described by.

$$\varsigma_r(\tau) = e_r^T \varsigma(\tau) \quad (4)$$

The *minimal polynomial* [15]-[16]associated with the matrix pair $\left[A, e_r^T \right]$ is the unique monic polynomial of the smallest degree

$$q_r(t) \triangleq t^{D_r+1} + \sum_{j=0}^{D_r} \alpha_{r,j} t^j$$

such that $e_r^T q_r(A) = 0$. It is thus straightforward to show that

$$e_r^T A^{D_r+1} + e_r^T \alpha_{r,D_r} A^{D_r} + \cdots + e_r^T \alpha_{r,1} A + e_r^T \alpha_{r,0} = 0$$

Multiplying $\varsigma(\tau)$ to both sides of the above equation, one may easily find that

$$\varsigma_r(\tau + D_r + 1) + \alpha_{r,D_r} \varsigma_r(\tau + D_r) + \cdots + \alpha_{r,0} \varsigma_r(\tau) = 0 \quad (5)$$

According to the Z-transform of the expression (5) and the final value theorem, it is obtained that

$$\lim_{\tau\to\infty}\varsigma_r(\tau)=\frac{\left[\varsigma_r(D_r),\varsigma_r(D_r-1),\cdots,\varsigma_r(0)\right]\beta}{1^T\beta} \tag{6}$$

where

$$\beta=\begin{bmatrix}1\\ \beta_{D_r}\\ \beta_{D_r-1}\\ \vdots\\ \beta_1\end{bmatrix}=\begin{bmatrix}1\\ 1+\alpha_{r,D_r}\\ \alpha_{r,D_r-1}+\beta_{D_r}\\ \vdots\\ \alpha_{r,1}+\beta_2\end{bmatrix} \tag{7}$$

Define the Hankel matrix of $\{\varsigma_r(0),\varsigma_r(1),\cdots,\varsigma_r(2\tau)\}$ by

$$\Gamma\{\varsigma_r(0),\varsigma_r(1),\cdots,\varsigma_r(2\tau)\}\triangleq \begin{bmatrix}\varsigma_r(0) & \varsigma_r(1) & \cdots & \varsigma_r(\tau)\\ \varsigma_r(1) & \varsigma_r(2) & \cdots & \varsigma_r(\tau+1)\\ & & \vdots & \\ \varsigma_r(\tau) & \varsigma_r(\tau+1) & \cdots & \varsigma_r(2\tau)\end{bmatrix}$$

Then for a general arbitrary initial condition, expect for a set of initial conditions with Lebesgue measure zero[16], the coefficient vector $\alpha_r=\left[\alpha_{r,0}\ \alpha_{r,1}\cdots\alpha_{r,D_r}\ 1\right]^T$ is achieved by computing the kernel of Hankel matrix $\Gamma\{\varsigma_r(0),\varsigma_r(1),\cdots,\varsigma_r(2(D_r+1))\}$, that is

$$\alpha_r\in\ker\left(\Gamma\{\varsigma_r(0),\varsigma_r(1),\cdots,\varsigma_r(2(D_r+1))\}\right) \tag{8}$$

The expression (8) can be used for the computation of the final consensus value via (6) and (7) based on $2(D_r+1)+1$ steps information $\{\varsigma_r(0),\varsigma_r(1),\cdots,\varsigma_r(2(D_r+1))\}$.

A further study finds that β can be obtained directly according to the kernel of the following difference Hankel matrix:

$$\text{fliplr}(\beta)\in\ker\left(\Gamma\{\varsigma_r(1)-\varsigma_r(0),\cdots,\varsigma_r(2D_r+1)-\varsigma_r(2D_r)\}\right) \tag{9}$$

where $\text{fliplr}(\beta)=\left[\beta_1,\beta_2,\cdots,\beta_{D_r},1\right]^T$ is the vector with elements arrayed in the opposite order with β. Besides, the required successive discrete-time steps are reduced to $2D_r+2$.

According to the above analysis, the algorithm of minimal-time consensus value computation is given as follow.

Algorithm $D_{mtcvc}(\varsigma_r(0))$ (The distributed minimal-time consensus value computation algorithm for robot r)

1. Observe successively the individual information $\varsigma_r(\tau)$, $\tau=0,1,\cdots$ and increase the dimension k of $\Gamma\{\varsigma_r(1)-\varsigma_r(0),\cdots,\varsigma_r(2k+1)-\varsigma_r(2k)\}$ until it loses rank.
2. Obtain the kernel of the defective Hankel matrix $\text{fliplr}(\beta)$ based on (9).
3. Compute the final consensus value from (6).

B. Distributed estimation of $\lambda_2(L)$

For estimating the second smallest eigenvalue $\lambda_2(L)$, the power iteration principle is chosen due to the good convergence properties and the cheap iteration step. Especially, it can be implemented in a completely distributed style as is shown below.

Given a matrix B, the standard power iteration can find the eigenvalue with the greatest absolute value and is carried out by the iteration

$$\omega^{(k+1)}=\frac{B\omega^{(k)}}{\left\|B\omega^{(k)}\right\|} \tag{10}$$

where $\omega^{(0)}$ is initialized by a vector with norm 1. When $\omega^{(k)}$ converges to an eigenvector, denoted by $v_{\max}$ satisfying $\|v_{\max}\|=1$, the associated eigenvalue can be obtained by

$$\lambda_{\max}(B)=v_{\max}{}^T B v_{\max}$$

Note that above power iteration scheme on Lapalacian L can only find the eigenvalue with the greatest absolute $\lambda_{\max}(L)$, not the second smallest eigenvalue $\lambda_2(L)$. The gap between them is bridged by the following lemma.

Lemma 2. Consider $G=(\mathcal{V},\mathcal{E})$ with the second smallest eigenvalue of graph Laplaciam $\lambda_2(L)$ and the associated eigenvector $\upsilon_2(L)$. Denote $B=(I_N-\varepsilon L)-\frac{1}{N}1_N1_N{}^T$, where I_N is the identity matrix of order N , 1_N is one vector with dimension N, and $0<\varepsilon<\frac{1}{d_{\max}}\le\frac{1}{N-1}$. Then

$$\lambda_N(B)=1-\varepsilon\lambda_2(L)$$

$$v_N(B)=v_2(L)$$

Proof. Assume a graph Laplacian matrix $L(G)$ with eigenvalues $0=\lambda_1(L)\le\lambda_2(L)\le\ldots\le\lambda_N(L)$ and eigenvectors $v_i(L)$, $1\le i\le N$, where $v_1(L)=1_N$. Then the eigenvalues of matrix $I_N-\varepsilon L$ can be obtained by

$$1-\varepsilon\lambda_N(L)\le 1-\varepsilon\lambda_{N-1}(L)\le\ldots\le 1-\varepsilon\lambda_1(L)=1$$

and the eigenvector respect to $\lambda_i(I_N-\varepsilon L)=1-\varepsilon\lambda_{N-i+1}(L)$ is $v_i(I_N-\varepsilon L)=v_{N-i+1}(L)$. Since $v_i(L)^T v_j(L)=0$ for any $1\le i,j\le N$, $i\ne j$ from Lemma 1, the matrix $B=(I_N-\varepsilon L)-\frac{1}{N}1_N1_N{}^T$ has the following characteristics.

For any $v_i(L)$, $i\ne 1$, it holds that

$$B*v_i(L)=(I-\varepsilon L)*v_i(L)-\frac{1}{N}1_N1_N{}^T*v_i(L)$$
$$=(I-\varepsilon L)*v_i(L)-0*1_N=(1-\varepsilon\lambda_i(L))*v_i(L)$$

and for $v_1(L)=1_N$, the following is true

$$B*1_N=(I-\varepsilon L)*1_N-\frac{1}{N}1_N1_N{}^T*1_N$$
$$=(I-\varepsilon L(t))*1_N-1_N=0*1_N$$

This shows that $v_i(L)$, $1\le i\le N$ is also the eigenvector of B. Besides, the eigenvalue of B corresponding to $v_i(L)$, $2\le i\le N$ is $1-\varepsilon\lambda_i(L)$ and the eigenvalue corresponding to $v_1(L)=1_N$ is 0. Obviously, for the square matrix B of order N, all $1-\varepsilon\lambda_i(L)$, $2\le i\le N$ and 0 constitute the set of eigenvalues, that is,

$$0\le 1-\varepsilon\lambda_N(L)\le 1-\varepsilon\lambda_{N-1}(L)\le\ldots\le 1-\varepsilon\lambda_2(L)$$

and the corresponding eigenvectors are $v_1(B)=1_N$, $v_i(B)=v_{N-i+2}(L)$, $2\le i\le N$. We thus have

$$\lambda_N(B)=1-\varepsilon\lambda_2(L)$$
$$v_N(B)=v_2(L)$$

□

Lemma 2 implies that $\lambda_2(L)$ and $v_2(L)$ can be obtained by implementing power iteration scheme (10) on the matrix $B=(I_N-\varepsilon L)-\frac{1}{N}1_N1_N{}^T$. Let $\omega^{(k)}=\left[\omega_1^{(k)},\ldots,\omega_i^{(k)},\ldots\omega_N^{(k)}\right]^T$. Then

$$D^{(k)}\triangleq B\omega^{(k)}=(I_N-\varepsilon L)\omega^{(k)}-\frac{1}{N}1_N1_N{}^T\omega^{(k)}$$
$$=(I_N-\varepsilon L)\omega^{(k)}-D_{mtcvc}\left(\omega_i^{(k)}\right)*1_N \tag{11}$$

where $D_{mtcvc}\left(\omega_i^{(k)}\right)$ is the average consensus value of all the elements in $\omega^{(k)}$ obtained by the distributed minimal-time consensus value computation of any robot i. It is worthy of note that (11) can be naturally computed in completely distributed manner in the sense that the calculation of each component of $D^{(k)}$ does not depend on any other component of $D^{(k)}$. Utilizing the distributed minimal-time consensus value computation algorithm again, we obtain $\omega^{(k+1)}$ by

$$\omega^{(k+1)}=\frac{D^{(k)}}{\left\|D^{(k)}\right\|}=\frac{D^{(k)}}{\sqrt{N*D_{mtcvc}\left(\left(D_i^{(k)}\right)^2\right)}} \tag{12}$$

When $\omega^{(k)}$ converges to $v_{\max}(B)$, i.e. $v_2(L)$, each robot can estimate the global parameter $\lambda_2(L)$ by

$$\lambda_{2,i}v_{2,i}=L_iv_2=\sum_{j\in N_i}a_{ij}\left(v_{2,i}-v_{2,j}\right)$$

where L_i is the i-th row component of matrix L. So the estimation value of $\lambda_2(L)$ obtained by robot i is

$$\lambda_{2,i}=\frac{\sum_{j\in N_i}a_{ij}\left(v_{2,i}-v_{2,j}\right)}{v_{2,i}} \tag{13}$$

A precise description of the distributed estimation algorithm for $\lambda_2(L)$ is given as follows.

Algorithm $E_{\lambda_2}(i)$ (The distributed $\lambda_2(L)$ estimation algorithm for robot i)

---

1. Initial $\omega_i^{(0)}=\frac{1}{\sqrt{N}}$.
2. Implement iteratively
$$D_i^{(k)}=1-\varepsilon\sum_{j\in N_i}\left(\omega_j^{(k)}-\omega_i^{(k)}\right)-D_{mtcvc}\left(\omega_i^{(k)}\right)$$
and $\omega_i^{(k+1)}=\frac{D_i^{(k)}}{\sqrt{N*D_{mtcvc}\left(\left(D_i^{(k)}\right)^2\right)}}$
until $\omega_i^{(k)}$ converges to the stable value $v_{2,i}$
3. Estimate $\lambda_2(L)$ by (13).

IV. DISTRIBUTED COVERAGE WITH ESTIMATED AND CONTROLLED CONNECTIVITY

Based on only local one-hop information, a distributed self-deployment algorithm is developed in this section to disperse the individual robot as far as possible from each other with the constraint that $\lambda_{2,i}>0$. The self-deployment algorithm does not require a prior environment model, and can be operated in a real-time, local, and purely distributed manner.

Let the local obstacle-free closed environmental region sensed by robot i be denoted by $\bar{E}_i\subset\bar{B}(x_i,r_s)$. The region of any robot i's maneuver capability is limited in $\bar{B}(x_i,\chi)$. Then the feasible motion region that robot i can move in the next time-step is defined by

$$\bar{A}_i\triangleq\bar{B}(x_i,\chi)\cap\bar{E}_i \tag{14}$$

which satisfies that

$$x_i(k+1)=x_i(k)+u_i(k)\in\bar{A}_i(k) \tag{15}$$

Besides, it is clear from (14) that $x_i(k)\in\bar{A}_i(k)$.

The motion control command $u_i(k)$ is generated with the aim of steering robot i far from its communication neighbors $N_i(k)$, but in its feasible region of motion $\bar{A}_i(k)$ and guaranteeing the resulting topology $G(k+1)$ with $\lambda_2(L)>0$. The following artificial repulsion potential field is adopted to guide robot i to move away from its communication neighbors:

$$U_{ij}=\begin{cases}\eta\left(\frac{1}{\left\|x_j-x_i\right\|}-\frac{1}{r_c}\right)^a, & \left\|x_j-x_i\right\|\le r_c\\ 0, & \left\|x_j-x_i\right\|>r_c\end{cases},j\in N_i \tag{16}$$

where r_c is the communication radius, η and a are both positive constants with $a \ge 1$. The design (16) always makes robot i lie in a virtual integrated repulsion potential field formed by its all neighbors N_i. Especially, $U_{ij} \to \infty$ if $\|x_j - x_i\| \to 0$ and $U_{ij} = 0$ if $\|x_j - x_i\| \ge r_c$. The integrated virtual repulsive force F_i imposed on robot i is given by

$$F_i = -grad\left[\sum_{j \in N_i} U_{ij}\right] = \sum_{j \in N_i} F_{ij} \tag{17}$$

Some critical issues should be considered to generate the control input u_i.

C1. According to (15), if $x_i(k) + F_i(k) \notin \bar{A}_i(k)$, the virtual repulsive force F_i should shrink to meet (15).

The detailed manipulation is presented as follows.

i) Denote the move direction by

$$e_i(k) = \frac{F_i(k)}{|F_i(k)|} \tag{18}$$

ii) Since $x_i(k) \in \bar{A}_i(k)$, the vector $e_i(k)$, starting at $x_i(k)$, will intersect with the boundary of $\bar{A}_i(k)$ at least once. Denote the closest intersection point by $z_i(k)$.

iii) $F_i(k)$ shrinks to $z_i(k) - x_i(k)$, i.e.,

$$F_i(k) = z_i(k) - x_i(k) \tag{19}$$

C2. If G' associated with $x_i'(k+1) = x_i(k) + F_i(k) \in \bar{A}_i(k)$ loses connectivity, that is, $\lambda_2(L) = 0$, the virtual repulsive force F_i should shrink to guarantee $\lambda_2(L) > 0$. Furthermore, under the condition of $\lambda_2(L) > 0$, F_i should be chosen as large as possible within acceptable precision.

The concrete manipulation includes:

i) Set the precision parameter p_i.

ii) Solve the following problem

(**Problem P.1**.)

Calculate $\{d_s \mid s = 1, 2, \cdots, p_i, d_s \in \{1, 0\}\}$,

such that the shrunk Force

$$\tilde{F}_i = \left(d_1 * \left(\frac{1}{2}\right) + d_2 * \left(\frac{1}{2}\right)^2 + \cdots + d_{p_i}\left(\frac{1}{2}\right)^{p_i}\right) * F_i \tag{20}$$

results in $\lambda_2(L) > 0$ for the topology associated with $x_i'(k+1) = x_i(k) + \tilde{F}_i(k)$, while the topology associated with $x_i''(k+1) = x_i(k) + \tilde{F}_i(k) + \left(\frac{1}{2}\right)^{p_i} * F_i$ has $\lambda_2(L) = 0$.

Robot i may solve this problem by the following course.

(**Solving Problem P.1**.)

i) $x_i^{temp}(0) = x_i(k)$;

ii) For $s = 1, 2, \cdots, p_i$,

Robot i moves to the temporary position

$$x_i^{temp}(s) = x_i^{temp}(s-1) + d_s * \left(\frac{1}{2}\right)^s * F_i$$

If the resulting $\lambda_{2,i}^{temp}(L) = E_{\lambda_2}(i) > 0$

$x_i^{temp}(s)$ keeps the value achieved by (20);

else

Robot i moves back to $x_i^{temp}(s-1)$;

$x_i^{temp}(s) \leftarrow x_i^{temp}(s-1)$;

end

end

Together with the above discussion on the critical issue, a detailed description of the distributed self-deployment for connected coverage problem is provided.

Algorithm $S_d(x_i(k))$ (The distributed self-deployment algorithm of connected coverage for robot i)

1. Generate the original virtual repulsive force F_i from (17).
2. Calculate the feasible motion region $\bar{A}_i(k)$ from (14). If $x_i(k) + F_i(k) \notin \bar{A}_i(k)$, then F_i shrinks based on (19).
3. Move to $x_i'(k+1) = x_i(k) + F_i(k) \in \bar{A}_i(k)$ and calculate the resulting $\lambda_{2,i}(L) = E_{\lambda_2}(i)$ associate with $x_i'(k+1)$.
4. If $\lambda_{2,i}(L) > 0$, set $u_i(k) = F_i(k)$ and the position of robot i at the next time-step $k+1$ is achieved by $x_i(k+1) = x_i'(k+1) = x_i(k) + u_i(k)$.

   If $\lambda_{2,i}(L) = 0$, solve Problem P.1. with $u_i(k) = \tilde{F}_i(k)$, $x_i(k+1) = x_i(k) + u_i(k)$.

V. SIMULATIONS

In this section, the distributed self-deployment algorithm $S_d(x_i)$ is applied to a system of twenty robots with sensing and communication radius 25. Fig.1(a)-(d) illustrate the group position of the system at different iteration steps.

Initially, the group configuration is randomly distributed in a square region of size 5×5, as shown in Fig. 1(a). With the communication radius $r_c = 25$, it is clear that the initial communication topology is a complete graph and thus is connected. The distributed connected coverage self-deployment algorithm $S_d(x_i)$ runs locally at each robot, driving each robot to spread out gradually with preserved connected topology, as depicted in Fig. 1(b)-(d).

In the final configuration of Fig. 1 (d), the group topology is the simplest connected graph, and each robot keeps the largest possible distance $r_c = 25$ with its neighbors, indicating that Algorithm $S_d(x_i)$ can well solve the connected coverage problem in a distributed manner, as expected. The corresponding coverage performance of the steered system is shown in Fig. 1 (e).

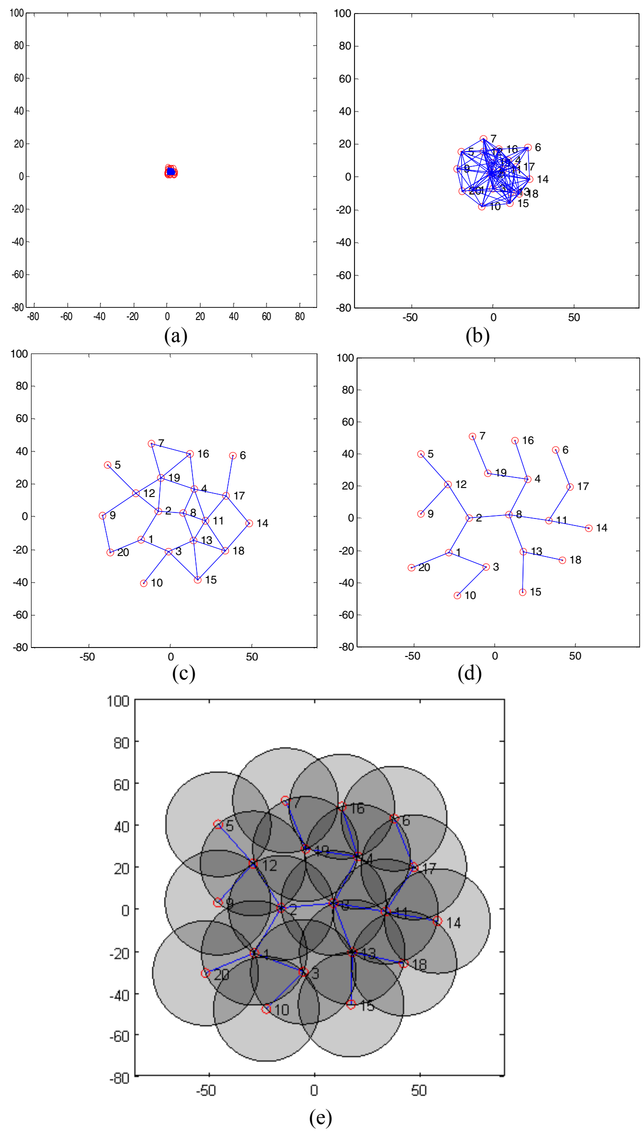

Fig.1. A group steered by Algorithm $S_d(x_i)$, where the red circles mark the states (positions) of robots, the blue lines mark the connections between the neighbor robots, and the grey regions mark the range covered by the robots.

VI. Conclusions

Connectivity maintenance, as a global property, is a very critical issue in the distributed cooperative control, especially for coordinative coverage that heavily involves dispersion behaviors. In such scenario, the robots tend to spread out and may cause the fragmentation of group. In contrast, compulsively preserving some special connections during the maneuvers may be too conservative to achieve the optimal coverage performance. This paper solves connectivity control via estimating the second smallest eigenvalue of group Laplacian, which maintains a sufficient freedom for each robot's motion. Besides, the minimal-time consensus adopted for the eigenvalue estimation is helpful for improving the cooperation efficiency. At last, the distributed self-deployment algorithm is proposed to pursue the optimal coverage while controlling the second smallest eigenvalue positive all the time. Simulation shows that the steered group gradually disperses and finally achieves the connected coverage objective.

Acknowledgment

The author Xiaoli Li thanks Dr. Ye Yuan for his valuable discussions on the minimum time consensus problem.

References

[1] H.M.Ammari, S.K.Das, "Centralized and Clustered k-Coverage Protocols for Wireless Sensor Networks," IEEE Transactions on Computers,Vol.61, No.1, pp.118-133, 2012.

[2] C.-F Huang , Y.-C Tseng, H.-L Wu, "Distributed protocols for ensuring both coverage and connectivity of a wireless sensor network," ACM Transactions on Sensor Networks (TOSN), vol.3 no.1, pp.1-24, 2007

[3] Z. Yun, X. Bai, D. Xuan, T.H. Lai, , and W. Jia, "Optimal Deployment Patterns for Full Coverage and k-Connectivity Wireless Sensor Networks," IEEE/ACM Transactions on Networking, vol. 18, no. 3, pp. 934 – 947, 2010.

[4] A. Howard, M. J. Mataric, and G. S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks, Autonomous Robots," Special Issue on Intelligent Embedded Systems, vol. 13, no. 2, pp. 113-126, 2002.

[5] M. Batalin, G. S. Sukhatme, "Spreading out: A local approach to multi-robot coverage," 6th International Conference on Distributed Autonomous Robotic Systems (DSRS02), Fukuoka, Japan, pp. 373-382, 2002.

[6] N. Heo, P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," IEEE Transactions on Systems, Man, and Cybernetics-Part A, 35(1), pp. 78-92, 2005.

[7] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," IEEE Transactions on Robotics and Automation, vol. 20, no. 2, pp. 243-255, 2004.

[8] K. Laventall , and J. Cortés, "Coverage control by multi-robot networks with limited-range anisotropic sensory," International Journal of Control, vol. 82, no.6, pp.1113-1121, 2009.

[9] J. Li , L. Cui , B. Zhang, "Self-deployment by distance and orientation control for mobile sensor networks," 2010 International Conference on Networking, Sensing and Control (ICNSC), pp. 549- 553, 2010.

[10] S. Podur and G. S. Sukhatme, "Constrained coverage for mobile sensor networks," IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, April 26-May 1, pp. 165-172, 2004.

[11] X. L. Li, Y. G. Xi, "Distributed Connected Coverage Control for Groups of Mobile Agents," International Journal of Control, vol. 83, no. 7, pp.1347-1363, 2010.

[12] Y. Kim, and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph laplacian," IEEE Transactions on Automatic Control, vol.51, no.1, pp.116-120, 2006.

[13] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," Automatica, vol.46, pp. 390-396, 2010.

[14] R. Horn, C. Johnson, Matrix analysis. Cambridge Univ Pr, 1990.

[15] S. Sundaram and C. N. Hadjicostis, "Finite-time distributed consensus in graphs with time-invariant topologies," Proceedings of American Control Conference, pp.711-716, 2007.

[16] Y. Yuan, G. Stan, L. Shi, M. Barahona and J. Goncalves. "Decentralised minimal time consensus," Automatic, in press, 2013.