Towards autonomous robot swarms for multi-target localisation and monitoring with applications to counter IED operations

Robert J. Mullen*, Sarah Barman and Paolo Remagnino

Faculty of Computing, Information Systems and Mathematics, Kingston University, Penrhyn Road, Kingston-Upon-Thames, KT1 2EE, London, UK
E-mail: r.mullen@kingston.ac.uk
E-mail: s.barman@kingston.ac.uk
E-mail: p.remagnino@kingston.ac.uk
*Corresponding author

Dorothy Monekosso

School of Computing and Mathematics, Ulster University, Shore Road, Newtownabbey, County Antrim, BT37 0QB, Northern Ireland
E-mail: dn.monekosso@ulster.ac.uk

Abstract: A swarm robotics approach is adopted in designing a fully autonomous multi-robot-based solution to the problem of locating generic targets within a given search space. A proof of concept system is developed and tested within a 3D simulation environment. A series of laboratory experiments are carried out to assess the performance of the system with respect to the given task of localising and monitoring generic targets, with reference to a counter IED scenario. Further experiments are carried out to evaluate the robustness to robot failure and scalability of the system.

Keywords: swarm robotics; swarm intelligence; multi-robot systems; autonomous robots; autonomous control laws; self-organisation; distributed robotics; robot target localisation; robot target monitoring; robot formation control; counter IED operations.


Biographical notes: Robert J. Mullen received his MRes in Imaging and X-Ray Physics from King’s College, University of London, in 2006. He is currently a Researcher working towards his PhD in Computer Science, within the Digital Image Research Centre, Kingston University, London. His main research interests are in the areas of swarm intelligence and swarm robotics, and image and video processing and analysis.
1 Introduction

Swarm robotics is a relatively new area of research and development that has emerged from the swarm intelligence paradigm (Bonabeau et al., 1999). Much of the inspiration behind swarm intelligence-based systems comes from observations of biological swarms in nature, and in particular the self-organising, emergent properties they exhibit, for example the foraging behaviours of a colony of ants (Dorigo and Gambardella, 1997; Grasse, 1959).

Swarm robotics largely considers systems of multiple, relatively simple, homogeneous robots, with local and limited sensing and communication abilities, that work collectively to achieve some unified goal (Dorigo and Sahin, 2004). The use of multiple robots necessitates the system to be scalable in nature, and the local and limited sensing and communication abilities provide a focus on a decentralised approach. A decentralised system can greatly improve robustness to robot failure, and the use of relatively simple robots offers the potential for the robots to be relatively small and inexpensive, thus increasing robot portability and expendability.
Advancements in robotics hardware are providing more and more options for smaller scale robots than were previously available, which provides more possibilities for swarm robotics systems to be developed to a deployment ready level. The attributes associated with swarm robotics systems, as summarised above, have advantages across a range of applications, including, but not limited to; security and defence, environmental monitoring, search and rescue, structure inspection and cleaning. The foundations of the methods presented in this paper are not limited to any particular application; however, we concentrate here on problem scenarios within security and defence, specifically the problem of localising and monitoring multiple targets/regions of interest (ROIs) with an emphasis on counter improvised explosive device (IED) operations.

The application of swarm robotics in security and defence spans platforms from sub-aqua to ground based, aerial and even space-borne, ranging in size from micro-bots to satellites and numbers from tens to hundreds. The methods presented in this paper are not limited to any particular platform; however, we concentrate here on the use of unmanned ground vehicles (UGVs).

Countering IEDs is a major ongoing problem receiving a vast amount of research and development across a broad range of disciplines. A large sub-set of countering IEDs is the problem of physically detecting them. Traditional detection methods, both in the military domain as well as in humanitarian demining, largely involve manual detection, which is a slow and hazardous method [US Department of State (USDS), 1998]. In recent years a multitude of sensor technologies have been developed to tackle the problem of physically detecting the vast array of different types of IEDs being used today (Zorrette, 2008). These technologies include visible light, infra-red and thermal imaging systems, ground-penetrating radar, acoustic sensors, magnetic resonance and chemical detection such as the method reported in Cumming et al. (2001). Methods of deployment for these sensors include airborne, attached to vehicles, carried by people and attached on-board robots. In cases where closer inspection is required, or indeed when ground level searching would provide higher accuracy in detection, robotic solutions offer the great benefit of distancing human contact from potential IEDs. Existing deployed robots such as the Talon are remote controlled by a human operator, and are used mostly for explosive ordinance disposal (EOD), and to inspect specific locations where a suspected IED has been located.

For the purpose of wide-area searching, we propose that future multi-robot systems have the potential to autonomously search large areas at close range, increasing efficiency in terms of time and reliability of detection (due to the use of multiple robots simultaneously), while distancing human contact from close contact with potential IEDs.

We describe in this paper the elementary functionality of a swarm robotics approach to this problem and provide proof of concept laboratory experiments to highlight the main properties of the system.

In Section 2, we give a brief overview of related work in the area of multi-robot systems, followed by a definition of the problem to be solved in Section 3. Section 4 details the proposed system and described the experiments carried out. Section 5 presents and analyses the results and finally Section 6 provides a discussion and gives concluding remarks on the work presented herein.
2 Related work

Research in the area of swarm robotics, and indeed multi-robot systems in general, remains very active. Many different methods are being developed to achieve autonomous multi-robot control, for a wide range of applications and robotic platforms. We present in this section a brief overview of a selection of related methods.

From the swarm intelligence paradigm, the concept of stigmergy (Grasse, 1959) has been used, for example in Sauter et al. (2005) and Garnier et al. (2007), to achieve multi-robot coordination and control by using artificial pheromones to guide the robots. Similarly, the well-established potential fields method (Khatib, 1986) is still widely used for robot control/coordination. Both of these methods involve computing virtual force vector-fields which guide the robots’ movements. Another similar method uses physics-based laws to compute local artificial forces (Spears et al., 2004; Morgan and Schwartz, 2005) to guide the robots’ movements. The latter method does not require the computation of a global vector field of artificial forces, rather each robot computes the forces it experiences locally and calculates a new displacement vector governed by physics-based laws.

The concept of flocking (Reynolds, 1987) is another well-established technique, which provides coherent coordination and control between multiple mobile agents, and has been used in robotics for example obstacle avoidance (Turgut et al., 2008) and UAV coordination and control (Labonte, 2009).

Particle swarm optimisation (PSO) (Kennedy and Eberhart, 1995), another nature inspired technique, is a population-based stochastic optimisation method inspired by observations of collective movement and social behaviour of biological entities in the wild. A target search algorithm is presented in Hereford and Siebold (2008) where a PSO algorithm is physically embodied in a number of mobile robots to facilitate coordination and control. A multi-robot search process is modelled by PSO in Pugh and Martinoli (2007), and PSO is used to optimise a sequence of controls for multi-robot formation control in Kwok et al. (2006) and Ngo et al. (2007).

A more engineering-based method is reported in Antonelli et al. (2008), which uses a kinematics control approach and merges a number of elementary behaviours into one final behaviour to facilitate the entrapment and escorting of a target by multiple robots.

The latter mentioned approach, along with the methods using physics-based laws to compute virtual forces, tend more towards a deterministic nature, as opposed to the stochastic nature of PSO and the ant-algorithm approaches of Sauter et al. (2005) and Garnier et al. (2007). The stochastic nature of many swarm intelligence methods makes them particularly good at solving unpredictable and dynamic problems. This can also however make it difficult to fully predict the behaviour of the solution itself (Winfield et al., 2008), which, in particular safety critical applications, might be an undesirable feature.

In this paper, we develop multi-robot coordination and control laws based on physics-based virtual forces, similar to those presented in Spears et al. (2004) and Morgan and Schwartz (2005). We employ the swarm intelligence technique of stigmergy to facilitate self-organisation, but implement this in a deterministic way, to maintain a predictable system as much as possible.
Towards autonomous robot swarms

3 Problem definition

The overall goal of the robot swarm is to provide the coordinates of any IEDs present in a specified search area. For the purpose of the laboratory experiments presented in this paper we assume that the robots are capable of detecting, to a given level of accuracy, an IED that is in close proximity (see Section 4). The sub-goals of the robot swarm are to disperse from a common deployment location, search the environment for IEDs, and upon detection, monitor the IED location and inform the operator of a positive detection. Informing the operator could be achieved by means of visual or audible communication of some form, or simple one-way communication to a central command post or nearby support vehicle. With additional ability to track the robots locations, there would be potential to create a map of detected potential IED locations across a wide area. This is however beyond the scope of this paper.

4 The proposed system

Our approach is swarm intelligence driven, and in particular agrees with the general accepted criteria for swarm robotics systems (Dorigo and Sahin, 2004). Perhaps the most relevant and important consequence of adopting swarm intelligence-based methods is that the design will focus on a decentralised approach. A decentralised approach reduces the levels of inter-robot dependency, and in turn increases the scalability of the system, as well as providing high levels of fault tolerance in terms of robot redundancy. For the particular application presented in this paper, an increase in scalability is useful when considering the flexibility of the system, for example being able to increase the number of deployed robots to search a larger area, with no additional computational cost. High tolerance to robot failure is of obvious benefit to this application given the dangerous nature of the task being carried out by the robots. Our approach also aims to minimise the sensory and communication requirements of the robots, again to increase scalability and robustness, as well as to reduce the cost and potential size of the robots, thus also increasing the expendability and portability of individual robots, which is again of particular interest to this chosen application.

4.1 Nature inspired

As described above, a swarm intelligence approach implies that at some stage in the design process, there has been inspiration taken from observations of systems in nature. We specifically employ the use of indirect communication, or stigmergy (Grasse, 1959), as observed in ant colonies in the wild. This type of indirect communication has been used extensively in computational problems for a multitude of applications (Mullen et al., 2009a), with particular success in combinatorial optimisation problems (Papadimitriou and Steiglitz, 1982) after being developed into algorithmic form in Dorigo et al. (1996). Stigmergy is used to drive the emergent behaviour and self-organisation that produces the desired collective behaviour from the swarm of robots.

The basic idea is that instead of communicating information directly between robots, which would create inter-robot dependency, we instead design the system such that each robot can sense neighbouring robots within its local vicinity. The locations of
neighbouring robots, together with other internal factors, are then used to calculate the robot’s new trajectory, in a reactive cooperative manner.

The internal factors depend on the specific task being performed, and are again in some cases inspired by nature (as detailed in Section 4.2).

4.2 Coordination control laws

The coordination control laws are central to the collective behaviour of the swarm, computing the trajectories of individual robots based on their local environment.

The aforementioned indirect communication is implemented here as virtual forces which each robot experiences as a result of its external neighbourhood and internal local state. All virtual forces are calculated from range and bearing measurements to neighbouring robots, targets and virtual beacons, as measured by the individual robot. The range and bearing measurements are input into physics-based force laws, namely a split-Newtonian potential inspired law (based on Newton’s law of gravitation), and a Lennard-Jones potential inspired law, to compute the virtual forces which are then weighted to give a total force vector acing upon the individual robot, which is in turn used to compute a new displacement vector.

This method is particularly useful for self-organising entities into regular patterns (Spears et al., 2004; Morgan and Schwartz, 2005; Mullen et al., 2009b) and can be modified with relative ease to facilitate a number of tasks relevant to the problem application posed in this paper. This method of coordinated control is fully distributed, with each individual robot computing its own virtual forces experienced, with no need to compute a global force map, and no inter-robot communication required. The adopted approach is a versatile and powerful tool for implementing distributed, self-organised behaviour in a swarm of mobile robots.

4.2.1 Formation control

The basic control law yields the self-organisation of the swarm into a regular spaced repeating lattice formation. This is achieved by each individual robot measuring the range, \( r \) and bearing, \( \theta \), to any neighbouring robots \( n \in N \), within a given ‘visible range’, \( r_{vis} \). The visible range is user-set; however, there may also be physical constraints imposed depending on the sensor method used to determine the range to neighbouring robots. For formation control we use the Lennard-Jones-based potential, as it provides a more gradual change between attractive and repulsive forces, which reduces the potential robot oscillation effect around the desired robot locations. The inter-robot force, \( F_R \) experienced due to neighbouring robot \( n \) is given by:

\[
F_{Rn} = \begin{cases} 
4\epsilon \left( \frac{R}{r} \right)^6 - \left( \frac{R}{r} \right)^7 & \text{if } r \leq r_{vis} \\
0 & \text{otherwise}
\end{cases}
\]  

(1)

where \( \epsilon \) is the maximum allowed force, \( R \) is the desired separation distance between neighbouring robots, and \( \sigma \) and \( \tau \) are control parameters. The \( x \) and \( y \) components are given by:

\[
F_{Rnx} = F_{Rn} \cos(\theta_n),
\]  

(2)

\[
F_{Rny} = F_{Rn} \sin(\theta_n).
\]  

(3)
Towards autonomous robot swarms

The total $x$ and $y$ components of the force experienced are then given by:

$$F_{R_{\text{total}x}} = \sum_{n=1}^{N} F_{R_{nx}},$$  \hfill (4)

$$F_{R_{\text{total}y}} = \sum_{n=1}^{N} F_{R_{ny}}.$$  \hfill (5)

We employ a discrete-time approximation and calculate the resultant $x$ and $y$ components of the velocity as:

$$v_{x} = F_{R_{\text{total}x}} \Delta t,$$  \hfill (6)

$$v_{y} = F_{R_{\text{total}y}} \Delta t.$$  \hfill (7)

From $v_{x}$ and $v_{y}$ we calculate the new displacement vectors as:

$$\Delta x = v_{x} \Delta t,$$  \hfill (8)

$$\Delta y = v_{y} \Delta t.$$  \hfill (9)

The calculated $\Delta x$ and $\Delta y$, together with the robot’s own bearing $\phi$, are then input into a low-level motor controller to actuate the robot’s movement towards the new desired location.

4.2.2 Distributed searching

The distributed search behaviour employs the split-Newtonian-based potential, and includes an additional internal factor as a force acting upon the robot. The split-Newtonian-based potential is implemented with a more abrupt change in attractive and repulsive forces, to achieve a higher distribution response amongst the robots. The force experienced due to neighbouring robot $n$, when executing the distributed search behaviour, is given by:

$$F_{R_{n}} = \begin{cases} 
  \frac{G}{r^2} & \text{if } r \leq R \text{ and } r \leq r_{\text{vis}} \\
  -\frac{G}{r^2} & \text{if } r > R \text{ and } r \leq r_{\text{vis}} \\
  0 & \text{otherwise}
\end{cases}$$  \hfill (10)

where $G$ is the gravitational constant, which effects the rate of change in force with distance. The total $x$ and $y$ components are computed in the same way as described above.

The inclusion of the additional force is nature inspired by the foraging behaviour of ants in the wild. The basic concept is that foraging ants are more likely to continue moving in their current heading and are less likely to make turns at large angles to their current heading (Chialvo and Millonas, 1995). To mimic this behaviour we include an attractive force directly infront of the robot in addition to the forces due to neighbouring robots. The internal force is calculated in a similar way to the inter-robot
forces, assuming an attractive virtual beacon at distance $S$ directly in front of the robot, the force is calculated as:

$$F_I = -\frac{G}{S^2},$$

with the $x$ and $y$ components given by:

$$F_{Ix} = F_I \cos(\phi),$$

$$F_{Iy} = F_I \sin(\phi).$$

The $x$ and $y$ components of the total force experienced under the distributed search control law, $F_{D_{total}}$, are given by a weighted combination of $F_{R_{total}}$ and $F_I$:

$$F_{D_{total}x} = (\alpha F_{R_{total}x}) + (\rho F_{Ix}),$$

$$F_{D_{total}y} = (\alpha F_{R_{total}y}) + (\rho F_{Iy}),$$

where $\alpha$ is the inter-robot force weight and $\rho$ is the search force weight.

### 4.2.3 Surrounding targets

The control law to facilitate the behaviour of surrounding a target is again a simple extension to the basic formation control, similarly using the Lennard-Jones-based potential. The inter-robot force, $F_R$, is again calculated from equation (1). Upon detecting the target we assume the robot ascertains the location of the target relative to its own position, so as to be able to calculate the target range, $r_T$, and bearing, $\theta_T$. The robot then experiences a force due to the detected target, given by:

$$F_T = 4\epsilon \left[ \left( \frac{R_T}{r_T} \right)^\sigma - \left( \frac{R_T}{r_T} \right)^\tau \right],$$

where $R_T$ is the desired distance of the robot from the target. As previously, the $x$ and $y$ components given by:

$$F_{Tx} = F_T \cos(\theta_T),$$

$$F_{Ty} = F_T \sin(\theta_T).$$

As with the distributed search behaviour, the $x$ and $y$ components of the total force experienced under the target surround control law, $F_{S_{total}}$, are given by a weighted combination of the total inter-robot force, $F_{R_{total}}$, as calculated by equations (1)–(5), and in this case, the force due to the detected target, $F_T$:

$$F_{S_{total}x} = (\alpha F_{R_{total}x}) + (\mu F_{Tx}),$$

$$F_{S_{total}y} = (\alpha F_{R_{total}y}) + (\mu F_{Ty}),$$

where $\mu$ is the target force weight.
4.3 Robot system architecture

The system is designed not to be limited to a specific robotic platform. The above control laws are the central components to the collective behaviour of the system at the swarm level, and are applicable to most types of mobile robotic platforms.

We scale-down the scenario to enable proof of concept testing and analysis in a laboratory setting. We use the e-puck (Mondada et al., 2009) robot as our UGV platform. The e-puck is a small-scale differential drive laboratory robot approximately 7 cm in diameter. It has a number of on-board sensors including: a forward facing camera, eight IR proximity sensors, accelerometer and three microphones. It is also equipped with a speaker, an LED ring and supports Bluetooth communication.

As a safety critical fail-safe, we implement the obstacle avoidance behaviour separate from the swarm-level behaviours, as a individual behaviour with an overriding priority. We use a generic reactive obstacle avoidance method utilising the e-puck’s on-board proximity sensors (of course this would be implemented accordingly given the specifics of the particular robotic platform being used).

Figure 1 A schematic diagram of the FSM

The different behavioural controls are implemented in the form of a finite state machine (FSM) as shown in Figure 1. The state transitions in the FSM are triggered by deterministic conditions based on sensory information received from the various on-board sensors of the robot. An overview of the individual robot architecture is given in Figure 2. Whilst in the distributed search state the robots only experience forces
from their internal search beacons and any neighbouring robots within their local visible range, \( r_{\text{vis}} \), executing the control law as defined by the equations in Section 4.2.2. A given robot does not experience any force due to any target until that robot has physically detected the specific target via its on-board camera. Once a specific target has been located, its location is assumed known to the detecting robot, and that robot then enters the target surround behavioural state, at which point it now experiences a force due to the detected target, as given by equation (16), and executes the control law as defined by the equations in Section 4.2.3.

Figure 2 A schematic diagram of the robot system architecture

In Figure 2, for our chosen robotic platform, the proximity sensors are the e-puck’s on-board IR sensors. The vision is the e-puck’s on-board camera, however this could be any equivalent sensor with the ability to detect the presence of the specific targets, with the ability to provide an estimate of the target’s position relative to the robot. For the results reported in this paper we simulate the range/bearing sensors by filtering absolute known positions of all the robots in the simulation. We have, however, simulated similar experiments using an on-board IR-based range/bearing system similar to the one reported in Gutierrez et al. (2008), yielding comparative results to using filtered absolute positioning. The odometry requirements are such that each robot knows its own global bearing, knowledge which we assume the robots can obtain (for example with an on-board compass).
The coordination control laws output displacement vectors, which, for our chosen robots, need to be converted into left and right wheel velocities. We have developed a motor control law derived from the one reported in Aicardi et al. (1995) for this purpose, which provides smooth closed-loop steering for differential-drive robots towards the desired location.

4.4 Laboratory experiments

Experiments are carried out in a 3D simulation environment using the Webots professional mobile robot simulation package (Michel, 2004), which includes an accurate model of the e-puck robot. Figure 3 shows a photograph of three e-puck robots in a triangular formation in our laboratory environment, on the left, with the corresponding Webots 3D simulation on the right.

Figure 3 (a) A photograph of three e-puck robots and (b) the corresponding 3D simulation in Webots

Since the problem of the actual detection of the IEDs is beyond the scope of this paper, we simplify this process for the purpose of the experiments, and represent the IEDs as coloured markers within the environment. We then use self-developed image processing algorithms to facilitate vision-based detection of the IEDs via the robots’ on-board cameras.

Figure 4 shows an example experiment set-up. The image on top [Figure 4(a)] shows a view of the 3D simulation environment at time $t = 0$, and the image on the bottom [Figure 4(b)] details a 2D map of the same. The environment shown in Figure 4 is of a bounded rectangular shape, chosen to bear resemblance to a stretch of road in a real world scenario. We include solid obstacles to simulate any generic real world obstacle that may be encountered (such as rocks, debris), and in the case of Figure 4 there are three targets to locate.

Since the focus of this research is not concerned with the design of individual robots and their performance, we do not concern ourselves with such external issues as terrain type, gradient, atmospheric effects and so forth. Although we appreciate these as important factors when considering operational deployment of a developed system, for this current work we assume that equivalent deployable robotic platforms to the ones used in our laboratory experiments would be available. We therefore use a simple zero gradient, smooth surface as the ground within our 3D environment.
Figure 4  (a) A screen-shot of the 3D simulation environment with (b) the corresponding 2D map

Note: In the 2D map, solid dark grey circles represent robot locations, black ‘x’ markers indicate target locations and solid black squares indicate obstacle locations.

We also ignore wheel slippage in the work presented in this paper, and use a kinematics-based model to simulate the robots’ movements. All objects within the environment are simulated with bounding boxes to enable realistic simulation of any occurring collisions.

5 Results

In this section, we present and analyse results from a number of laboratory experiments designed to show the self-organising behaviour obtained from the developed control laws. Furthermore, we report the results from experiments to show the scalability, adaptability and robustness of the system. Finally, we show a case study experiment designed to simulate the specific task of target localisation and monitoring, with reference to counter IED operations.
Figure 5  Plots of the robot locations and trajectories at different time-steps during a formation experiment

Notes: Top-left: $t = 1s$; top-right: $t = 10s$; middle-left: $t = 20s$; middle-right: $t = 30s$; bottom-left: $t = 100s$; bottom-right: $t = 600s$.
Solid dark grey circles indicate robot locations, light grey lines indicate the trajectory history.

Figure 5 shows plots of the robot locations ($N = 10$) and past trajectories at different points in time during an experiment showing the self-organisation of the robot swarm from a compact deployment configuration with a common bearing $\phi = 0$, to a quasi regular spaced lattice configuration. This experiment was carried out using only the formation control law described in Section 4.2.1, with $\epsilon = 1.0$, $R = 30$ cm, $r_{vis} = 150$ cm, $\sigma = 0.1$ and $\tau = 0.05$. From Figure 2, the robots use here only the range/bearing and odometry sensors, which provide information of the range and bearing to neighbouring robots every 100 ms for the formation coordination control law. The
formation coordination and control law then calculates a new displacement vector which the motor control law converts into left and right wheel speeds to drive the robot towards the desired location. From time $t = 1\, \text{s}$ each robot manoeuvres in response to the virtual forces experienced due to the relative positions of the neighbouring robots, with the design of the control law ensuring that each robot attempts to maintain a specified separation distance $R$ from neighbouring robots. This self-organising behaviour provides a fully distributed method of creating and maintaining a regular formation pattern of mobile robots. This behaviour is a useful method for use in a range of applications, such as providing mobile sensor lattices, mobile communication lattices and autonomous payload distribution.

Figure 6 shows the average separation distance between robots versus time for the formation experiment, carried out with three, five, ten, 20 and 30 robots separately. As the gradients of the plots approach zero, the swarm has reached a stable configuration. We see from Figure 6 that the time taken to reach a stable configuration does not vary significantly with increasing numbers of robots between three and 30. This scalability is due to the distributed nature of the swarm robotics approach, and greatly improves the flexibility of the system. The key factor to the distributed nature of the system is the non-requirement for explicit inter-robot communication. The small increase in convergence time in Figure 6 is simply due to the fact that with an increasing number of robots, some of the robots may have to travel further before the swarm achieves stability.

**Figure 6** A plot showing the average separation distance between robots vs. time, for a formation experiment
Figure 7  Plots of the target location and robot locations and trajectories at different time-steps, showing adaptability and robustness to robot failure and target movement.

Notes: Top-left: $t = 1s$; top-centre: $t = 10s$; top-right: $t = 100s$; middle-left: $t = 120s$; middle-centre: $t = 125s$; middle-right: $t = 180s$; bottom-left: $t = 185s$; bottom-centre: $t = 200s$; bottom-right: $t = 300s$.

Solid dark grey circles indicate robot locations, light grey lines indicate robot trajectory history, a black ‘x’ marker indicates the target location.

Section 4.2.3 detailed the extension of the formation coordination and control law to facilitate a target surround behaviour. We present the target surround behaviour in an experiment where six robots have the goal of surrounding a target of known location. We also show the robustness and adaptability of the behaviour by removing two of the robots (to simulate total robot failure/destruction) and moving the target, at separate occasions during the experiment. The target surround behaviour was run with the following settings: $\epsilon = 1.0$, $R = 30 \text{ cm}$, $R_T = 15 \text{ cm}$, $r_{vis} = 50 \text{ cm}$, $\gamma = 0.1$, $\tau = 0.05$, $\alpha = 1.0$ and $\mu = 2.0$. Figure 7 shows plots of the robot locations and past trajectories, along with the target location, at different points in time during the experiment. At
time $t = 1s$ the robots start in a line formation approximately 90 cm North of the target. Guided by the target surround coordination and control law the robots approach the target then surround the target, forming a quasi unified perimeter at specified distance $R_T$. After the robots have surrounded the target, at time $t = 120s$ two robots are removed, forcing the swarm to reconfigure in order to maintain a quasi uniform perimeter. Once the swarm has reconfigured around the target, at time $t = 180s$ the target is moved approximately 60 cm North of the robots. The robots then manoeuvre to the target’s new location and achieve a quasi uniform surrounding perimeter by time $t = 300s$.

Figure 8 shows the average inter-robot distance and the average distance between the robots and the target, versus time, for the above experiment. Both the inter-robot and robot-target plots reach approximately zero gradient by approximately 30s. We see a slight deviation from zero gradient in both the inter-robot and robot-target separation at time $t = 120s$ where two of the robots are removed from the experiment. At time $t = 180s$ we see a large spike in the robot-target separation distance, as the target is moved, with a rapid convergence back to zero gradient at the previous separation distance, showing the fast response of the swarm to the target displacement. The inter-robot separation distance only suffers a minor deviation from zero gradient when the target is moved, as the robots remain in close proximity to one another as they move towards the new target location.

**Figure 8** A plot showing the average separation distance between robots and between all robots and the target, vs. time, for a target surround experiment.

The final experiment we report utilises the full system of the FSM (Figure 1) and robot architecture given in Figure 2. This experiment simulates the problem of searching for targets of known appearance, and once located, surrounding the targets to provide visual coverage and a physical perimeter. We again liken this simulation to a real world scenario of searching for and localising IEDs in a given area.
The experiment takes place in a bounded environment of a size 500 cm by 200 cm, with four obstacles and three targets, as shown in Figure 4. The robots deploy from a common corner of the environment, in a pseudo-random cluster, with random orientations $\phi$, and execute the FSM, starting in the distributed search state. The target surround behaviour is executed with the same parameters as above. The distributed search parameters are set-up as with the target surround experiment, above, with the additional distributed search parameters taking the values: $G = 500, S = 10$ cm, $\rho = 1.0$ and, $r_{\text{vis}} = 250$ cm when executing the distributed search behaviour.

Figure 9 gives 2D plots of the environment showing the robot positions ($N = 10$) at different points in time. From time $t = 1s$ the robots execute the distributed search behaviour, distributing from the deployment location to search the environment for targets. As the robots detect targets they change to the target surround state and manoeuvre to within a 15 cm range of the target. As more robots detect the same target they manoeuvre into a quasi regular perimeter around the target, providing multiple FOV coverage of the target and physically enclosing the target as best as possible given the number of robots. From Figure 9, we see that by time $t = 100s$ all three targets have been located, with two robots monitoring one target and one robot monitoring the other two targets, respectively. By time $t = 900s$ all of the robots have detected a target, with four robots monitoring two of the targets respectively and two robots monitoring the other target.

**Figure 9** Plots of the robot locations at different time-steps during a search and surround experiment

Notes: Top-left: $t = 1s$; top-right: $t = 100s$; bottom-left: $t = 300s$; bottom-right: $t = 900s$. Solid dark grey circles indicate robot locations, black ‘x’ markers indicate target locations, solid black squares indicate obstacle locations.

Figure 10 shows plots of the average inter-robot separation, averaged over five experiment runs, versus time for different numbers of robots. As the gradients converge to zero this corresponds to more robots surrounding targets, thus reducing the degree of
change in the inter-robot separation. We see similar convergence times for three to 15 robots, showing the scalability of the system for the given task scenario of searching and surrounding targets. We note that the lower average separation for three robots is due to the fact that only two targets were discovered when running the experiment with only three robots. The target furthest from the deployment location (Figure 9) was never located, and thus the range of the average separation only spanned between the two targets nearest the deployment location, once convergence on the targets had occurred.

**Figure 10** A plot showing the average separation distance between robots vs. time, for different numbers of robots performing a search and surround experiment

![Figure 10](image.png)

<table>
<thead>
<tr>
<th>No. robots</th>
<th>Time taken to locate (s)</th>
<th>Distribution score</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 target</td>
<td>2 targets</td>
</tr>
<tr>
<td>3</td>
<td>60.7</td>
<td>194.5</td>
</tr>
<tr>
<td>5</td>
<td>79.8</td>
<td>220.6</td>
</tr>
<tr>
<td>10</td>
<td>57.8</td>
<td>104.8</td>
</tr>
<tr>
<td>15</td>
<td>36.2</td>
<td>75.9</td>
</tr>
</tbody>
</table>

Table 1 A table showing the times taken to locate targets, and distribution scores, for the target search and surround experiment performed with different numbers of robots

Table 1 shows the average times taken for one, two and three targets to be located for different numbers of robots performing the target search and surround experiment. Also shown is a distribution score (ranging from 0 to 1, with 1 being the optimum), which is derived from the number of robots distributed around each detected target, with respect to the number of available robots (specifically, a normalised ratio of the number of robots around each target to the robot/target ratio, averaged over the total number of targets). With 15 robots all targets are located in the fastest time, with the highest distribution score. This is perhaps not surprising given the robot to target ratio, and indeed future work will include experiments with varying numbers of targets. Interestingly the results for five robots yield the slowest detection times. There are
of course other factors influencing these results, such as target positioning, obstacle size, numbers and positioning, environment size and shape and deployment locations. Again these factors will be investigated in future work, with a view to determining the optimum number of robots given the expected number of targets, the search space and other known factors.

6 Conclusions and discussion

A swarm robotics approach to multi-robot coordination and control has been developed using artificial forces as indirect communication to facilitate distributed self-organising behaviour within a group of homogeneous mobile robots. This approach was used to enable a number of swarm level behaviours, namely: formation control, distributed search and target surround. A number of case study experiments were carried out to demonstrate these behaviours and analyse the levels of scalability, adaptability and robustness to robot failure. Finally, the behaviours were merged in a FSM to tackle the problem of searching for and monitoring/enclosing multiple targets in a simulated counter IED experiment.

We have shown how simple indirect communication between individual robots can yield distributed self-organising behaviour that can be adapted to facilitate a number of useful behaviours at the swarm level, and furthermore, that a number of similar behaviours can be merged to form a simple yet effective system for tackling a number of problems in the security and defence domain.

Although the experiments in this paper could apply to a range of scenarios, we refer to the counter IED operations as this is undoubtedly an area of high priority with regards to developing new technologies with potential to reduce the levels of required direct human contact in these processes. The robustness to robot failure and potential expendability of the robots of the system presented in this paper may be particularly advantageous for tackling the problem of countering IEDs, given the destructive nature of these devices.

Although the experiments presented in this paper were solely simulation based, we are in the process of performing similar experiments on real e-puck robots in a purpose built laboratory enclosure with an overhead tracking system to analyse the robots movements and provide real-time robot pose information where necessary. Additional future work will involve analysing the safety critical aspects of the system with respect to the reliability of the robots sensors, and the impact on the decisions made within the FSM.

Acknowledgements

The authors would like to thank the reviewers for their constructive comments which helped to improve the quality of the final paper. The authors also thank the editor of this journal.
References


Towards autonomous robot swarms


