2016 IEEE First International Conference on Internet-of-Things Design and Implementation

A neural network and IoT based scheme for performance assessment in Internet of Robotic Things

Cristanel Razafimandimby
Inria Lille - Nord Europe
France
Email: jean.razafimandimby_anjalalaina@inria.fr

Valeria Loscri
Inria Lille - Nord Europe
France
Email: valeria.loscri@inria.fr

Anna Maria Vegni
University of Roma Tre
Italy
Email: annamaria.vegni@uniroma3.it

***Abstract*—Internet of Robotic Things (IoRT) is a new concept introduced for the first time by ABI Research. Unlike the Internet of Things (IoT), IoRT provides an active sensorization and is considered as the new evolution of IoT. This new concept will bring new opportunities and challenges, while providing new business ideas for IoT and robotics' entrepreneurs. In this paper, we will focus particularly on two issues: (i) connectivity maintenance among multiple IoRT robots, and (ii) their collective coverage. We will propose (i) IoRT-based, and (ii) a neural network control scheme to efficiently maintain the global connectivity among multiple mobile robots to a desired quality-of-service (QoS) level. The proposed approaches will try to find a trade-off between collective coverage and communication quality. The IoT-based approach is based on the computation of the algebraic connectivity and the use of virtual force algorithm. The neural network controller, in turn, is completely distributed and mimics perfectly the IoT-based approach. Results show that our approaches are efficient, in terms of convergence, connectivity, and energy consumption.**

Keywords*—*IoT-based, connectivity maintenance, coverage, IoRT, neural network.

I. Introduction

Nowadays, Internet of Things (IoT) technology begins to take an important place in economic systems and in society daily life [8]. It has got a large success in several application areas, ranging from smart city applications [2] [16], smart grid [17], etc. However, sensors constituting IoT paradigm are only passive so far. Later, adding an active role for sensors will be needed, in order to optimize the systems where they are present.

Robotic systems match very well to this new need, since robots can sense and interact with their environment. That is why ABI Research introduced a new concept called Internet of Robotic Things (IoRT) [18]. IoRT was defined as an intelligent set of devices that can monitor events, fuse sensor data from a variety of sources, use local and distributed intelligence to determine a best course of action, and then act to control or manipulate objects in the physical world [18]. This new concept is expected to be the evolution of IoT and robotics.

In various IoRT applications (e.g: smart agriculture, smart environment monitoring, smart exploration, smart disaster rescue, etc), the use of mobile robots' teams brings many advantages over one powerful IoRT robot. As a matter of fact, a team of robots can accomplish tasks more efficiently, faster and more reliable than a single robot [5][14][3]. To carry out cooperative tasks, IoRT team members need to communicate with each other, often via a wireless link (i.e. Wifi, Bluetooth). Maintaining communication among multiple mobile IoRT robots is therefore a crucial issue.

Many approaches have been designed to maintain the connectivity of multi-robot and multi-agent systems. These approaches can be classified into two groups i.e. (i) local and (ii) global connectivity maintenance. With the local connectivity maintenance, the initial set of edges which define the graph connectivity must be always preserved over time. Unlike local connectivity maintenance, global connectivity maintenance allows suppression and creation of some edges, as long as the overall connectivity of the graph is conserved. Different examples and results related to these approaches can be found in [11][20][4][21][12][6].

In Multi-Robot Systems (MRS), global connectivity maintenance is often used since the local connectivity maintenance presents some restrictions. Besides connectivity maintenance, the major problematic in global connectivity maintenance approaches is how to maximize the network connectivity. Maximizing the connectivity is important to ensure reliable communication between any pair of IoRT robots. Many works based on graph theory were proposed in the literature to face this problematic. These works are extensively used in multi-agent systems and are based on the maximization of the algebraic connectivity. In this paper, we try to migrate some of these ideas in IoRT applications. Precisely, we will use the graph connectivity metric to maintain the global connectivity of IoRT robots' team, when they are in mobility.

This paper addresses also the coverage issue. In general, coverage issue aims to determine how well the sensing field is monitored or tracked by sensors. In literature, Virtual Force Algorithm (VFA) was widely used to formulate this problem [22]. However, these methods have limitations since there are situations that do not allow the systems to converge in a stable state [7]. We will present a new solution to this problem later in this paper.

To summarize, in this paper we address two problems i.e. (i) connectivity maintenance, and (ii) collective coverage. However, it is interesting to mention that maximizing these two parameters simultaneously is difficult (if not impossible).

978-1-4673-9948-7/16 $31.00 © 2016 IEEE
DOI 10.1109/IoTDI.2015.10

Maximize the collective coverage may lead poor communication quality and conversely (i.e. a very good communication may lead poor coverage). It follows that our goal is to capture the trade-off between collective coverage and communication quality. Leveraging on the above motivations, in this paper, we propose two motion control strategies which maintain global connectivity between IoRT robots to a desired QoS level. The first approach is an IoT-based while the second in a distributed trained neural network controller.

The main contribution of this paper is the design of two approaches that have the following properties:

- The first approach is IoT-based, IoRT robots works with a central object which has high computation capability for network connectivity computing/monitoring and for the robot motion decision;
- Our both approaches converge to the desired communication quality level;
- Connectivity between any pair of IoRT robots is kept all along the deployment procedure;
- Our approaches use a distributed virtual force algorithm when the access to the central object is available and a distributed trained neural network controller otherwise. Both strategies are computed locally and based only on the local neighborhood information.

The rest of this paper is organized as follows. Section 2 provides some backgrounds which include information on algebraic graph theory and neural networks. Section 3 describes the IoT-based approach, while Section 4 details the neural network approach. Section 5 provides the simulation results. Finally, Section 6 is dedicated to the conclusions.

II. BACKGROUNDS

A. Graph representation and Eigenvalues

Multi-Robot Systems (MRS) can be represented by a graph $G(V, E)$ where V is the set of vertices representing each IoRT robot and $E \subseteq V^2$ is the set of edges. E can be be defined as : $E = \big\{(i,j) \in V^2 \mid i \neq j \wedge d(i,j) \leq R\big\}$, where $d(i,j)$ is the euclidean distance between i-th and j-th IoRT robots and R is the communication range. Following the above definition, let N_i be the one-hop neighborhood of the i–th IoRT robot. Thus, N_i is the set of IoRT robots which can exchange information with IoRT robot i. N_i can be defined as follow :

$$N_i = \{j \in V \mid d(i,j) \leq R\} \tag{1}$$

Definition 1. An undireted graph G is connected if there exists a path between each pair of vertices.

In this paper, the graph G may evolve over time due to the IoRT robots motion but has to be always connected.

Definition 2. Let define a matrix $A \in R^{n\times n}$. The scalar λ is an eigenvalue of A if there exists a non-zero vector w such that

$$A.w = \lambda.w \tag{2}$$

The vector w is called eigenvector of A corresponding to λ.

B. Laplacian matrix and algebraic connectivity

Given a undirected graph G, its Laplacian matrix L is defined as:

$$L(G) = \Psi(G) - A(G) \tag{3}$$

where:

- $A(G)$ is the weighted adjacency matrix of graph G whose entries A_{ij} is defined as in [11]:

$$A_{ij}= \begin{cases} 1 & d(i,j) < D_{th} \\ e^{\frac{-5(d(i,j)-D_{th})}{R-D_{th}}} & D_{th} \leq d(i,j) \leq R \\ 0 & d(i,j) > R \end{cases} \tag{4}$$

- $\Psi(G)$ is a diagonal matrix such as the components $\Psi_i = \sum_{i=1}^{n} A_{ij}$ along the diagonal
- D_{th} is the desired distance between each pair of IoRT robots.

The Laplacian matrix $L(G)$ holds some interesting properties:

1) Let **1** be the column vector of all ones. Then, L**1** = 0.
2) Let λ_i, $i = 1, ..., n$ the eigenvalues of the Laplacian matrix $L(G)$.

- The eigenvalues of $L(G)$ can be ordered such that

$$0 = \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq ... \leq \lambda_n \tag{5}$$

- $\lambda_2 > 0$ if and if only the graph G is connected. The second-smallest eigenvalue λ_2 is called also algebraic connectivity of the graph G. The value of λ_2 indicates how weel connected the graph is.

Further details on graph theory and proof can be found in [9].

C. Artificial Neural Networks

Artificial Neural Networks (ANN) was inspired by the human brain and was designed as a computational model to solve specific problems [1]. It's architecture is defined by (i) a basic processing elements called artificial neurons, and (ii) the way in which they are interconnected. The output value of a neuron is given by:

$$output = f(\sum_i w_i x_i + b) = f(W^T X + b) \tag{6}$$

where

- x_i: the inputs
- w_i: connections' weights between x_i and the neuron
- W: weights' vector
- X: inputs' vector
- b: the bias
- f: the activation function

The basic architecture of ANN contains three neuron layers: input layer, hidden layer and output layer. In this case, the outputs of one layer become the inputs of next layer [1]. A typical artificial neuron and a basic ANN are illustrated in Figure 1.

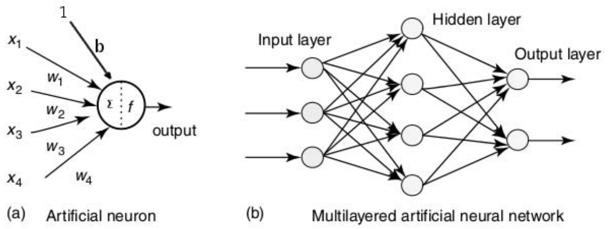


Figure 1: Architecture of an artificial neuron and a multilayered neural network

A key element of an artificial neural network is its ability to learn. This meant that ANN has to learn from a data set in order to match the inputs to the desired output. During the learning process, weights and biases are adjusted till the desired output will be reached. There are several learning algorithm but in this paper we will use the backpropagation algorithm [19].

III. IoT-BASED APPROACH

As we saw in the previous section, an undirected graph is connected if and only if $\lambda_2 > 0$. Therefore, any strategy which maintiens λ_2 at positive values guarantees global connectivity among multi-robot systems.

In this section, we present an IoT-based approach which is capable of maintaining desired wireless communication coverage among neighboring robots.

The proposed approach uses a Central Object (CO) with high computation capability to compute and monitor the connectivity of the overall multi-robot system. We assume that each IoRT robot knows its own position by using GPS or other localisation system. Beacon messages are used to allow IoRT robots exchange their positions with their one-hop neighbors. Each IoRT robot in the system applies a distributed virtual force algorithm to control its movement. This computation is only based on the local neighborhood information. In order to keep the desired distance and hence the desired connectivity quality with its neighbor, the $i-th$ IoRT robot should move away from the IoRT robot $j \in N_i$ if $d(i,j) < D_{th}$ and should move close if $d(i,j) > D_{th}$. D_{th} is the desired distance between each pair of IoRT robots. This simple control law generates a vector position $\overrightarrow{P_{ij}}$ such that the $i-th$ IoRT robot keeps the line of sight of the IoRT robot j. $\overrightarrow{P_{ij}}$ is defined as :

$$\overrightarrow{P_{ij}} = \begin{cases} (0.1 \times k \times error, \theta_{ij}) & if\, d(i,j) > D_{th}\, and\, error > \epsilon \\ (k \times error, \theta_{ji}) & if\, d(i,j) < D_{th}\, and\, error > \epsilon \end{cases} \tag{7}$$

where :

- $error =\mid d(i,j) - D_{th} \mid \quad (7)$
- θ_{ij} is the orientation of the line segment from robots i to j;
- k is the damping coefficient
- ϵ is a lower bound of $error$. It will be used in order to avoid useless small movements.

In order to overcome the problem in the original VFA, we set the attractive coefficient w_a to one tenth of repulsive coefficient k ($w_a = 0.1 \times k$).

When the $i-th$ IoRT robot has more than one neighbor, its new position is calculated as the summation of the position decisions with respect to all the neighbors :

$$\overrightarrow{P_i} = \sum_{j \in N_i} \overrightarrow{P_{ij}} \tag{8}$$

After calculating their new positions, each IoRT robot sends the computed position to the Central Object (CO). Then, CO computes the algebraic connectivity λ_2 of the IoRT robots network according to the formula (3). The central object CO allows each IoRT robot to move to their new positions if and only if $\lambda_2 > 0$. This guarantees that global connectivity is always kept all along the deployment procedure. It is important to note that IoRT robots and CO can communicate to each other through an IoT platform.

The following algorithm summarizes our approach:

Algorithm 1 IoT-based (runs every t units of time)

```
Phase I : Neighbor Discovery
  MyNeighbor = FindNeighbor(RobotId)
Phase II : Compute the position $\overrightarrow{P_{ij}}$ between two robots
  Compute $\overrightarrow{P_{ij}}$ using Formula (7)
Phase III : Compute the new position $\overrightarrow{P_i}$
  Compute $\overrightarrow{P_i}$ using Formula (8)

Phase IV : Compute algebraic connectivity
  Compute $\lambda_2$ of the dynamic Laplacian matrix $L(G)$
Phase V : Deployment
  if $\lambda_2 > 0$ then
    move to $\overrightarrow{P_i}$
  else
    do not move
```

IV. ANN-BASED APPROACH

As we mentioned before, the connection to the central object is not always possible. For example, a rescue operation may be difficult after a disaster when the access to the central node is not available. An approach which easily adapts to any type of situation and environment is more than necessary.

To meet this need, we provide an ANN-based technique which can perfectly mimic the behaviors of IoT-based approach. The ANN-based approach is completely distributed and is trained from a set of data. The data set is obtained by using the IoT-based approach and we use backpropagation algorithm to train ANN. The trained ANN is constituted by 2 input units and 1 output unit. The 2 input units are $d(i,j)$

and θ_{ij}, while the output is $\overrightarrow{P_{ij}}$. Therefore, the trained ANN is executed locally for each IoRT robot to control its movement according to its neighbor's distance $d(i,j)$ and angle θ_{ij}. When the position $\overrightarrow{P_{ij}}$ is estimated, the new position $\overrightarrow{P_i}$ of the IoRT robot is computed by using the formula (8). Then, the collective movement of all IoRT robots will allow our trained ANN converge to the desired distance D_{th}. The global connectivity will also keep if our ANN is well trained (i.e. if training error equals zero or near to zero). The algorithm below illustrates our ANN-based approach:

Algorithm 2 ANN approach (runs every t units of time)

```
Phase I: Neighbor Discovery
    MyNeighbor = FindNeighbor(RobotId)
Phase II: Estimate the position $\overrightarrow{P_{ij}}$ between two robots
    $\overrightarrow{P_{ij}} = trained\_ann(d(i,j), \theta_{ij})$
Phase III: Compute the new position $\overrightarrow{P_i}$
    Compute $\overrightarrow{P_i}$ using Formula (8)
Phase IV: Deployment
        move to $\overrightarrow{P_i}$
```

V. EVALUATION & DISCUSSION OF THE RESULTS

In this section we first describe the simulation parameters and then provide the simulation results of our approaches. We are interested in studying how our approaches converge to the desired distance D_{th} between any pair of IoRT robot (hence to the desired communication quality matching D_{th}). We will see also how the density of IoRT robots influences the traveled distance of a robot. The importance of taking account the algebraic connectivity before taking a movement decision will be also highlighted. Our approaches will be compared to the approach described in [13] called hereafter EVFA (Extended Virtual Force-Based Approach). EVFA was designed by its authors to overcome the connectivity maintenance and nodes stacking problems in the traditional Virtual Force Algorithm (VFA). Unlike our approaches, EVFA is based only on the orientation force and the judgment of distance force between node and its one-hop neighbors.

We assess our techniques w.r.t. (i) the algebraic connectivity, (ii) the robot traveled distance, (iii) the average distance, and (iv) the QoS level expressed in terms of RSSI (Received Signal Strength Indicator). Simulations have been carried out for a variable number of robots (i.e. from 5 to 50 robots) in an area of 3x3 km.

A. Simulation parameters

All the algorithms in this paper were implemented in version 2.29 of Network Simulator with patch from [10] that reflect a realistic channel propagation and error model. The patch is used in order to provide the effect of interference and different thermal noises to compute the signal to noise plus interference ratio (SINR) and accounting for different bit error rate (BER) to SINR curves for the various codings employed [15]. Table 1 summarizes the parameters used in the simulations.

Physical	Propagation model	Two ray ground
	Error model	Real
	Antennas gain	Gt = Gr = 1
	Antennas height	ht = hr = 1 m
	Communication range	250 m
Statistics	Number of samples	100
	Simulation time	3000s
	Confidence Interval	95%
Mobility	Computation of the new position	see formula (8)
	Damping coefficient k	0.5
	D_{th}	212 m
ANN	Layer number	4
	Input number	2
	Output number	1
	Neuron's number in hidden layers	15
	Desired Error	0.00001
	Max epochs	10000
	Activation function	sigmoid symmetric
	Learning rate	0.2
	Training algorithm	backpropagation
Topology	Topology width	3000 m
	Topology height	3000 m

Table I: Simulation parameters

B. Simulation results

All of the following results are the average of 100 times simulations and we assume that the topology is totally connected at the beginning of the simulation.

We can observe in the Figures [3-6](a) that IoT-based approach always kept the global connectivity since it always take account the algebraic connectivity constraint. Unlike IoT-based, EVFA has a connectivity problem when the robots density is low and this can explain why EVFA traveled a lot when the number of robot is less than 15 (see Figures 2(a)). The goodness of connectivity is observed in EVFA when the robots density is higher (i.e. $>$ 15 robots). However, as we mentioned before our goal is not to maximize the algebraic connectivity but just to keep it always greater than zero. This condition is enough to keep the global connectivity.

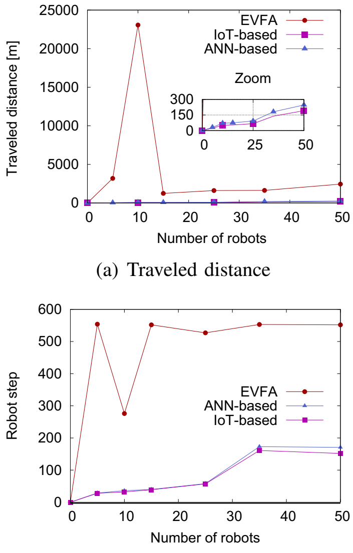
Figures [3-6](c) and [3-6](d) illustrate the convergence of our algorithms to the desired distance and the desired communication quality (RSSI) throughout the simulation. We can notice that our approaches converge quickly to the aforementioned parameters which is not always the case for EVFA. We can see also that ANN mimics perfectly the behaviours of the IoT-based approach. This is due to the fact that our neural network has been well trained.

Figures [3-6](b) depict the distance traveled by a robot during the simulation time. By considering the relationship between energy and traveled distance, we can say that our approaches are energy efficient as compared to EVFA. However, it is observed that ANN consumes a bit more than Iot-based approach. This is because ANN has made a bit more step[1] to converge (see Figures 2(b)).

As a conclusion, it is worth to say that the global connectivity is reached with ANN approach since it was well trained and inherited the characteristics of IoT-based approach.

[1] Each robot increments its counter when it decides to move. In this paper, robot step is defined as the average of the maximum counter value achieved by one robot during the simulation.

It should be noted that the coverage rate strongly depends on the initial network topology. If initially the robots are very close to each other, the convergence to the desired distance enlarges the collective coverage. Else, if the distances between robots are initially bigger than the desired distance, the coverage rate will decrease but the communication quality will improved. This has been proven but we omit it in this paper for lack of space.



(a) Traveled distance

(b) Robot step

Figure 2: Traveled distance and robot step according to the robots number

VI. Conclusions

In this paper, we implemented a IoT-based and an ANN control scheme to maintain global connectivity among multiple IoRT robots. The proposed approaches tried to capture the trade-off between network coverage and communication quality expressed as RSSI level. The proposed algorithms allow the whole IoRT robot network converges to the desired distance, and hence the desired communication quality. Through extensive simulation we showed that our approaches outperform the EVFA approach proposed in [13], in terms of traveled distance and convergence time. Moreover, Our proposed methods always maintain the global connectivity throughout the simulation.

Acknowledgment

This work was partially supported by a grant from CPER Nord-Pas-de-Calais/FEDER Campus Intelligence Ambiante.

References

[1] Ajith Abraham. Artificial neural networks. *handbook of measuring system design*, 2005.

[2] Gianluca Aloi, Luca Bedogni, Marco Di Felice, Valeria Loscri, Antonella Molinaro, Enrico Natalizio, Pasquale Pace, Giuseppe Ruggei, Angelo Trotta, and Nicola Roberto Zema. Stemnet: an evolutionary network architecture for smart and sustainable cities. *Transactions on Emerging Telecommunications Technologies*, 25(1):21–40, 2012.

[3] Tamio Arai, Enrico Pagello, and Lynne E Parker. Editorial: Advances in multi-robot systems. *IEEE Transactions on robotics and automation*, 18(5):655–661, 2002.

[4] Noury Bouraqadi, Serge Stinckwich, Victor Moraru, Arnaud Doniec, et al. Making networked robots connectivity-aware. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3502–3507. IEEE, 2009.

[5] Y Uny Cao, Alex S Fukunaga, and Andrew Kahng. Cooperative mobile robotics: Antecedents and directions. *Autonomous robots*, 4(1):7–27, 1997.

[6] Arnaud Casteigts, Jérémie Albert, Serge Chaumette, Amiya Nayak, and Ivan Stojmenovic. Biconnecting a network of mobile robots using virtual angular forces. *Computer Communications*, 35(9):1038–1046, 2012.

[7] Jiming Chen, Shijian Li, and Youxian Sun. Novel deployment schemes for mobile sensor networks. *Sensors*, 7(11):2907–2919, 2007.

[8] Council. Internet of things council. http://www.theinternetofthings.eu. Accessed November 18, 2015.

[9] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2):298–305, 1973.

[10] Marco Fiore. Ns-2.29 wireless update patch. http://perso.citi.insa-lyon.fr/mfiore/research.html. Accessed November 19, 2015.

[11] Maria Carmela De Gennaro and Ali Jadbabaie. Decentralized control of connectivity for multi-agent systems. In *Decision and Control, 2006 45th IEEE Conference on*, pages 3628–3633. Citeseer, 2006.

[12] M Ani Hsieh, Anthony Cowley, R Vijay Kumar, and Camillo J Taylor. Maintaining network connectivity and performance in robot teams. 2008.

[13] Jun Li, Baihai Zhang, Lingguo Cui, and Senchun Chai. An extended virtual force-based approach to distributed self-deployment in mobile sensor networks. *International Journal of Distributed Sensor Networks*, 2012, 2012.

[14] Lynne E Parker. Multiple mobile robot systems. In *Springer Handbook of Robotics*, pages 921–941. Springer, 2008.

[15] Javier Del Prado Pavon and Sunghyun Choi. Link adaptation strategy for ieee 802.11 wlan via received signal strength measurement. 2:1108–1113, 2003.

[16] Riccardo Petrolo, Valeria Loscri, and Nathalie Mitton. Towards a smart city based on cloud of thoings. In *Proceedings of the 2014 ACM international workshop on Wireless and mobile technologies for smart cities, WiMobCity*, pages 61–66. ACM, 2014.

[17] Ou Qinghai, Zhen Yan, Li Xianghen, Zhang Yiying, and Zeng Lingkang. Application of internet of things in smart grid power transmission. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 2, pages 1293–1303. IEEE, 2003.

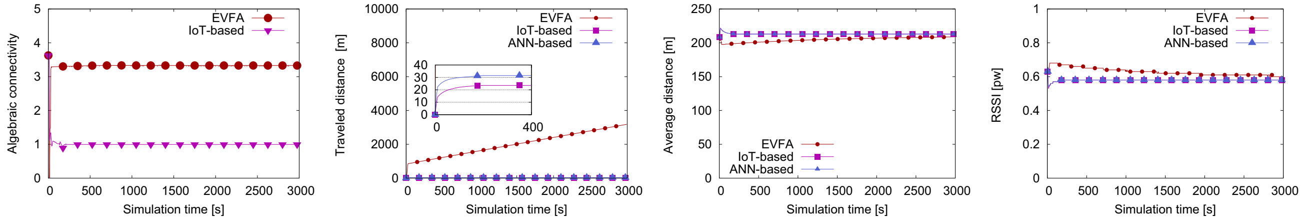
[18] ABI Research. Internet of robotic things. https://www.abiresearch.com/market-research/product/1019712-the-internet-of-robotic-things. Accessed November 2, 2015.

[19] Martin Riedmiller. Advanced supervised learning in multi-layer perceptronsfrom backpropagation to adaptive learning algorithms. *Computer Standards & Interfaces*, 16(3):265–278, 1994.

[20] Lorenzo Sabattini, Cristian Secchi, Nikhil Chopra, and Andrea Gasparri. Distributed control of multirobot systems with global connectivity maintenance. *Robotics, IEEE Transactions on*, 29(5):1326–1332, 2013.

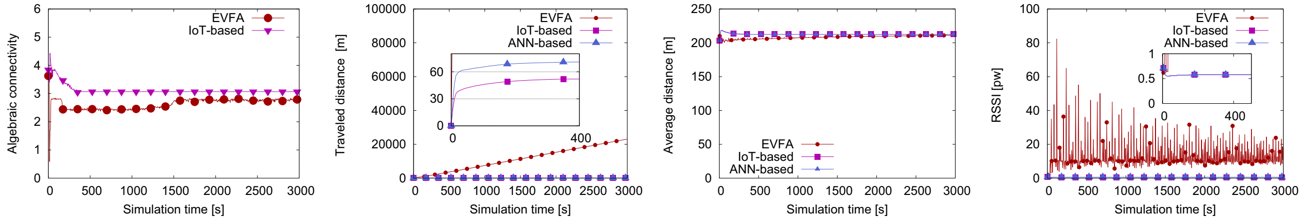
[21] Ethan Stump, Ali Jadbabaie, and Vijay Kumar. Connectivity management in mobile robot teams. In *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, pages 1525–1530. IEEE, 2008.

[22] Yao Zou and Krishnendu Chakrabarty. Sensor deployment and target localization based on virtual forces. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 2, pages 1293–1303. IEEE, 2003.



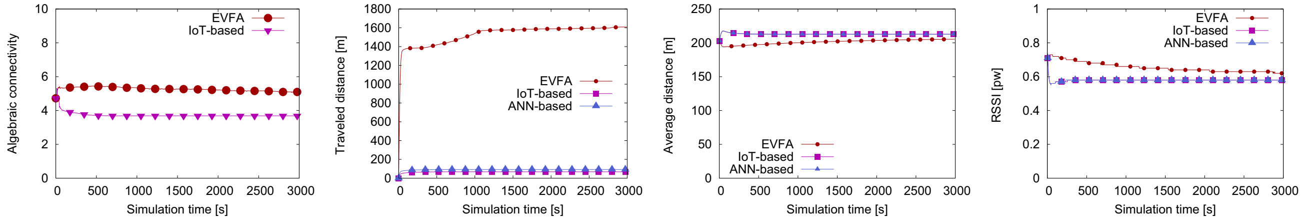
(a) Algebraic connectivity (b) Traveled distance (c) Position (d) RSSI

Figure 3: Simulation results obtained with 5 robots moving in 3x3 km area



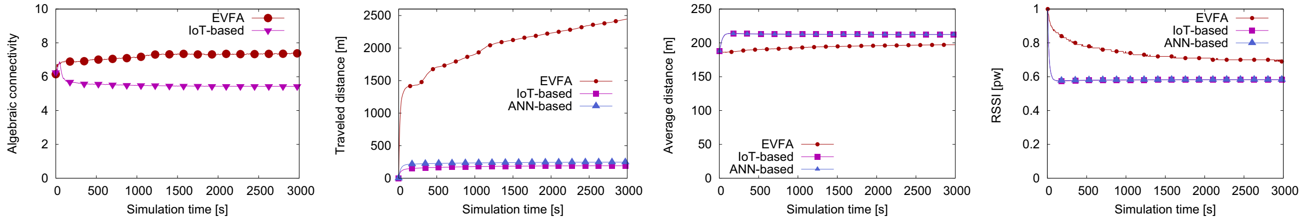
(a) Algebraic connectivity (b) Traveled distance (c) Position (d) RSSI

Figure 4: Simulation results obtained with 10 robots moving in 3x3 km area



(a) Algebraic connectivity (b) Traveled distance (c) Position (d) RSSI

Figure 5: Simulation results obtained with 25 robots moving in 3x3 km area



(a) Algebraic connectivity (b) Traveled distance (c) Position (d) RSSI

Figure 6: Simulation results obtained with 50 robots moving in 3x3 km area