Multi-Robot Coverage: A Bee Pheromone Signalling Approach

Ipek Caliskanelli, Bastian Broecker, and Karl Tuyls

Department of Computer Science, University of Liverpool
{ipek.caliskanelli,bastian.broecker,karl.tuyls}@liverpool.ac.uk

Abstract. In this paper we propose **BeePCo**, a multi-robot coverage approach based on honeybee colony behaviour. Specifically, we propose a honeybee inspired pheromone signalling method that allows a team of robots to maximise the total area covered in an environment in a distributed manner. The effectiveness of the proposed algorithm is experimentally evaluated on two different scenarios: when network connectivity is assured and when network connectivity is ignored. Simulated experimental results in various scenarios indicate that **BeePCo** provides a stable area coverage.

Key words: multi-robot systems, bio-inspired, bee-inspired, coverage

1 Introduction

Recent years have seen a rapidly growing interest in multi-robot systems for automatically surveilling environments of different size, type and complexity. Multi-robot systems (MRS) consist of multiple interacting robots, each executing an application-specific control strategy, which is not centrally steered. The interest in MRS for surveillance is largely motivated by the wide range of application areas including the protection of safety-critical technical infrastructures and buildings, search and rescue scenarios, the monitoring of danger zones which cannot be entered by humans in the case of for instance a nuclear incident or a bio-hazard, etc. As such automated surveillance has become a well studied topic in multi-robot research with a strong practical relevance.

A key advantage of robot-based surveillance lies in its flexibility achieved through possible positional changes of the robots, which makes this form also suited for surveillance applications in unknown or complex environments. In contrast to stationary wireless sensor based surveillance systems or networks, however, robot-based surveillance systems have not yet found their way to real-world applications on a broader scale. Two interrelated key components of every multi-robot surveillance system are exploration and coverage of a potentially unknown environment. The term *exploration* refers to the discovery of all traversable regions of the environment through one or several robots [28]. The term *coverage* refers to the maximisation of (or the process of maximising) the total area covered by the sensors of the involved robot(s) [28].

Previously, we have investigated three different biological inspirations: the stigmergy principle of ants, foraging behaviour of honeybee colonies and pheromone signalling procedure of honeybees. *StiCo*, the stigmergy principle, is based on the observations of ant colonies, and used as a coordination mechanism for coverage by multi-robot systems [28]. Foraging behaviour of honeybee colonies [23] are inspected and used to solve robot coordination, navigation and path planning issues in multi-robot platforms. *PS* [7], honeybee inspired pheromone signalling procedure, is used to address load balancing and redundancy control issues in wireless sensor networks.

In this paper we are mainly concerned with coverage issues of multi-robot systems. Specifically we explore the performance outcomes of the bees *pheromone signalling* procedure, which we call *BeePCo*, when applied to the coverage problem in multi-robot systems. *BeePCo* mechanism proposed is inspired by biological processes: how social insects (bees) control and orchestrate with other members of a hive [1, 2]. As abstract agents, individual bees have many similarities with robots (as do bee colonies with MRSs). The required similarities are in terms of individual wellbeing (bee/robot) and collective welfare (colony/MRS). With our approach, we enable coordination among robots in terms of which direction should each individual move based only on local information. The proposed approach is evaluated using a simulator.

The remainder of this paper is structured as follows. Section 2 reviews the related work in the areas of multi-agent coverage and bio-inspired techniques in networked distributed systems. Section 3 covers pheromone signalling based coverage algorithms for MRSs together with the required biological background. The paper continues with the experimental setup and results in Section 4. We conclude in Section 5.

2 Related Work

The section gives an overview of relevant literature that has attempted to describe, analyse, or efficiently exploit bio-inspired techniques for multi-agent coverage problem. This section is split into two main parts of the problem targeted in this research: Section 2.1 provides examples of existence work in the fields of multi-agent coverage in MRSs, whereas Section 2.2 shows the significant bio-inspired research work in the fields of networked distributed systems in general.

2.1 Multi Robot Coverage

The concept of coverage is a metric for evaluating robotic systems which was first introduced by Gage [13]. Gage defines three basis types of coverage: blanket coverage, where is objective is to achieve node formation which maximises the total detection area; barrier coverage, which aims to minimise the probability of undetected intrusion through the barrier; and sweep- or repetitive-coverage with the goal to cover all accessible interest points in an given environment over time, while maximising the rate of visits over all points and minimising the total distance travelled by all robots.

Blanket coverage is most common for the deployment of mobile sensor networks in an unknown environment; the sensor nodes are initially placed in a compact configuration, where the nodes are trying to spread out such that the area covered by the network is maximised. One example for such a use case is a hazardous material leak in a damaged structure. Mobile sensor nodes equipped with chemical sensors spread out from a initial position to gather information about location and concentration of the hazard. Due to the fact that the communication infrastructure could be damaged, the nodes have to insure their own network structure even if single nodes get lost or destroyed. Many approaches in this field are based on the potential field technique first introduced by Khatib [22].

Barrier and repetitive-coverage problems are originating from the computational geometry *Art Gallery Problem* [10] and its variant for mobile guard for mobile guards, the *Watchman Route Problem* [24].

In robotics, *repetitive-coverage* can be described as a problem where a team of robots has to visit multiple *points of interests*(POI) in a known environment frequently, to perform certain tasks. The goal of such algorithms is to keep the average visit frequency over all POIs high, while achieving a minimal total travelled distance and a balanced workload over all robots. Typical real world use cases for such problems are patrolling, lawn mowing and chemical spill clean up. Many approaches concerning multi-robot patrol partition the area into subareas divided between the robots. Inside such a sub-area, each robot applies a single robot patrol algorithm. Ahmadi and Stone [1] describe a negotiation-based approach for distributing the area between the robots and dealing with events such as addition or removal of robots to the environment. Jung and Sukhatme [19] introduce a region based approach for tracking targets in a system with mobile robots and stationary sensors.

Another important form of multi-robot coverage is *terrain coverage* or multi-robot exploration. It can be defined as a problem where a robot tries to visit each and every location in a continues bounded unknown environment by avoiding obstacles and perform defined tasks [8, 12, 25]. A terrain coverage algorithm must generate a coverage path, which is a chain of motion steps for a robot, the optimal coverage path takes minimal time and guarantee to cover the entire terrain and perform the task efficiently.

Most approaches divide the in the environment into grid cells and explored one cell at the time until the whole area is covered. One of the first was Spanning Tree Coverage (STC) which solves single robot coverage optimistically [11]. The same idea was applied by Hazon and Kaminka on a multi-robot system [17].

Batalin et al. propose a two multi-robot algorithm, which spread the robots in the terrain and makes them avoid each others sensing area [3].

Several authors proposed marked in multi-robot exploration, in which robots are making bids on a sub task of a exploration attempted [33, 36]. These bids are based on values such as expected information gain and travelled cost to a particular location. This approach seems to minimise the costs while maximising the benefit.

2.2 Bio-Inspired Solutions

Bio-inspired solutions are often used to solve complex problems (i.e. MAC level routing, load balancing, task allocation and resource scheduling, network coverage, and emergence) in the broad research area of distributed systems with a particular interest on wireless sensor networks, many and multicore systems, swarm intelligence and multi-robot systems to make systems more reliable, efficient and self-organised. Ant colony optimisation, bee colony optimisation and artificial immune system are three of the most commonly used biological inspirations.

Based on the observation of the collective foraging behaviour of ants, many research studies are held on **Ant Colony Optimisation (ACO)** on the ability of ants to converge on the shortest path from their nest to a food source to improve energy efficiency and QoS in routing. ARA [15], AntHocNet [9], ARO [35] and *StiCo* [28] can be listed as some of the key researches of ACO.

Conforming to this swarm metaphor, **Bee Colony Optimisation (BCO)** was introduced by Karaboga et al. [21], [20]. Scientist are inspired by varies different behaviours of bees: foraging behaviour in Lemmens at al. [23], Beehive protocol [34], BeeSensor [29]; bees mating procedure in [32], [27]; pheromone signalling mechanism in PS [7].

AIS is inspired by the human/mammalian immune system. Sensitivity to detecting environmental change, and identifying the foreign/infectious agents is used, particularly for security purposes in anomaly detections. SASHA [4], DSR [31], [30], DNRS [2] are some of the significant research in the fields of autonomous distributed systems inspired by AIS.

BTMS [16] uses zygote differentiation to extend the network lifetime whilst speeding up task mapping and scheduling. Homogeneous nodes begin in a default state and within time nodes differentiate themselves dynamically to perform distinct tasks according to their location.

In our previous work, pheromone signalling based load-balancing, *PS* [5, 7], we present a dynamic technique for WSNs that is applied at run time at the application layer. *PS* is inspired from the pheromone signalling mechanism found in bees and provides distributed WSN control that uses local information only. *PS* is unique; unlike many load balancing approaches are applied at link or network layer [18, 14, 34] and balance only communication load, *PS* is an application-layer protocol and manages both computation and communication load. In [6], we extend our initial *PS* technique by introducing additional network elements in the form of robotic vehicles for Wireless Sensor and Robot Networks (WSRNs). We merge different subclasses of cyber-physical systems (sensors and robots) together to increase the area coverage *effectively*, which directly increases the service availability and extend the network lifetime by benefiting from their heterogeneity. *Effective* area coverage in this research is defined as achieving the highest service availability by moving less. To achieve the desired effective area coverage, we have extended our *PS* technique to guide robots towards the areas of the sensor field where the sensor nodes have run out of battery and are unable to provide service. The same pheromone signalling process is applied into multi-robot systems in this research and explained in details in the next sections.

3 Pheromone Signalling Based Coverage Technique

We described our previous work on pheromone signalling algorithm which is applied on a WSNs domain. Unlike our previous work on WSNs, this paper focuses on applying pheromone signalling technique on MRSs. Both WSNs and MRSs contains different application-specifications, and in order to indicate the application domain we change the name of the pheromone signalling technique (from *PS* on WSNs) to *BeePCo* on MRSs. The bee-inspired coverage algorithm, *BeePCo*, described in this section is a completely decentralised approach that have low computation and communication overheads.

Changes in pheromone levels are used by many social animals to orchestrate the colony by assigning responsibilities to each individual. Roberts [26] explains the process of larvae differentiation in beehives as an example of such orchestration. Bees have developed a special hormonal system to ensure every beehive has a queen, which maintains the stability of the colony and orchestrates the behaviour of all other bees. Throughout its life, a queen bee stimulates a pheromone called Queen Mandibular Pheromone (QMP), which makes the worker bees aware of its presence as queen. This hormonal mechanism works as follows: the worker bees lick the queen bee and pass the pheromone to the others. If there is no pheromone passed through the worker bees, they will then consider the queen as dead. In that case, workers will select a larva to be fed with large amounts of the royalactin protein. That protein induces the differentiation of honeybee larvae into a queen. If worker bees keep receiving the pheromone, they will be aware that there is a queen bee to orchestrate the colony and will take no action towards building a new queen.

The proposed coverage technique is inspired by the behaviour described above. The role of queen bee denotes a robot that is responsible for managing the execution of all service requests it receives. Throughout this paper we will refer these robots as Queen Robot (QR) and their responsibility (service) is to patrol in the field. The basic strategy of the algorithm is based on the periodic transmission of pheromone by QRs, and its retransmission by recipients to their neighbours. The pheromone level at each robot decays with time and with distance to the source. All robots accumulate pheromone received from other QRs and if at a particular time the pheromone level of a robot is below a given threshold this robot will differentiate itself into a QR. To make it clear, the threshold we used for this work is 0 and all the robots are QRs at all times. Although we do not particularly benefit from robot differentiation for this work, we still describe the differentiation process for the completeness of this work and to provide a base for our future work. In the *BeePCo* technique, the level of pheromone indicates the resource usage and robot density in a particular area of the network. Areas in the robotic arena that have lower level of pheromone at a given time demonstrate less resource usage, and less robot density as opposed to other parts of the network. This means, areas with low pheromone level have either a low coverage or not covered at all.

The proposed *BeePCo* algorithm consists of four parts which are executed on every robot of the MRS: two of them are time-triggered (differentiation cycle and decay of pheromone), whereas other two (propagation of received pheromone and robotic move) occurs at the same time in one event-triggered process. The first time-triggered part, referred to as the differentiation cycle (Algorithm 1), is executed by every robot of the MRS every T_{QR} time units. On each execution, the robots checks its current pheromone level h_i against a predefined level $threshold_{QR}$. We set the $threshold_{QR}$ to 0 for this paper, which means all the robots are assign to be QRs and they remain as QRs until they run out of energy. QRs transmits pheromone to its network neighbourhood to make its presence felt. Each pheromone dose hd is represented as a two-position vector. The first element of the vector denotes the distance in hops to the QR that has produced it (and therefore is initialised as 0 in line 4 of Algorithm 1). The second element is the actual dosage of the pheromone that will be absorbed by the neighbours.

Algorithm 1 Differentiation Cycle

1: every T_{QR} do
2: **if** $(h_i < threshold_{QR})$ **then**
3: $\quad QR_i$ =true
4: $\quad$ broadcast $hd = \{0, h_{QR}\}$
5: **else**
6: $\quad QR_i = false$
7: **end if**

The event-triggered part of*BeePCo* deals with the propagation of the pheromone released by QRs (as described above in the differentiation cycle) and received at neighbouring robots. The purpose of propagation is to extend the influence of QRs to its surroundings (their directly connected neighbours in the communication range). Propagation is not a periodic activity, and happens every time a robot receives a pheromone dose. Its pseudocode given in Algorithm 2. Upon receiving a pheromone dose, a robor checks whether the QR that has produced it is sufficiently near for the pheromone to be effective. It does that by comparing the first element of hd with a predefined $threshold_{hopcount}$. If the hd has travelled more hops than the threshold, the robot simply discards it. If not, it adds the received dosage of the pheromone to its own pheromone level h_i and propagates the pheromone to its neighbourhood. Before forwarding it, the robot updates the hd vector element by incrementing the hop count, and by multiplying the dosage by a decay factor $0 < K_{HOPDECAY} < 1$. This represents pheromone transmission decaying with distance from the source. Once the pheromones are propagated, Move Cycle occurs as an event-triggered way. As well as propagation cycle, move cycle also occurs when a robot receives pheromone. Move cycle illustrates the general behaviour of a robot as given in Algorithm 3.

Algorithm 2 Pheromone Propagation Cycle

```
while hd is received do
  if (hd[1] < threshold_hopcount) then
    h_i = h_i + hd[2]
    broadcast hd = {hd[1] + 1, hd[2].K_HOPDECAY}
  else
    drop hd
  end if
  go to BeePCo Move Cycle
end while
```

Algorithm 3 Move Cycle

```
if (network connectivity is assured) then
  if (numConnectedNodes¿1 and backwardsMove==0) then
    if (pheromone received) then
      PS-guided moving decision
    else
      broadcast communication link request
      establish local communication links
    end if
  else if (numConnectedNodes¡1 and backwardsMove==0) then
    move backwards to previous location
  else
    broadcast communication link request
    establish local communication links
  end if
  if (network connectivity is not assured) then
    if (pheromone received) then
      PS-guided moving decision
    else
      Keep moving as the direction of the last move
      broadcast communication link request
      establish local communication links
    end if
  end if
end if
```

If a robot receives pheromone it makes a moving decision and selects a target destination in the opposite direction of the received pheromone based on

BeePCo. The moving decision of robots are based on vector addition and its pseudo code appears in Algorithm 4. Given the mathematical formulation in the pseudocode and assuming all the robot know their location as x and y coordinates, we calculate the angle of the received pheromone with the use of the sender's x and y coordinates. To do this, we resolve the horizontal and vertical components based on the amount of received pheromone level, h_i, and the coordinates of the QRs. In order to find the magnitude, we sum up all the horizontal and vertical components. In order to determine the direction of the magnitude, we take arctangent of the magnitude and resolve x and y coordinates. This process happens on-demand as the robotic agents receive pheromone from as part of propagation cycle.

Algorithm 4 Moving Decision

if $(h_i > 0)$ **then**
  for all the received pheromones (p) of the robot **do**
    $diff_X = p_{Sender_X} - currentCoordinate_X$
    $diff_Y = p_{Sender_Y} - currentCoordinate_Y$
    $\theta = ArcTangentQuadrant(diff_Y, diff_X)$
    $component_X = p.hd * \cos\theta$
    $component_Y = p.hd * \sin\theta$
    $Sum_X += component_X$
    $Sum_Y += component_Y$
  end for
end if
$magnitude = \sqrt{{Sum_X}^2 + {Sum_Y}^2}$
$\theta_{destination} = ArcTangentQuadrant(Sum_Y, Sum_X)$
apply 180 degrees shift to $\theta_{destination}$
clear all received pheromones

If a robot does not receive any pheromone by the time it arrives to its destination then the robot keep going in the same direction that it moved last. This happens when robots are not in each others communication ranges (when they cannot receive pheromone with each) to allow them spreading in the area.

The second time-triggered part of the algorithm, shown in Algorithm 5 is a simple periodic decay of the pheromone level of each robot. Every T_{DECAY} time units, h_i is multiplied by a decay factor $0 < K_{TIMEDECAY} < 1$.

Algorithm 5 Decay Cycle

for every T_{DECAY} **do**
  $h_i = h_i.K_{TIMEDECAY}$
end for

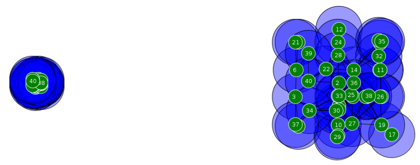
4 Evaluation Environment and Experimental Results

To evaluate *BeePCo*, we have designed a three-tier system-level simulation model that represents the application-layer (consisting of tasks), platform layer (consisting of robots) and the mapper (that maps the tasks from the application-layer to the platform-layer). Our system-level simulator, *Fast*, is written in Java and it is an abstract simulator - trading accuracy for efficiency, scalability and flexibility. For further details about *Fast*, please see [5, 7].

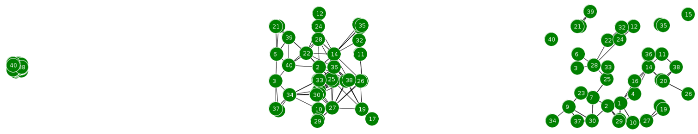
The set of the experimental work presented in this section aims to show the area coverage and network connectivity of the proposed *BeePCo* technique. Area coverage in this study is referred as maximisation of the total area covered by the sensors of the involved robot(s), as defined in [28]. On the other hand, network connectivity is referred to the ability of transfer data between robots. As the proposed technique is more suitable for non-critical applications, we focus on participial network connectivity instead of full connectivity. The simulation setup is developed on a system of 20, 30 and 40 robots with each having a sensing and communication radius of 25 cm. The application arena size is set to 300cm x 300cm, where the robots initially deployed randomly in the centre of the arena, in a square region of size 5cm x 5cm. For this set of experiments, we create two different scenarios that uses *BeePCo* algorithm as follows:

- *BeePCo with network assurance* represents a case where the robots try to keep the wireless communication channels alive throughout the simulation. This restricts robots to move further along side each other; instead they stay close by. The algorithm is forcing robots to move backwards if a robot is not connected to at least one other robot in the arena. This scenario is developed to be used for applications where events are expected to be reported to the sink, such as patrolling purposes that long-term periodic reported is required.
- *BeePCo without network assurance* represents a case where a wide spread of the robots on the arena is appreciated without concerning about the communication between each other. This scenario is developed for one-shot applications where fast exploration of an unknown is needed, such as rescue scenarios.
- *MaxCo* represents the optimal case when robots' transmission range do not intersect with each other. This scenario is a benchmark for the maximum possible coverage of deployed robots with zero surveillance area overlap in the 300cm x 300cm arena.

Fig. 1 shows the spread of 40 robots when network assurance is applied on *BeePCo* algorithm. Fig. 1(a), 1(b), 1(c) represents the layout of the robots and their spread over the arena incrementally. Area that is covered by each robot is plotted with blue circle to indicate the transmission range of each robot, and is normalised with the surface of the arena. Fig. 1(d), 1(e), 1(f) shows how the active wireless communication links between the robots evolve as the time elapse. Robots are deployed in the middle of the arena in the initial stage as shown in Fig. 1(a), and 1(d). In the Initial stage, coverage is very low, whereas



(a) Initial state coverage (b) Mid state coverage (c) Final state coverage

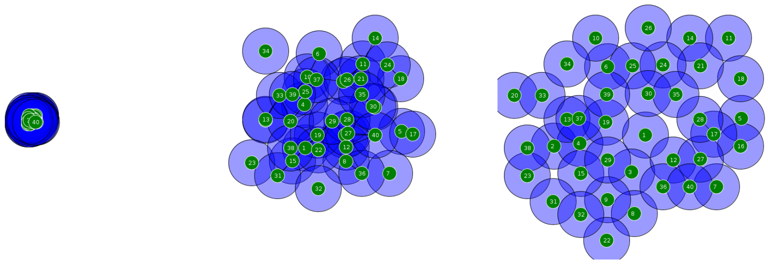


(d) Initial state communication links (e) Mid state communication links (f) Final state communication links

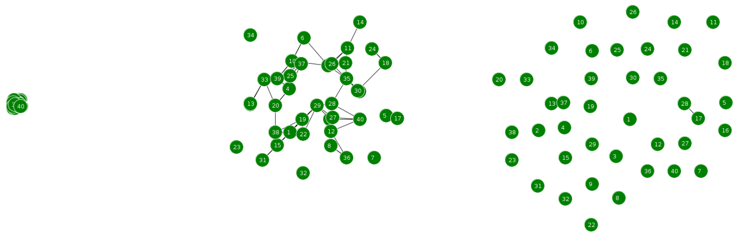
Fig. 1: Spread of a MRS of 40 robots: (a),(b),(c) area coverage, and (d),(e),(f) connectivity of the communication links when network connectivity is assured by *BeePCo*.

connectivity is very high. As expected, this situation occurs only because the robots are cluttered in the middle of the arena. As the time elapse *BeePCo* algorithm proceeds. Robots start spreading in the arena as shown in Fig. 1(b), and 1(e); the area coverage increases where the network connectivity remains high. At the end of the simulation, robots spread out as much as they can do whilst trying to keep the wireless communication channels active as shown in Fig. 1(c), and 1(f).

Fig. 2 illustrates the spread if the robots when network is not assurance on *BeePCo* algorithm. Similar to Fig. 1, area coverage and connectivity is inspected on an arena with 40 robots. To clarity, for this set of experiments, we use the same setup with Fig. 1. Fig. 2(a), and 2(d) exhibits the initial stage of the robots after deployment. As can see from the Fig. 2(a), *BeePCo* algorithm without network assurance performs very similar to Fig. 1(a), and 1(d) *BeePCo* algorithm with network assurance. Later on as the simulation evolves, Fig. 2(b) and 2(e) start spreading wider as opposed to Fig. 1(b), and 1(e). By the end of the simulation, robots are being spread all along the arena where the area coverage get very high as shown in Fig. 2(c). Unlike the area coverage, network connectivity is



(a) Initial state coverage (b) Mid state coverage (c) Final state coverage



(d) Initial state communication links (e) Mid state communication links (f) Final state communication links

Fig. 2: Spread of a MRS of 40 robots: (a),(b),(c) area coverage, and (d),(e),(f) connectivity of the communication links when network connectivity is assured by *BeePCo*.

lost almost entirely. Dramatic difference in the network connectivity is observed between Fig. 1(f) and Fig. 2(f).

The results shown in Fig. 3 is based on 30 different runs over six different configurations (with and without network assurance each the 3 alternatives for the number of robots in the environment), in a total of 180 simulation runs to ensure the statistical significance. Each run simulated the case study for 13 weeks, to illustrate the long-term effects of the spread of the *BeePCo* coverage algorithm based on bees pheromone signalling process. MaxCo results are based on mathematical calculations based on the total transmission area over total area.

Fig. 3 shows the percentage of the area coverage (a), (c) and the percentage of network connectivity (b), (d). *BeePCo* algorithm to illustrate the effects of the number of the robots on two different scenarios: with and without the network assurance. In addition to experiments shown in Fig. 1 and 2, number of robots is varied (with 20 and 30) and experiments are held whilst the simulation setup kept the same. Our observations are as follows:

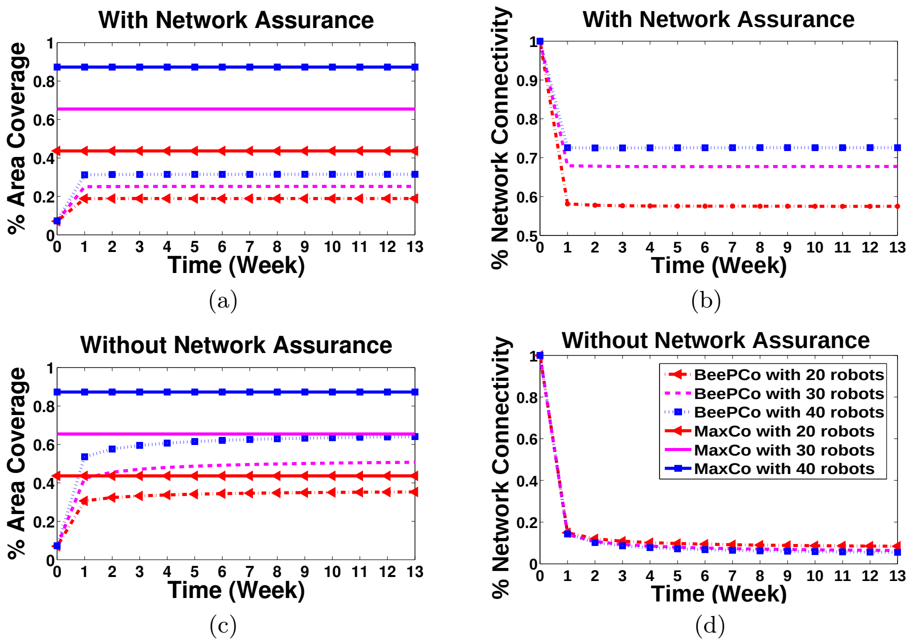


Fig. 3: Experimental results: (a),(c) % area coverage, (b),(d) % network connectivity with different number of robots in a MRS.

Independent from the network assurance, the area coverage increases as the number of the robots increase. This behaviour is shown in Fig. 3(a) and 3(c) on *BeePCo* algorithm and MaxCo; area coverage achieves highest percentage with 40 robots. Area coverage is approximately 10% more when network connectivity is not assured in all three ranges of number of robots. Network connectivity is very low when network is not assured as shown in Fig. 3(d) and is irrelevant from the number of the robots. Fig. 3(a), exhibit that *BeePCo* algorithm with network assurance increases network connectivity 50% more than *BeePCo* with no network assurance. This, does not reflect on area coverage, which we believe is the benefit of *BeePCo* with network assurance. The performance difference in terms of area coverage between *BeePCo* with and without network assurance is less than 10% in a system with 20 robots. Although this difference increases upto 15% as the number of robots increases, we believe *BeePCo* with network assurance brings much more benefit (as opposed to *BeePCo* without network assurance) in terms of connectivity by 50%.

5 Conclusions

In this paper, we have described a bee-inspired robot guidance technique, *BeePCo* in an attempt to address multi-robot coverage problem. The multi-robot coordination and coverage problem is a complicated problem in itself, especially when the limited processing capacity of robots are encountered. As all the communication between the robots are through the wireless medium, it is essential to manage the robot coordination with a computationally lightweight algorithm that consumes low energy. Therefore, we propose to improve multi-robot coverage by guiding the robotics towards the areas where the robot density is low with the use of bees pheromone signalling algorithm. Experimental results on with and without network connectivity assurance demonstrate that our proposed *BeePCo* technique encourages robots to spread apart from each other using the pheromone signalling process. Effects of deployed number of robots on area coverage and network connectivity on both cases where network is assured or not is encountered, and observations are reported.

In the future, we would like to consider the resource limitations of the robots, examining the tradeoff between the total distance taken by a robot and the total service availability of the MRS. From our experience in pheromone signalling algorithm on WSNs, *BeePCo* algorithm can apply in MRSs for redundancy control on top of the current coverage and connectivity procedure in a multi-objective manner. It can be easily inferred from the *BeePCo* differentiation cycle that each robot makes its own decision on whether and when it becomes a QR by referring to local information only: its own pheromone level h_i. Although, for this paper we have allowed all robots to be QRs by setting the predefined $threshold_{QR}$ to 0, that is done only to focus the single objective: to tackle multi-robot coverage problem. In future, we would like to inspect the MRSs behaviour when $threshold_{QR}$ is set to higher number to actually enable robot differentiation. This should allow for a highly self-organised behaviour which fits the requirements for high-density networked MRSs.

References

1. Ahmadi, M., Stone, P.: A multi-robot system for continuous area sweeping tasks. In: Proceedings of the IEEE International Conference on Robotics and Automation. pp. 1724–1729 (May 2006)
2. Atakan, B., Akan, O.B.: Immune system based distributed node and rate selection in wireless sensor networks. In: Bio-Inspired Models of Network, Information and Computing Systems, 2006. 1st. pp. 1–8 (Dec 2006)
3. Batalin, M., Sukhatme, G.: Spreading out: A local approach to multi-robot coverage. In: Asama, H., Arai, T., Fukuda, T., Hasegawa, T. (eds.) Distributed Autonomous Robotic Systems 5, pp. 373–382. Springer Japan (2002)
4. Bokareva, T., Bulusu, N., Sanjay Jha: SASHA: toward a self-healing hybrid sensor network architecture. Embedded Networked Sensors, 2005. EmNetS-II. The 2nd IEEE Workshop on pp. 71–78 (May 2005)
5. Caliskanelli, I., Harbin, J., Indrusiak, L. Polack, F., Mitchell, P., Chesmore, D.: Runtime optimisation in wsns for load balancing using pheromone signalling. In: NESEA, 2012. 3rd IEEE Int. Conf, on (dec 2012)
6. Caliskanelli, I., Indrusiak, L.: Using mobile robotic agents to increase service availability and extend network lifetime on wireless sensor and robot networks. In: INDIN, 2014. 12th IEEE Int. Conf. on (July 2014)

7. Caliskanelli, I., Harbin, J., Soares Indrusiak, L., Mitchell, P., Polack, F., Chesmore, D.: Bio-inspired load balancing in large-scale wsns using pheromone signalling. International Journal of Distributed Sensor Networks (2013)
8. Correll, N., Martinoli, A.: Collective inspection of regular structures using a swarm of miniature robots. In: Ang, MarceloH., J., Khatib, O. (eds.) Experimental Robotics IX, Springer Tracts in Advanced Robotics, vol. 21, pp. 375–386. Springer Berlin Heidelberg (2006)
9. Di Caro, G., Dorigo, M.: Antnet: distributed stigmergetic control for communications networks. J. Artif. Int. Res. 9(1), 317–365 (Dec 1998)
10. El-Sherbeny, N.A.: Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods. Journal of King Saud University - Science 22(3), 123 – 131 (2010)
11. Gabriely, Y., Rimon, E.: Spanning-tree based coverage of continuous areas by a mobile robot. Annals of Mathematics and Artificial Intelligence 31(1-4), 77–98 (2001)
12. Gabriely, Y., Rimon, E.: Competitive on-line coverage of grid environments by a mobile robot. Computational Geometry 24(3), 197 – 224 (2003)
13. Gage, D.W.: Command control for many-robot systems. NAVAL COMMAND CONTROL AND OCEAN SURVEILLANCE CENTER RDT AND E DIV SAN DIEGO CA (1992)
14. Gouda, M.G., McGuire, T.M.: Accelerated heartbeat protocols. In: 18th Int. Conf. on Distributed Computing Systems. pp. 202–209 (1998)
15. Gunes, M., Sorges, U., Bouazizi, I.: ARA-the ant-colony based routing algorithm for MANETs. Parallel Processing Workshops, 2002. Proc. Int. Conf. on pp. 79–85 (2002)
16. Hamouda, Y., Phillips, C.: Biological task mapping and scheduling in wireless sensor networks. In: Communications Technology and Applications, 2009. ICCTA '09. IEEE Int. Conf. on. pp. 914 –919 (Oct 2009)
17. Hazon, N., Kaminka, G.A.: On redundancy, efficiency, and robustness in coverage for multiple robots. Robotics and Autonomous Systems 56(12), 1102 – 1114 (2008), towards Autonomous Robotic Systems 2008: Mobile Robotics in the {UK} 10th British Conference on Mobile Robotics - Towards Autonomous Robotic Systems (TAROS 2007)
18. Heinzelman, W., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: System Sciences, 2000. Proc. of the 33rd Annual Hawaii Int. Conf. on. vol. 2, p. 10 (Jan 2000)
19. Jung, B., Sukhatme, G.: Tracking targets using multiple robots: The effect of environment occlusion. Autonomous Robots 13(3), 191–205 (2002)
20. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (ABC) algorithm. Applications Soft Comput. 8(1), 687–697 (2008)
21. Karaboga, D.: Artificial bee colony algorithm. scholarpedia 5(3), 6915 (2010)
22. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. In: Robotics and Automation. Proc. 1985 IEEE Int. Conf. on. vol. 2, pp. 500–505 (Mar 1985)
23. Lemmens, N., De Jong, S., Tuyls, K., Nowé, A.: Bee behaviour in multi-agent systems. In: Adaptive Agents and Multi-Agent Systems III. Adaptation and Multi-Agent Learning, pp. 145–156. Springer (2008)
24. Packer, E.: Robust Geometric Computing and Optimal Visibility Coverage. Ph.D. thesis, Stony Brook, NY, USA (2008), aAI3338238

25. Pirzadeh, A., Snyder, W.: A unified solution to coverage and search in explored and unexplored terrains using indirect control. In: Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on. pp. 2113–2119 vol.3 (May 1990)
26. Roberts, M.B.V.: Biology: a functional approach. Nelson (1986), http://www.worldcat.org/isbn/0174480199
27. Sahoo, R., Sardar, A., Singh, M., Ray, S., Sarkar, S.: Trust based secure and energy efficient clustering in wireless sensor network: A bee mating approach. In: Maji, P., Ghosh, A., Murty, M., Ghosh, K., Pal, S. (eds.) Pattern Recognition and Machine Intelligence, Lecture Notes in Computer Science, vol. 8251, pp. 100–107. Springer Berlin Heidelberg (2013), http://dx.doi.org/10.1007/978-3-642-45062-4_13
28. Sahraei, B.R., Alers, S., Tuyls, K., Weiss, G.: Stico in action. In: International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013. pp. 1403–1404 (2013)
29. Saleem, M., Farooq, M.: Beesensor: A bee-inspired power aware routing protocol for wireless sensor networks. In: Giacobini, M. (ed.) Applications of Evolutionary Computing, Lecture Notes in Computer Science, vol. 4448, pp. 81–90. Springer Berlin Heidelberg (2007)
30. Sarafijanovic, S., Le Boudec, J.Y.: An artificial immune system approach with secondary response for misbehavior detection in mobile ad hoc networks. Neural Networks, IEEE Trans. on 16(5), 1076–1087 (Sep 2005)
31. Sarafijanović, S., Boudec, J.Y.: An artificial immune system for misbehavior detection in mobile ad-hoc networks with virtual thymus, clustering, danger signal, and memory detectors. In: Nicosia, G., Cutello, V., Bentley, P., Timmis, J. (eds.) Artificial Immune Systems, Lecture Notes in Computer Science, vol. 3239, pp. 342–356. Springer Berlin Heidelberg (Jan 2004)
32. Senthilkumar, J., Chandrasekaran, M.: Improving the performance of wireless sensor network using bee's mating intelligence 55 (2011)
33. Sheng, W., Yang, Q., Tan, J., Xi, N.: Distributed multi-robot coordination in area exploration. Robotics and Autonomous Systems 54(12), 945 – 955 (2006)
34. Wedde, H., Farooq, M., Zhang, Y.: Beehive: An efficient fault-tolerant routing algorithm inspired by honey bee behavior. In: Ant Colony Optimization and Swarm Intelligence, LNCS, vol. 3172, pp. 83–94. Springer Berlin Heidelberg (2004)
35. Xiao-Min Hu, Jun Zhang: Ant routing optimization algorithm for extending the lifetime of wireless sensor networks. SMC, 2010 IEEE Int. Conf. on pp. 738–744 (Oct 2010)
36. Zlot, R., Stentz, A., Dias, M.B., Thayer, S.: Multi-robot exploration controlled by a market economy. In: ICRA. pp. 3016–3023. IEEE (2002)