Connecting Robots with Concurrent Exploration of Control and Communications

Zhe Yan*, Xin Wang†, Daegeun Yoon‡ and Dongliang Xie§
Department of ECE, Stony Brook University
NY,Stony Brook, 11790
Email: *zhe.yan@stonybrook.edu, †xwang@ece.sunysb.edu,
‡ykudfor1@gmail.com, §dongliangxie@gmail.com

***Abstract*—Multi-robot systems(MRS) have many applications and the efficient operation of MRS relies on coordination of robots. However, it is difficult to build network connections among randomly distributed robots in the presence of robot movements and weak wireless channels. In this work, we propose to jointly exploit communications and motion control to efficiently establish robot connections . To achieve this goal, we concurrently use MUSIC and particle filter to more accurately and efficiently estimate robot signal directions, built on which signal strength-based potential field is formed to control robot motion to establish and maintain communication links. Our studies based on testbed and simulations demonstrate the effectiveness of our algorithm in networking robots, with much higher number of robots connected compared to peer algorithms.**

I. Introduction

Multi-robot systems (MRS) [1] are widely used in severe environment for applications such as disaster rescuing and national security. Wireless communications are often exploited for robots to coordinate their actions. The efficiency of the system depends on the collaboration among robots. In a rescuing environment, the robots may be dropped from vehicles from the air or on the ground. Although robots are dropped within a region, they may not be close enough to communicate with each other and thus cannot coordinate their actions. In addition, due to the severe fading and unstable channel conditions, existing communication links may break. Although MRS have broad applications, the literature work generally assume there exists a network to connect all robots. The above practical issues are often ignored. The lack of communication among robots will significantly compromise the performance of the multi-robot system.

The goal of this work is to develop a set of schemes that can facilitate robots to establish and maintain communication links, which in turn allows for flexible coordination among robots thus significantly increasing the performance and functionalities of multi-robot systems.

Although robots cannot directly communicate, they may be able to sense the signals from others. The sensing range is generally much larger than the communications range. The range will be even larger if the transmission is carried over low frequency channel and using small transmission rate. Based on this communication feature, we propose the use of dual-channel communication, one beacon channel for transmission of signaling messages to significantly increase the range for robot coordination thus the efficiency of MRS and one data channel to support large data exchange among robots. Specifically, the range increase of transmission on beacon channel will be exploited to establish the communication network among robots.

Despite that the larger transmission range over beacon channel provides a higher chance of establishing robot connections, to the best of our knowledge, there is no existing work that exploits this feature to actively build a communication network. Different from conventional work where the communication links are passively maintained, our proposed scheme will concurrently exploit communications and robot motion control to actively establish communication links and network.

An isolated robot may be able to sense signals from others. If a robot moves towards the signal sources, it will help the robot to connect with other networked nodes. This requires the finding of the direction of the signal and the control of the robot motion to establish the communication links. However, the finding of signal direction is hard when the received signal strength is extremely low, especially in the presence of large noise and channel variation while these are often experienced by MRS working in a harsh environment. Particle filer [2], [3] is a good candidate for signal estimation when the distribution of the received signals is unknown and varies over time, however, it involves a large overhead to update the weights of particles and process the data. On the other hand, MUSIC algorithm [4] is often used to estimate a signal direction in a more stable environment but cannot adapt well with the dynamics. In order to reliably and efficiently form network connections, in this work, we exploit the concurrent use of particle filter and MUSIC algorithm to effectively reduce the signal estimation space and ensure more accurate and low-overhead estimation and tracking of signal directions. Finally, to establish and maintain network connections in the presence of channel randomness, we novelly apply potential-field-based control with wireless signal measurements to adapt the network topology in response to environment changes.

The rest of the paper is organized as follows. We revise the related work in Section II, and introduce our system model in Section III. We present the details of our algorithms in Section IV, and our scheme for network maintenance in Section V. We conclude the work in Section VII.

II. Related Work

To the best of our knowledge, there is no work actively building robot network from the scratch taking into account the severe wireless transmission conditions. In this section, we first review the basic work in MRS field, and then review the work on robot localization and mapping.

A. Multi-robot Systems (MRS)

A multi-robot system can work in various situations, such as disaster rescuing and the exploration of an unknown area. A multi-robot network (WANET) [5] can be implemented as a decentralized ad hoc wireless network. The authors in [6] propose to applysome new concepts such as stochastic gradient to help the robots to deal with some probabilistic events. The focus of the work in [7] is to deploy a multi-robot system with heterogeneous robots for optimal system functions under the nonholonomic constraint. Different from the literature work, the aim of our work is to design algorithms to enable connections of robots in MRS which is essential for flexible and efficient robot coordination and task completion.

B. SLAM(Simultaneous localization and mapping)

Simultaneous localization and mapping (SLAM) [8] is a technique used by robots (or digital machines) to construct a map of an unknown environment or to update a map within a known environment while simultaneously keeping track of the machine's location in the physical environment. In [8], the authors proposed a range-only SLAM with occupancy maps. The method will simultaneously update and check the exploited map of the robot based on some range-only information. To complete a SLAM task, there is no requirement of keeping all robots connected, although a connected network of robots can collaborate to complete the task more efficiently. Making a map of a geographic area is one type of task of MRS but not our focus, although robots in our proposed system can also detect the unknown environment around them. In order to combat severe communication conditions, robots in our system will form a virtual map based on the received signal strength to facilitate network formulation and maintenance, which will also in turn allow for more efficient SLAM if needed.

Rendezvous [9] can be used to help the robots to form networks too. Requiring robots to meet in the same location at a given time, a rendezvous algorithm will significantly compromise the flexibility and performance of MRS. Although our scheme allows a much higher chance of forming network among robots and much larger robot coordination range, our scheme works only when robots can sense signals from others albeit weak. Rendezvous methods can complement our scheme when no signals can be detected by robots.

III. Problem and System Model

We consider a system with a set of small robots randomly distributed in a domain of interest. Each robot is equipped with multiple antennas. The robots may be dropped from an airplane for disaster rescuing or other national defense purpose. As robots may be dropped to different locations, it is difficult for all the robots to be networked and able to communicate. In addition, as a result of severe communication environment the robots often face, some existing communication links may break.

To better exploit the cooperation among robots for more efficient task execution, the aim of this work is to facilitate the robots in the system to form connections. Specifically, our work exploits both control and communications to *establish* a network of robots in the presence of constant robot movement and harsh wireless communications conditions.

In this section, we first introduce the communication model we consider for the system, and then present the challenges and the system architecture.

A. Communication with Dual Ranges

There are two types of communication ranges related to the transmission of a packet: 1) transmission range L_t, inside which a node can receive or overhear the packet transmission, and 2) carrier sensing range L_s, inside which a node can sense the signal but may not be able to decode it correctly.

For two mobile robot nodes R_i and R_j with a distance $d_{i,j}$ in between, if a signal is transmitted by R_j at a power P_j on a frequency channel f, the power received at R_i can be expressed as [10]

$$P_{i,j} = \alpha^2 P_j G_j G_i 10^{x/10}/[(4\pi f)^2 d_{i,j}^{\xi}]. \quad (1)$$

Here, G_i and G_j are the antenna gains of R_i and R_j respectively. The average signal power received by R_i usually decays at a factor $\xi \in [2,4]$ as $d_{i,j}$ increases. The channel condition between R_i and R_j varies due to multi-path fading represented by α^2, where α is a random variable following the Rayleigh or Rician distribution, and shadowing represented by $10^{x/10}$, where x is a shadowing fading factor following the log-normal distribution. With a transmission rate r and channel bandwidth B, the normalized per-bit signal (E_b) to interference plus noise (I_t) ratio (SINR) at R_i is

$$SINR_{i,j} = (E_b/I_t)_{i,j} = (P_{i,j}/r)/[(P_{int\,i} + P_{n\,i})/B], \quad (2)$$

where $P_{int\,i}$ and $P_{n\,i}$ are the average interference and noise power received by R_i respectively. A signal transmitted by R_j can be successfully received and decoded by R_i only when $SINR_{i,j} \geq \gamma_{t\,i}$, where $\gamma_{t\,i}$ is R_i's decoding threshold and depends on R_i's decoding capability. When $\gamma_{s\,i} \leq SINR_{i,j} < \gamma_{t\,i}$, where $\gamma_{s\,i}$ denotes R_i's signal sensing threshold, R_i is inside the signal sensing range L_s but outside the transmission range L_t of R_j. R_i can sense the transmission from R_j but cannot decode the signal. Usually $L_s \geq 2L_t$, so the transmission from a sender can be sensed by a receiver at a much longer distance. Equ. 1 and 2 show that both the transmission range and carrier sensing range depend on the transmission power and rate, and channel bandwidth, conditions and frequency.

For a given transmission power, the transmission range and carrier sensing range increase when the transmission frequency and rate decrease. Based on this observation, we consider the use of *dual radio channels* for multi-robot system (MRS). Each robot will be equipped with two radios, one tuned to the low-frequency *beacon channel* for robot coordination and the other operating on a high-frequency *data channel* for high-rate data transmission. A robot will transmit low-rate beacons which will further increase the beacon range. The beacon channel can have a much longer range than the data channel. If the data rate and frequency of the data channel are m_r times and m_c times those of the beacon channel, the beacon transmission range can be $\sqrt[\xi]{m_r m_c^2}$ times the data transmission range, and the beacon sensing range will be more than $2\sqrt[\xi]{m_r m_c^2}$ times the data transmission range.

In this paper, we focus on the network establishment part. We will exploit the use of beacon channel and robot mobility control to establish new connections among robots and form a communication network, and maintain existing connections under communication channel dynamics. Each robot will periodically send out beacon messages to facilitate the establishment and maintenance of the network. If a robot cannot sense any signals from its team members it can increase its own beacon transmission power to increase the chance for other robots to detect and respond to its request (by moving towards it or sending a stronger signal). For a robot which temporarily loses the connection due to channel fluctuation, power control can be the quickest way to recover the connection. After getting connected or reconnected, the involved robots can adjust their positions and reduce the power to the normal range.

The use of beacon channel here is fundamentally different from the conventional dual-channel schemes [11]–[13] which are applied in an existing network to perform handshaking between nodes to reduce the hidden terminal and exposed terminal problems. The novel use of the beacon channel to establish and maintain connections for multi-robot coordination will largely expand the connected operation range of robots, improve the communication reliability and coverage, enhance the operational flexibility and efficiency of MRS, and save energy for energy-limited MRS.

B. System Model and Challenges

In the MRS system we consider, each robot is equipped with an antenna array which can be used to send and receive signals. In order to enable more flexible MRS function and facilitate the connection of nodes, we consider the coordination among robots through the longer-range beacon channel. In this work, we focus on the design of algorithms to establish the communication links between an isolated robot and the other ones at a given transmission frequency and power. Specifically, our algorithm will efficiently find the direction of sensed signals and exploit robot motion control to drive robots to move towards the signal sources.

An antenna array can be applied to find the direction of signals sensed. In order to connect a robot with others, the robot first measures the signals and then determines the direction to move. This process will be followed as the robot moves step by step towards the signal sources, and the measurements in steps follow a Markov process. As a practical system often has the capability of getting the received signal strength indicator (RSSI), we use $y(t)$ to represent the RSSI measured by a robot at time t. A robot, however, cannot directly measure the direction of signal arrival (DoA) $x(t)$. There is a need to estimate $x(t)$ to guide robot movement for forming the communication connection.

To find out the DoA is not a simple task. The movements of robots and the channel fading will make the signal received weak and vary in strength. Practically, there is no model to accurately capture the relation between the distance and signal strength, which makes the detection of signal direction hard. As the distribution of $x(t)$ is unknown, we will apply Particle filter (PF) [2] [3] to estimate its distribution based on the measured values of $y(t)$. Particle filter can work recursively to effectively track the direction of movements.

In order to better estimate the distribution of a variable of interest, Particle filter uses a large number of particles to help update the posterior probability. In this work, a particle corresponds to an RSSI value at a specific location between the robot and signal sources. Each particle is given a weight to indicate its importance. The set of particles are not all actually measured but are simulated, with the weight of the particles updated upon each RSSI measurement. The estimation of the parameters of the particle filter [14] will help us to determine the distribution of RSSI in two dimensional space, which further helps to better estimate and track DOA to guide the movement of the robot.

Simply using a large number of particles to develop PF would introduce a big computational overhead and takes long time to find the correct moving direction. As an alternative technique, MUSIC [4] can be used by an antenna array to find the direction of signals. However, the direction detected would be very inaccurate if the received signal strength is weak. As robots constantly move, MUSIC algorithm also cannot be used to efficiently predict and track the direction changes.

For more efficient and reliable finding of direction of the signal sources, we will exploit the concurrent use of PF and MUSIC in this work. Specifically, MUSIC will be applied to guide the update of weights of particles, so those falling outside the angular range estimated by MUSIC will be given low PF weights and even be trimmed. This allows PF to quickly converge to the right direction of sources.

In conventional communication networks, it is hard to connect a far-away node and the network is easy to break in an environment with severe channel fading. The motion of robot provides the unique benefit to establish and maintain the network with the exploration of controlled movement. The question is how a robot can be driven to move for network establishment. As the only information we can get is the measured signal strength, we introduce the concept of signal strength-based potential field. Rather than forming the potential field based on location which is difficult to know in practice, the potential field in our system is formed according to the measured strength of the signals. Robots will be driven to move towards a direction that can achieve the target field value. This method is easy to implement than conventional motion control method, and also helps to address the challenge of ensuring reliable communications under severe and varying channel conditions.

The basic network establishment process is summarized in Fig. 1. Based on the input signal strength measurement, the Particle filter and MUSIC algorithm will work interactively to effectively update the weights of PF through recursive Bayesian iteration. The derived signal strength and direction of movement will be applied to form the potential field to control the motion of robots.

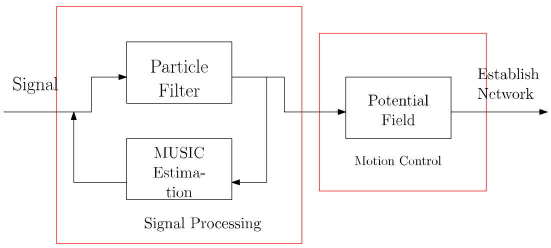

Fig. 1. System Block Diagram

Rather than simply driving robots to move closely, to increase the work efficiency of a MRS system, it is often desirable for the robot group to cover a large area. This would allow MRS to monitor a large domain for better security. In addition, a communication link may break as a result of channel dynamics and robot movement. In order to ensure a large network coverage while maintaining the network connectivity, our system will include a network maintenance scheme with concurrent use of RSSI-based Voronoi diagram and PID control.

IV. Network Establishment

We focus on the design of algorithms to establish network among robots which are beyond the communication range but within the sensing range. If a robot cannot communicate with others but can sense some signals around, it can move towards the signal source(s) until it is within the communication range of other robots. This requires the finding of the direction of the sources and the control of the robot to move along the direction. As each robot is equipped with multiple antennas, they may be applied to find the direction of signals based on the received signal strength. However, the received signals may be weak and unstable as a result of channel fading and robot movement. This makes the direction finding difficult.

In this section, we first present our basic algorithm used to continuously measure the received signal strength and derive the direction of the signal based on particle filter in Section IV-A. We then describe how MUSIC algorithm can be applied to facilitate PF to more efficiently and quickly estimate the signal direction in Section IV-B. Finally, we present our motion control scheme based on the signal strength and signal direction estimated in Section IV-C.

A. Recursive Bayesian Estimation of the Signal Direction

A robot may sense the existence of signals from one robot or a robot group in the distance. Different from the target detection, to establish the communication network, there is no need to differentiate the signals from different robots. An isolated robot only needs to search for other robots based on their aggregate signal received. As the signal measured can be weak and dynamic, for more reliably tracking the received signal and finding the direction to drive robot movement, we exploit the use of Particle filter. The weights of N particles are initialized to $1/N$. Then the Particle filter is formulated through a recursive Bayesian process, which will evolve over time based on consecutive measurement and filter update process, with the variables $x(t)$ and $y(t)$ predicted for the next time instant denoted as $x(t+1)$ and $y(t+1)$. At each time instant, our system will update the Particle filter following the steps below:

1) Getting the current measurement $y(t)$
2) Signal propagation
3) Updating the system parameters according to the Bayesian Rule
4) Weight update and normalization
5) Going to the first step and iterating until a target condition is met
6) Estimating the distribution of $x(t)$

A robot will measure the RSSI value $y(t)$, and then the particle weights will be updated through a signal propagation process following the Bayesian Rule. Then the particles weights are normalized and the ones whose weights are very small will be discarded. Then the input parameters of the Particle filter will be updated. After several updating periods, we will have more knowledge about the distribution of $x(t)$.

In this section, we first introduce the Bayesian model of our problem, and then describe how the distribution of particles evolves over time.

1) Problem Introduction: The signal will become stronger when the robot gets closer to the sources. In each step of movement, the robot would like to find out the direction that could lead to the strongest received signal. In order to achieve this goal, in a step, a robot will estimate the DoA for the next step based on the strength of the signal received in previous steps using recursive Bayesian Estimation [15] [16].

The signal measurement and DoA finding are performed at discrete time instants in our system corresponding to steps of movement. We let x_k to denote the direction estimated at step k, and $\hat{w}_k^{(i)}$ to be the weight set for a particle i. $\hat{w}_k$ denotes the weight vector value at time k. N is the number of particles.

$$\hat{w}_k = [w_k^{(1)}, w_k^{(2)}, \ldots, w_k^{(N)}] \quad (3)$$

The direction x_{k+1} can be estimated based on the previous direction derived x_k and the current weight of the particles w_k, i.e., $x_{k+1} = f(x_k, w_k)$. We can derive x from the measurement y, i.e., $x_k = h_k(y_k, v_k)$, with v_k being a random noise variable. So y_k will be updated after x_k as the new sample is measured. In the next iteration, we can measure new y_{k+1} to estimate the x_{k+1}.

Let X_{k-1} represent the set of previous values of x, including

$x_{k-1}, x_{k-2} \cdots x_0$. We can define the following probability:

$$P\left(y_k|y_{k-1}\right) = \int P\left(y_k|y_{k-1}, w_{k-1}\right) P\left(w_{k-1}|y_{k-1}\right) dw_{k-1}. \tag{4}$$

As w_{k-1} will not be influenced by y_{k-1}, i.e, $P\left(w_{k-1}|y_{k-1}\right) = P\left(w_{k-1}\right)$, our algorithm can work recursively.

$$\begin{aligned} P\left(y_k|Y_{k-1}\right) = \int \delta\left(y_k - f_{k-1}\left(y_{k-1}, w_{k-1}\right)\right) \\ P\left(w_{k-1}\right) dw_{k-1} \end{aligned} \tag{5}$$

According to the Bayesian rule, we have the following formula.

$$P(y_k|x_k) = \frac{P(x_k|y_k)P(y_k|X_{k-1})}{P(x_k|X_{k-1})} \tag{6}$$

Here, the denominator can be calculated by:

$$P(x_k|y_k) = \int \delta(x_k - h_k(y_k, v_k))p(v_k)dv_k \tag{7}$$

The likelihood for x and y can be calculated using statistic distribution such as Gaussian distribution [2] in communication system. Here δ is the Dirac delta function.

2) Particle Propagation: For a finite set of particles, the performance of the algorithm is dependent on the choice of the proposed distribution:

$$\pi(x_k|X_{k-1}, Y_k) \tag{8}$$

$$P(x_k|y_k) = \sum w_k^{(L)} \delta(x_k - w_k^{(L)}) \tag{9}$$

The optimal proposal distribution in equation (9) is given as the sum of delta function at each particle.

$$\pi(x_k|X_{k-1}, Y_k) = p(x_k|x_{k-1}, y_k). \tag{10}$$

However, the transition probability of the prior distribution is often used as the importance function for weight update, since it is easier to draw particles (or samples) and perform subsequent importance weight calculations:

$$\pi(x_k|X_{k-1}, Y_k) = p(x_k|x_{k-1}). \tag{11}$$

Sequential Importance Resampling (SIR) filters with transition prior probability distribution as importance function are commonly known as bootstrap filter and condensation algorithm. Our weight function can be calculated based on the Gaussian distribution around each particle. So our particle filter is the implementation of Gaussian particle filter.

$$w_k^{(L)} = P(y_k^{(L)}|x_k^{(L)}) \tag{12}$$

The w_k can be updated and normalized in the particle filter process. The most important equation to update is the equation 17 and equation 18.

To perform particle filter, we first draw samples from the proposal distribution. We then choose some sample point from the 2D space and denote the x value at the $L-th$ particle or sample as $x_k^{(L)}$. This particle is generated randomly at the beginning with a direction value x and will be updated when we resample the signal.

$$x_k^{(L)} \sim \pi(x_k|x_{0:k-1}^{(L)}, y_{0:k}) \tag{13}$$

Here L is the Lth particle and y is the measured signal strength. After we draw Lth sample, we update the weights up to a normalizing constant:

$$\hat{w}_k^{(L)} = w_{k-1}^{(L)} \frac{p(y_k|x_k^{(L)})p(x_k^{(L)}|x_{k-1}^{(L)})}{\pi(x_k^{(L)}|x_{0:k-1}^{(L)}, y_{0:k})} \tag{14}$$

Note that we can use the transition prior probability distribution as the importance function. It will help us simplify the equation **(15)**.

$$\pi(x_k^{(L)}|x_{0:k-1}^{(L)}, y_{0:k}) = p(x_k^{(L)}|x_{k-1}^{(L)}) \tag{15}$$

The equation (16) can be simplified to the following format. In this equation, the weight will be updated according to the conditional probability of signal strength y.

$$\hat{w}_k^{(L)} = w_{k-1}^{(L)} p(y_k|x_k^{(L)}) \tag{16}$$

3) Implementation of Particle Filter: With a simpler format of w, we can develop our algorithm for the robot system. We pick N particles in the region of interest. Instead of distributing the particles into a large region which leads to inaccurate particle representation, as we will introduce in Section IV-B, the region we consider will be restricted and lie around the coarse direction found with the MUSIC algorithm. The weights of all particles are initialized with equal weight at $1/N$ [17]. Then the value $w_k^{(L)}$ corresponding to the weight of the L_{th} particle at the time k can be calculated recursively as follows:

$$w_k^{(L)} = w_{k-1}^{(L)} p(y_k|x_k^{(L)}) \tag{17}$$

As the update is made only based on the states of the recent two steps, the storage cost is low. The probability $p(y_k|x_k^{(L)})$ can be calculated according to the function $\hat{y} = h^{-1}(x)$. Then the probability will be estimated according to the Gaussian distribution.

The weights can be normalized as

$$w_k^{(L)} = \frac{\hat{w}_k^{(L)}}{\sum_{J=1}^{P} \hat{w}_k^{(J)}}. \tag{18}$$

As time goes on and more samples are taken, some of the weights will be very small and approach zero. In order to improve the efficiency of Particle filter, we will determine the effective number of particles and the updated new particles in the Particle filter before we update x and take more samples.

The effective number of particles N_{eff} can be calculated as

$$\hat{N}_{eff} = \frac{1}{\sum_{L=1}^{N} \left(w_k^{(L)}\right)^2} \tag{19}$$

If the effective number of particles is less than a given threshold, i.e., $\hat{N}_{eff} < N_{thr}$, we perform re-sampling by drawing P particles from the current particle set with probabilities proportional to their weights and replacing the current particle set with this new one. We then set $w_k^{(L)} = 1/N$.

With weights of particles updated, we can update the direction of signal sources as follows:

$$x_{k+1} = f(x_k, w_k) \tag{20}$$

We can repeat the previous procedures until the robot can communicate with the robot(s) it receives signals from.

B. Regulation of Particle Region for Efficient State Estimation

The cost of Particle filter increases quickly with the number of particles used. As discussed in Section III-B, we can apply MUSIC algorithm to coarsely estimate the arriving angle of the received signal so that the particle filter can work in a restricted region (Fig. 2) to significantly reduce the computational expense caused by the large number of particles.

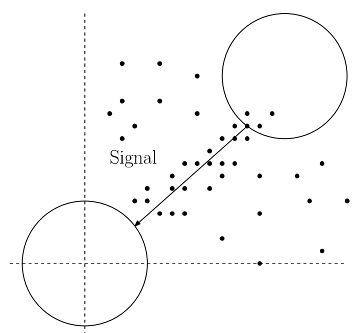

Fig. 2. Particle Filter Implementation Region

MUSIC detects angular frequencies in a signal by performing an eigen decomposition on the autocorrelation matrix of the data vector of samples taken from the received signal. We use variable y to denote the received signal strength, s to denote the original signal strength without noise, and N the additive noise. A is the matrix serving as the transfer function from original signal s and signal strength y.

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{s} + \boldsymbol{N} \tag{21}$$

The autocorrelation matrix of signal $\boldsymbol{y}$, $\boldsymbol{R_y}$, is calculated as

$$\boldsymbol{R_y} = E[\boldsymbol{y}\boldsymbol{y}^{\boldsymbol{H}}]\boldsymbol{A}\boldsymbol{R}_{ss}\boldsymbol{A}^{\boldsymbol{H}} + \boldsymbol{\sigma}^2\boldsymbol{I} \tag{22}$$

where $\boldsymbol{R_s}$ is defined as

$$\boldsymbol{R_s} = E[\boldsymbol{s}\boldsymbol{s}^{\boldsymbol{H}}]. \tag{23}$$

Then we can get the eigen values $\boldsymbol{V_n}$ from the matrix $\boldsymbol{A}^{\boldsymbol{H}}$. The power function corresponding to angle θ can be found as

$$\boldsymbol{P(\theta)} = \frac{\boldsymbol{a}^{\boldsymbol{H}}\boldsymbol{(\theta)}\boldsymbol{a(\theta)}}{\boldsymbol{a}^{\boldsymbol{H}}\boldsymbol{(\theta)}\boldsymbol{V_n}\boldsymbol{V_n^H}\boldsymbol{a(\theta)}} \tag{24}$$

where $a(\theta)$ is the column vector in Matrix $\boldsymbol{A}$. The θ corresponding to the peak value of the power function $\boldsymbol{P(\theta)}$ is the estimated arriving signal direction.

In order to reduce the complexity of Particle filter, we select the particles within the angular range $[\theta - \pi/4, \theta + \pi/4]$ in our scheme. This eliminates the unnecessary particles, which will not only reduce the computational cost and time, but also increase the accuracy of the angle estimation.

C. Motion Control Based on Potential Field

After processing the received signals, we can extract two useful data values: the received signal strength y and the estimated direction of arrival x. The robot can communicate with others only if the signal strength y is above a signal decoding threshold ϵ. The value x informs the robot the direction to move in order to connects with other robots it receives signals from. In this section, we will introduce the motion control algorithm that can drive the robot movement according to the x and y values evaluated from the previous steps.

The motion control algorithm should guide the robot how much to move, and which direction to move. As the estimated signal has both the amplitude and direction, it is very natural to connect it to the physical field, i.e., the potential field, which shares the same feature and has some well defined theory to follow. The potential field is formed due to the difference between two values. In our work, the field strength is related to the relative direction and signal strength. Different from a force which is associated with a direction, the potential field does not have direction and all fields can be added. The potential field has been used in the literature to guide movement based on a known target location, where the location to be reached is an attractive pole for the end effect while obstacles form repulsive surfaces for the manipulator parts.

Different from the literature work, the positions of the remote robots are unknown. Specifically, the motion control needs to drive a robot to move towards the other robots until its received signal strength is above the signal decoding threshold, and we need an attractive field to generate the force to drive robot movement. On the other hand, it is beneficial for robots to be as far as possible so they can cover a large area for better surveillance. So a repulsive field is needed when robots get too close. Further, a repulsive field can also prevent a robot from hitting an obstacle. In this work, we focus on the fields that help establish and maintain the communication links.

We define a potential function U to capture all factors that may influence our estimation of the signal strength and field value: $\nabla U = [\frac{\partial U}{\partial q_1}, \ldots, \frac{\partial U}{\partial q_n}]^T$. Without loss of generality, we consider a variable space R^2, while our algorithm can also work in the space of R^3 space. We consider a signal field which is a function of the received signal strength y, and is composed of both the attractive field $U_{att}(y)$ and the repulsive field $U_{rep}(y)$

$$U(y) = U_{att}(y) + U_{rep}(y) \tag{25}$$

where

$$U_{att}(y) = \frac{1}{2}k_{att}f(y)^2 - k_{att}f(y) \tag{26}$$

$$U_{rep}(y) = \begin{cases} \frac{1}{2}k_{rep}(\frac{1}{f(y)} - \frac{1}{f(y_o)})^2 & \text{if } f(x) \leq \epsilon \\ 0 & \text{if } f(x) \geq \epsilon \end{cases} \tag{27}$$

k_{att} and k_{rep} represent the scaling factors of attractive field and repulsive field, respectively. $f(x)$ represents the measured RSSI strength, while ϵ is the target RSSI threshold. This filed is just a component when we implement the field. The total field will be the sum of $U_{att}(y)$ and $U_{rep}(y)$.

The forces, F_{att} and F_{rep} can be obtained by negative gradient of potential functions, U_{att} and U_{res}.

$$-\bigtriangledown U_{att} = F_{att} = -k_{att}f(y) + k_{att} \tag{28}$$

$$-\bigtriangledown U_{rep}(x) = F_{req} = \begin{cases} \frac{1}{2}k_{rep}(\frac{1}{f(y)} - \frac{1}{f_o})\frac{1}{f^2(y)} \\ \qquad \text{if } f(x) \leq \epsilon \\ 0 \\ \qquad \text{if } f(x) \geq \epsilon \end{cases} \tag{29}$$

As a robot node may have a few neighbors, there will be several different attractive and repulsive fields around:

$$U(x) = \sum_i U_{att}(y) + \sum_j U_{rep}(y) \tag{30}$$

Considering the limited computation ability of robots and maximum number of robots in our design, the repulsive field is generated with sum of at least tree independent field we defined earlier. Corresponding distribution is shown in Fig 4

1) Attractive Field: The attractive field which drives the robot to move closer to the signal source is defined as follows

$$U_{att}(x) = \begin{cases} \frac{1}{2}k_{att}f^2(y) - k_{att}f(y) & \text{if } f(y) \leq \epsilon \\ \frac{1}{2}k_{att}f_o^2 & \text{if } f(y) \geq \epsilon \end{cases} \tag{31}$$

The threshold ϵ has been set for the attractive force to make the function bounded in a reasonable range. When the distance is larger than ϵ, the gradient will be 0 in that interval. The force generated by the attractive field will only influence in the range of ϵ. The attractive field will attract the neighboring robots when the gets to far from the objective robot. Repulsive one will make sure they don't bump with each other and maximize the coverage.

2) Repulsive Field: As an opposite field of attractive field, the repulsive one will prevent the robot from getting too close to each other. So here comes our repulsive definition.

$$U_{rep}(y) = \begin{cases} C_{max} & \text{if } 0 \leq f(y) \leq f_{max} \\ \frac{1}{2}k_{rep}(\frac{1}{f(x)} - \frac{1}{f_o})^2 & \text{if } f_{max} \leq f(y) \leq \epsilon \\ 0 & \text{if } f(y) \geq \epsilon \end{cases} \tag{32}$$

The f_{max} is the maximum possible value of the received signal strength. Since the gradient in innermost part is zero, it will conflict with the definition of repulsive field. So we will define the repulsive force separately.

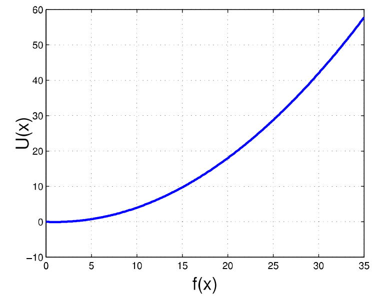

Fig. 3. Attractive Field

$$F_{rep}(y) = \begin{cases} F_{max} & \text{if } 0 \leq f(y) \leq f_{max} \\ \frac{1}{2}k_{rep}(\frac{1}{f(y)} - \frac{1}{\epsilon})\frac{1}{f^2(y)} \bigtriangledown f(y) & \text{if } f_{max} \leq f(y) \leq \epsilon \\ 0 & \text{if } f(x) \geq \epsilon \end{cases} \tag{33}$$

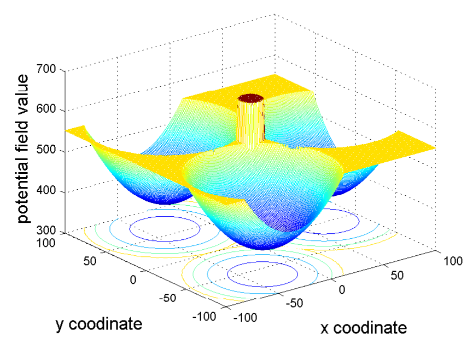

Fig. 4. 3D plot of Potential Field for single mobile robot

With our definition of the potential field, if a single robot is located in the center of the field, we will have the field distribution as shown in Fig 5. The repulsive force in the center not only prevents bumping field, but also keeps the signal strength received by a robot to be around the target SNR value.

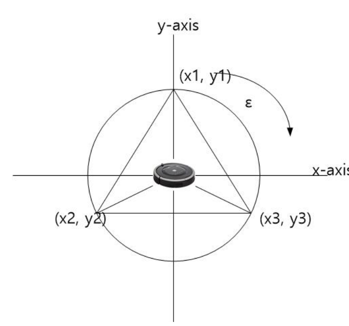

Fig. 5. 2D plot for rotating potential field

Using (0,0) as the reference location for each robot, 3 center points of attractive field will be selected. The parameter ϵ can be a smaller value compared to the field value at communication range. So it works like a threshold. When a robot is selected as the attractive center, the three points should satisfy two conditions as follows:

$$\begin{aligned} \epsilon &= radius \\ |f(x_i, y_i)| &\le f_{max} \end{aligned} \tag{34}$$

where the (x_i, y_i) is the center of a mobile robot. In addition, attractive points should not be too close to other robots. In order to have a larger coverage area, we can set the field to rotate in the updating period, so that the robot will not be trapped to a local minimum point. The rotation matrix for two dimensional graph can be defined as in equation 11, where θ is the rotation angle and the relationship can be denoted as the rotation matrix R.

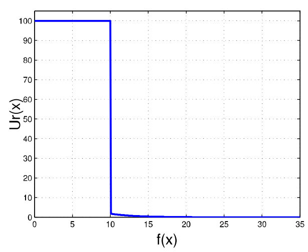

Fig. 6. Repulsive Field

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{35}$$

This rotating matrix will allow a robot to rotate the direction of the field according to the current received signal strength and the estimated DOA x. The rotation operation helps to adjust the potential field slightly according to the change in the direction of arrival.

V. Maintain Communication

It is important to maintain the network connections with constant robot movement and severe link conditions. Instead of letting all robots to gather within a small area which compromises the function of MRS [18], it is important to ensure a maximum coverage of MRS while maintaining the connectivity among robots.

Voronoi diagram is widely used in many topology control problems. In this section, we integrate use of Voronoi diagram with RSSI and DoA found in the previous steps introduced to maximize the coverage of the multi-robot system.

To maintain the RSSI between neighboring robots around the target value, we further apply PID control to work with Voronoi diagram. The input of the controller is the RSSI received from other robots. Defining $u(t)$ as the controller output, the formula of the PID algorithm is:

$$\mathrm{u}(t) = \mathrm{MV}(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{d}{dt} e(t) \tag{36}$$

In Eq. 36, the proportional, integral, and derivative terms are summed to calculate the output of the PID controller. The

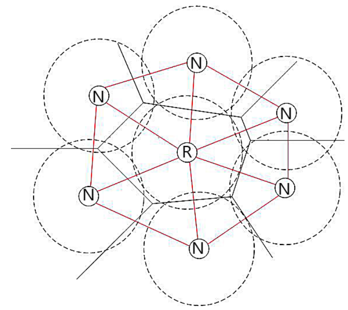

Fig. 7. Communication range after moving to the centroid of Voronoi polygon

model will keep working until the error between the measured RSSI and the target RSSI value approaches zero.

VI. Evaluation

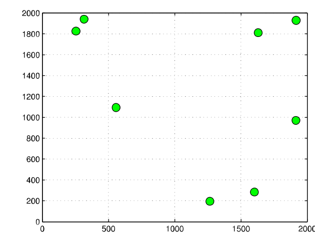

Fig. 8. Initial distribution of robots.

To evaluate the performance of our design, we build a testbed with eight Roomba robots each equipped with four antennas. Eight robots are randomly distributed within a field of 500*500 meters. We measure the signal strength of robots at different locations, and then feed the measured signal strength data into matlab simulator. The distribution of robots is shown in fig 8. [1]

A. Direction Finding with Music and Particle Filter

Using our improved version of MUSIC algorithm, we can detect the arrival signal angle according to the strength of the signals received by the antenna array. In Fig. 9, the direction of signal corresponds to the peak value of the MUSIC power, and we can see four typical signals are detected to come from different directions. The output is the angle in radius.

B. Maximum Number of Robots in the Largest Group

Our algorithm focuses on getting more robots connected in the communication range and maintaining the existing links. In fig 10, concurrent use of MUSIC and Particle filter allows the highest average number of robots to be connected in the largest connected group. The number doubles that using MUSIC algorithm alone, and triples that using the random movement or rendezvous scheme. This demonstrates the effectiveness of our algorithm in improving the accuracy of DoA estimation, thus ensuring more efficient network formulation.

[1] In the fig 8 11 12, the x and y legends represents the x coordinate and y coordinate in 2D space.

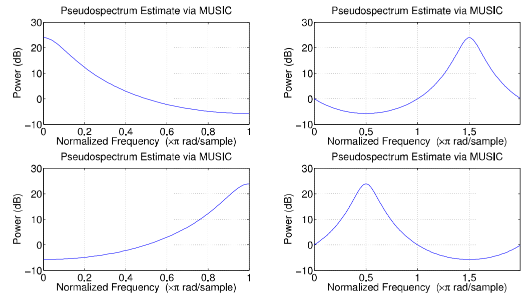

Fig. 9. Direction estimation with MUSIC and Particle filter.

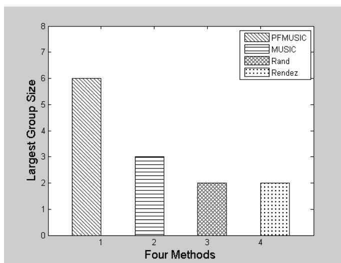

Fig. 10. Maximum number of robots in the largest group

C. Algorithm Evaluation

To evaluate the performance of our algorithm in networking a robot group, we compare our algorithm with the MUSIC only algorithm in fig 11 and fig 12. The blue line denotes the robot which is within the communication range. Our algorithm gets more robots connected with its use of particle filter for more accurate estimation of the direction of arrival of signals in the presence of noise and dynamic wireless channel conditions.

VII. Conclusion

MRS have broad applications. The efficiency and functionalities of MRS rely on flexible robot coordination, which in turn depends on reliable communications between robot nodes. However, robots dropped into the fields may not be connected initially and existing connections may break as a result of robot movement and severe wireless communication environment. There is a lack of study on efficient connection formulation among a randomly distributed group of robots in the presence of weak and random wireless channel conditions and mobility. We propose to pro-actively build and maintain network connections with interactive and integrated use of communications and motion control techniques. Specifically, we exploit concurrent use of MUSIC and Particle filter to more accurately estimate the direction of sensed robot signals, and apply signal-based potential field to further guide robot movement. Our performance studies demonstrate that our proposed methods are very effective in connecting robots into a network.

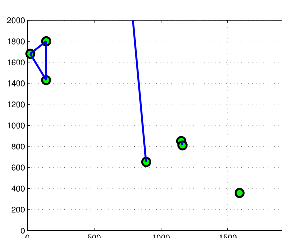

Fig. 11. Network topology with Music algorithm only

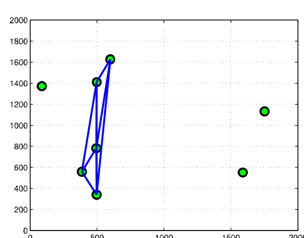

Fig. 12. Network topology with our algorithm

References

[1] Brian Gerkey, Richard T Vaughan, and Andrew Howard. The player/stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings of the 11th international conference on advanced robotics*, volume 1, pages 317–323, 2003.

[2] Jayesh H. Kotecha and P.M. Djuric. Gaussian particle filtering. *Signal Processing, IEEE Transactions on*, 51(10):2592–2601, Oct 2003.

[3] Jayesh H. Kotecha and P.M. Djuric. Gaussian sum particle filtering. *Signal Processing, IEEE Transactions on*, 51(10):2602–2612, Oct 2003.

[4] Yameng Jiao, Jianguo Huang, and Yunshan Hou. Multidimensional music doa estimation using ant colony optimization algorithm. In *Signal Processing (ICSP), 2010 IEEE 10th International Conference on*, pages 291–294, Oct 2010.

[5] S. Fujiwara, T. Ohta, and Y. Kakuda. An inter-domain routing for heterogeneous mobile ad hoc networks using packet conversion and address sharing. In *Distributed Computing Systems Workshops (ICDCSW), 2012 32nd International Conference on*, pages 349–355, June 2012.

[6] Yanjiang Huang, R. Chiba, T. Arai, T. Ueyama, and J. Ota. Integrated design of multi-robot system for pick-and-place tasks. In *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pages 970–975, Dec 2013.

[7] T. Zhang, Md. Hasanuzzaman, V. Ampornaramveth, and H. Ueno. Construction of heterogeneous multi-robot system based on knowledge model. In *Robotics and Biomimetics (ROBIO). 2005 IEEE International Conference on*, pages 274–279, 2005.

[8] L. Jaulin. Range-only slam with occupancy maps: A set-membership approach. *Robotics, IEEE Transactions on*, 27(5):1004–1010, Oct 2011.

[9] Li Xiong and Huang Jianguo. Bayesian high resolution doa estimator based on importance sampling. In *Oceans 2005 - Europe*, volume 1, pages 611–615 Vol. 1, 2005.

[10] M. Schwartz. *Mobile Wireless Communications*. Cambridge University Press, Cambridge, UK, 2005.

[11] Z. J. Haas and J. Deng. Dual busy tone multiple access (DBTMA)-a multiple access control scheme for ad hoc networks. *IEEE Trans. on Communications*, 50(6):975–985, June 2002.

[12] S. Wu, Y. Tseng, and J. Sheu. Intelligent medium access for mobile ad hoc networks with busytones and power control. *IEEE Journal on Selected Areas in Communications*, 18(9):1647–1657, September 2000.

[13] H. Zhai, J. Wang, and Y. Fang. Ducha: A new dual-channel mac protocol for multihop ad hoc networks. *IEEE Transactions on Wireless Communications*, 5(11):3224–3233, November 2006.

[14] P.M. Djuric, M. Vemula, and M.F. Bugallo. Target tracking by particle filtering in binary sensor networks. *Signal Processing, IEEE Transactions on*, 56(6):2229–2238, June 2008.

[15] M. Bocquel, H. Driessen, and A Bagchi. Multitarget particle filter addressing ambiguous radar data in tbd. In *Radar Conference (RADAR), 2012 IEEE*, pages 0575–0580, May 2012.

[16] E.-S. Abdoul-Moaty, T.R. Abdoul-Shahid, A El-Din Sayed Hafez, and M. Abd-El-Latif. A particle filter for mutistatic radar tracking. In *Aerospace Conference, 2014 IEEE*, pages 1–5, March 2014.

[17] F.A. Ruslan, R. Adnan, A.M. Samad, and Z.M. Zain. Parameters effect in sampling importance resampling (sir) particle filter prediction and tracking of flood water level performance. In *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*, pages 868–872, Oct 2012.

[18] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, volume 2, pages 500–505, Mar 1985.