Adaptive Behavior
http://adb.sagepub.com/

A Distributed Feedback Mechanism to Regulate Wall Construction by a Robotic Swarm

Robert L. Stewart and R. Andrew Russell
Adaptive Behavior 2006 14: 21
DOI: 10.1177/105971230601400104

Adaptive **Behavior**

A Distributed Feedback Mechanism to Regulate Wall Construction by a Robotic Swarm

Robert L. Stewart, R. Andrew Russell
Centre for Perceptive and Intelligent Machines in Complex Environments: Intelligent Robotics, Monash University, Australia

Nest construction is an impressive achievement for social insects. They employ templates (or patterns) in the environment to guide construction tasks and robot swarms can use similar mechanisms. Early work examined how a spatio-temporal varying template can be utilized by a swarm of minimalist robots to build the framework for a linear wall. From this, it became clear that a method for regulating construction in a variable environment was needed. To this end, a social insect inspired distributed feedback mechanism is proposed and verified. Direct and indirect transfer of information between individuals, through the use of signals and cues, is shown to facilitate closure of a feedback loop allowing the system to operate close to optimally. Perturbation experiments demonstrate the robotic swarm's ability to adapt to environmental changes. The work described in this paper provides the basis for a distributed robotic system capable of constructing any given planar structure—a goal implicit in many of the suggested applications for collective construction.

Keywords swarm robotics · multi-agent adaption · distributed feedback · collective construction · template

1 Introduction

1.1 Motivation

Swarm robotics has been loosely defined as "the study of how collectively intelligent behavior can emerge from [the] local interactions of a large number of relatively simple physically embodied agents" (Dorigo & Şahin, 2004). Typical problem domains for the study of swarm-based robotic systems include foraging, box-pushing, clustering, sorting, and formation forming (Krieger, Billeter & Keller, 2000; Kube & Zhang, 1997; Beckers, Holland, & Deneubourg, 1994; Holland & Melhuish, 1999; Spears, Spears, Hamann, & Heil, 2004). Swarm robotic systems are often seen to display many of the attributes typical of collective (or swarm) intelligent systems in general. These include robustness, distributedness, adaptability, flexibility, self-organization, and minimalism (Bonabeau, Dorigo, & Theraulaz, 1999; Kurabayashi, 1999). These attributes can often make a swarm-robot solution preferable to a single robot solution (Holland, 1996).

Another problem domain is that of collective construction and recently there has been renewed interest in this area. Collective construction tasks involve multiple robots assembling or depositing building material at discernible and globally relevant spatial locations. Through this process, the structures built from the building material form patterns, observable from a

Correspondence to: Robert Stewart, Centre for Perceptive and Intelligent Machines in Complex Environments: Intelligent Robotics, Monash University, Clayton campus, Wellington Road, Clayton, Victoria 3800, Australia. *E-mail*: robert.stewart@eng.monash.edu.au
Tel.: + 61 3 9905 5715, *Fax*: + 61 3 9905 3454 (general office)

(2006), Vol 14(1): 21–51.
[1059–7123(200612) 14:1; 21–51; 063329]

global vantage point. For coherent patterns to emerge, individuals in the robot swarm must coordinate their construction activities in both space and time. The question of how best to do this should be motivated by the type of structures we would like robot swarms to build. For this reason it is beneficial to consider the potential applications of research in this area.

Many applications have been envisaged for robot swarms capable of collective construction. Such robots could build retaining walls around radioactive materials or gas leaks, construct levee banks to restrain floodwaters, build roads and irrigation canals, or perhaps even build replacement homes for people in areas hit by disaster. Sending robots to the moon or on interplanetary missions, to establish foundations and infrastructure before human arrival, has also been proposed (Brooks, Maes, Matarić, & More, 1990). With the miniaturization of robots equipped with construction capabilities (Melhuish, 2000), it is likely that still further possibilities may arise. For example, micro or nano robots could repair tissue in human bodies or assemble molecules into microscopic structures. What is common to many of these predicted applications is the often implicit desire for robot swarms to build structures conforming to a design of our choosing. For this reason, we consider it useful to pose a *generalized two-dimensional (2D) collective construction problem*, with the hope that it (and solutions to it) may later be extended to three dimensions:

> *How might a robot collective be designed such that, when provided with adequate building material, the robots are capable of loosely constructing any given planar structure?*

It is worth noting that the above problem is analogous to the (pattern formation) "inverse problem" posed by Bonabeau (1997).

The work detailed in this current paper describes a feedback mechanism that will help facilitate a solution to this generalized 2D collective construction problem. The remainder of this paper is divided into a number of sections. Firstly, we review work related to collective construction, highlighting the benefits and limitations of proposed systems (Section 1.2). Our own previous work on the subject is then reintroduced (Section 1.3). It is made clear that to proceed further with the development of our system there is a need for feedback within the swarm. Implementing a feedback system then provides the focus for much of the remainder of the paper. The approach taken is inspired by feedback mechanisms found in social insects (Section 1.4).

After describing the experimental procedure (Section 2), the need for feedback is emphasized in a series of trials that study the effect of a key parameter (latency time) in an environment that was not changing significantly (Section 3). Then, drawing on techniques used by social insects, a design for a cue-signal based distributed feedback mechanism is proposed that allows for an adaptable choice of latency time (Section 4). This introduces a fundamental trade-off between information quality and speed which is investigated in more detail (Section 4.1). Information flow throughout the system is then analyzed explicitly by studying the results obtained in one particular trial (Section 4.2). Following this, two perturbation trials are then detailed that demonstrate the swarm's ability to respond to changes in the environment (Section 4.3). Factors influencing the results are then highlighted in Section 5 before an overall discussion in Section 6 and directions of future research are outlined in Section 7. Finally some concluding remarks are made (Section 8).

1.2 Related Works

The notion of having swarms of robots perform collective construction tasks is not, in itself, novel. In 1990, Brooks et al. suggested that multiple robots, employing simple rules, might be able to level the ground at a lunar construction site. However, it is only recently that the feasibility of ideas such as this have been demonstrated in practice. Parker, Zhang, and Kube (2003), in a technique inspired by blind bulldozing in ants (Franks, Wilby, Silverman, & Tofts, 1992), showed that simple bulldozer robots could clear rubble from an area to form a "nest," approximately circular in shape. To do this, the robots needed only a sensor, to detect the force exerted by rubble on the robot's plough, and a contact sensor, to detect collisions with other robots.

A similar "minimalist" solution to another construction problem was demonstrated by Melhuish, Welsby and Edwards (1999). Linear wall building, again inspired by ants, was found to be possible when robots utilized templates or patterns already present in the environment. Two templates were used by the robots to build their wall. The first was created by a linear strip of white tape pre-placed on the floor of the test arena.

The robots used this heterogeneity by depositing building material at a certain distance from it. To achieve orientation during this task, the robots used a second template provided by a pre-placed bank of halogen lights.

Wawerla, Sukhatme, and Matarić (2002) have also demonstrated linear wall construction. In their work, blocks of alternating color were placed by robots along a straight line starting with a pre-placed seed block located underneath a beacon. Each robot was equipped with a laser range finder and was able to find the beacon and use it for orientation during construction. This work investigated the effect a small exchange of information (namely, the color of the next required block) between robots had on the efficiency of the system. While the structure was not very complicated, coordination was still required since blocks had to be placed in a prescribed order.

Werfel (2004) proposed, and demonstrated in simulation, a method by which multiple robots could build arbitrary structures with non-crossing walls. A leader robot acted as a stationary beacon, around which the structure was built. The leader also provided information about the building of this structure. Some robots were instructed to take on the role of becoming a "corner." This involved them parking themselves at critical spatial positions relative to the beacon. Other robots then built linear or curved walls between the corners. This system is less minimalist in nature since the robots are required to use radio communications to share information and must maintain a private coordinate system in order to localize themselves.

One of the mechanisms thought to be used by social insects in building structures is stigmergy. This term was originally introduced by Grassé (1959) to explain the coordination and regulation of construction activities in termites. Holland and Melhuish (1999) express this phenomenon succinctly. "In Grassé's vision," they say, "a worker deposits a piece of building material (does 'work') in a particular location; this changes the sensory input subsequently obtained at that location and hence may change the behavior produced (and the work done) at that location in the future." In other words stigmergy is the "guidance of work performed by social insects through the evidences of work previously accomplished" (Wilson, 1975). In this process, called stigmergic communication, insects indirectly communicate with one another through the environment. In this paper we follow Wilson's view that stigmergic communication is perhaps best regarded as a subset of a more general form of indirect communication called sematectonic communication (note, however, that this distinction is not usually made by other researchers). Here sematectonic refers to "any form of behavior or physiological change [produced] by the evidences of work performed by other animals" (Wilson, 1975). For stigmergy, this behavioral or physiological change is manifest in "the guidance of additional work" (Wilson, 1975).

Stigmergy can take two forms depending upon the type of stimuli builders respond to. When builders respond to a quantitative stimulus of some intensity and then transform it (through work) "into the same stimulus [but] with a higher intensity" the process is called quantitative stigmergy (Cazamine et al., 2001). For example, in termite construction, building material is deposited that emits a cement pheromone. This acts to attract further deposits, thereby increasing the pheromone intensity and focusing "building effort on 'hot spots'" (Ladley & Bullock, 2005). Qualitative (or discrete) stigmergy on the other hand, is when "the stimuli differ from each other qualitatively" and can "elicit different responses" (Anderson, 2002). Through qualitative stigmergy there is the potential for complex structures to be generated.

Theraulaz and Bonabeau (1995) demonstrated this concept of qualitative stigmergy in a computer simulation model in which artificial agents were able to build three-dimensional structures, some of which resembled those built by real wasps. In the simulations, artificial agents randomly wandered between empty cells on a three-dimensional lattice. These agents were programmed to place one of two types of brick in a cell when the state of the neighboring cells corresponded to one of a certain number of stimulating configurations. It was found that coherent structures could be constructed by an appropriate choice of stimulating configurations and these together constituted what was called a coordinated algorithm. After applying a Factorial Correspondence Analysis, it was concluded that "coordinated algorithms produce relatively stable, coherent shapes belonging to a small, compact subspace of the space of all possible architectures, and that non-coordinated algorithms generate unstable, structureless shapes" (Theraulaz & Bonabeau, 1995).

All of the coherent structures obtained were observed to display signs of modularity. As suggested by Bonabeau, Theraulaz, and Cogne (1998), this modularity may have been a natural consequence of the assembly process and/or due to exogenous factors. For a few structures the results were deterministic in that simulation trials always produced exactly the same architectures. These deterministic architectures resulted "through the implicit handshakes and interlocks" at every building stage (Theraulaz & Bonabeau, 1995). However, in general, the same set of stimulating configurations generated similar yet physically different structures on separate simulation runs.

It is possible to apply a stigmergic approach in trying to solve the generalized collective construction problem. The problem demands a deterministic final structure as well as the ability to generate structures that are not necessarily modular. In the simulations discussed above, qualitative stigmergy was only shown to generate modular structures, and, in most of these cases, the structures generated were not deterministic. However, by increasing the number of different types of bricks available, the space of possible architectures can be vastly increased to encompass any given architecture. Here modularity is still likely to be present, but it may occur on a scale too large to be observed. Whilst allowing for the use of many different types of bricks is potentially limiting, a number of researchers have still followed this approach and these are discussed below.

Recently, Howsman, O'Neil, and Craft (2004) have proposed an approach for building assemblies in outer space using stigmergy. In the simulation examples they gave, a number of potentially useful space structures were built. Twenty different colored blocks were allowed, giving sufficient scope to generate the structures of interest. Whilst this research was directed towards some specific applications, other works have focused on developing techniques for decomposing more general shaped structures into sets of local rules.

Jones and Matarić (2003) demonstrated through simulation how a large class of two-dimensional goal structures could be built by intelligent self-assembling agents. In their work, the assembly agents themselves were the building material,[1] connecting to one another according to a "transition rule set." Agents (equivalent to bricks) were distinguishable by their state value and the transition rules determined the local conditions under which agents would connect themselves to other agents. An algorithm was developed whereby a consistent transition rule set could be developed for a certain class of structures that were all connected. To allow disconnected structures to be built, the work was extended by Li and Zhang (2005). They suggested the use of multiple seeds as a way of resolving problems with the earlier technique. Evident in both works, however, was the need for a large number of different agent states (equivalent to distinguishable bricks). In one example, for a structure composed of 245 agents, over one hundred different agent states were required. It would seem therefore, that multiple brick states is one possible precondition for allowing certain structures to be generated. Perhaps another, less feasible, precondition would be to increase the perceptual range of the robots.

In yet another approach, Werfel, Bar-Yam, and Nagpal (2005) have described a system (again simulated) in which robots construct structures from prefabricated intelligent building blocks, which may be functionally different. When connected, the building blocks have the ability to communicate with one another via a physical interface. Each block is also able to communicate at short range with adjacent robots. To build a given structure, a seed block (which also acts as a beacon) is first required. The construction process then proceeds in a relatively straightforward manner. Block-carrying robots follow the perimeter of the formative structure and ask adjacent previously placed blocks if they can make a deposit. This may depend on the block type they are carrying. Each block attached to the structure maintains a map and is aware of the current structure and the desired final structure. When it is first attached, a block is given its own position in a common coordinate system. Based on the geometric and functional constraints of the desired structure, a block can authorize an adjacent robot to make a deposit. Using this technique, structures with perimeters of any shape can be built. However, these structures have to be connected and cannot have closed internal cavities.

1.3 Previous Work

1.3.1 A New Approach to Collective Construction

The work detailed in the current article develops the basis for a new technique for solving the generalized 2D collective construction problem. Emphasis will be placed on achieving total system minimalism. As such, all building blocks will be homogeneous and will not carry additional state information, radio communication between robots will not be allowed, a global coordinate system will not be available, and robots will not be required to maintain a map. Because simulation experiments may neglect issues associated with developing real robot systems, all experiments detailed in this article have been conducted with real robots, thereby validating the approach taken. It is the intention of the work to draw inspiration from the abilities of social insects. One of the simplest mechanisms used by social insects during construction tasks is the template mechanism. The extent to which robots can use templates to solve generalized construction problems is another motivation for this project.

Of the above works cited (Section 1.2), our work is most similar to that of Melhuish et al. (1999) who also used minimalist robots and templates. However, instead of the templates taking the form of spatially and time-invariant pre-placed heterogeneities, the swarm in this article creates its own template that can vary in space and with time. The template is in the form of a gradient and individuals use the quantitative information contained in it to facilitate construction. As with Werfel (2004) we also have one key individual who does not participate in the construction itself. However, unlike the leader robot in that work, which provided explicit and complex information to builders, the purpose of our key individual is only to establish a light-field gradient in the environment and to move it over time. In the next section we describe our previous work and demonstrate that to proceed further with the template system, there is a need to introduce a feedback mechanism. This then helps form the basis for a solution to the generalized 2D collective construction problem posed at the beginning of this article.

1.3.2 Pre-existing Environmental Templates Of the mechanisms thought to be employed by social insects during construction tasks, namely templates, stigmergy, self-organization, and self-assembly, the template mechanism is one of the simplest (Theraulaz, Bonabeau, & Deneubourg, 1999). Often taking the form of chemical, temperature, humidity, and light heterogeneities, templates are patterns in the environment that can be used by insects to guide their building activities (Theraulaz, Gautrais, Camazine, & Deneubourg, 2003). With a template the "blueprint of the nest 'already exists' in the environment" (Theraulaz et al., 2003).

The pheromone produced by a *Macrotermes subhyalinus* queen establishes a template in the environment in the form of a chemical gradient (Bruinsma, 1979; Theraulaz et al., 1999). Worker termites are "programmed" to deposit building material within a window of chemical concentration resulting in the construction of a royal chamber around the queen. As the pregnant queen grows in size, the chemical template changes and the chamber is enlarged to accommodate her. As another example, the ants *Leptothorax albipennis*, construct a doughnut-shaped wall around their brood/adult ant cluster using grains of sand, earth particles or fragments of stone. It is likely that a physical template is used during this task where workers walk a certain distance from the cluster before making a deposit (Franks & Deneubourg, 1997; Franks et al., 1992).

Templates can also be used by robots to guide them in construction tasks. In previous work this was verified when a robot used a template to construct doughnut-shaped structures (Stewart & Russell, 2003). Experiments were undertaken involving the use of a light source that created a light-field template. The illuminance was found to be a monotonically decreasing function of distance from the light source—following the inverse square law and the cosine law of illuminance (Born & Wolf, 1964). A minimalist robot equipped with directional and absolute light sensors, as well as a gripper, was programmed to deposit building blocks within a window of illuminance (similar to the concentration window used by the termites, *M. subhyalinus*).

It was found that the simple robot could make use of the template to construct doughnut-shaped structures. By varying the window limits the placement of the emergent structure's wall and its thickness could be varied. Figure 1 shows a typical structure being built using this technique. It resembles the nest structure built by the ants (*L. albipennis*) mentioned earlier. As a result of the study it was found that robots can make use of complexity inherent in their environment, in the form of spatially varying templates, to construct relatively complex structures without the use of global knowledge, a blueprint or even a map.

1.3.3 Self-Generated Spatio-temporal Varying Templates Having to rely on templates already present in an environment may restrict the complexity of structures that can be built. To overcome this, the previous work was extended by an investigation into the use of spatio-temporal varying templates for wall building tasks (Stewart & Russell, 2004). The hope was that templates of this sort, which are not confined in space

Figure 1 A doughnut-shaped wall being constructed by a minimalist robot.

and can vary with time, might also be useful in solving the generalized 2D collective construction problem. However, as an initial investigation to establish a basis for this future work, a simple wall construction task was chosen. A 4-member robotic swarm consisting of an organizer robot and three builder robots was used in the work.

We consider this multi-robot system to be a robot swarm according to criteria proposed by Dorigo and Şahin (2004). These criteria have been introduced to determine the degree to which a study may be called "swarm robotic." The criteria according to Dorigo & Şahin (2004) are:

(i) The study should be relevant for the coordination of large numbers of robots;
(ii) the robotic system being studied should consist of relatively few homogenous groups of robots, and the number of robots in each group should be large;
(iii) the robots being used in the study should have difficulties in carrying out or completing the considered tasks on their own, and their performance should improve when they cooperate;
(iv) the robots being used in the study should only have local and limited sensing and communication abilities.

The system introduced below (and the extended system detailed in this paper) are considered to meet criteria (i), (iii), and (iv), as well as the first part of criterion (ii). It is envisioned that future implementations of this system might have the swarm consisting of numerous organizer robots and builders working concurrently. Under these conditions, criterion (ii) would be met in full. Since the system meets the majority of the criteria well, it is considered appropriate to refer to this system as a robot swarm.

Social insect terminology can be used to further classify the system. In a social insect colony individuals may organize themselves into "functional adaptive units" ("parts") to solve tasks of a complex nature (Anderson & McShea, 2001). One example is the formation of a team where individuals combine to perform two or more subtasks concurrently (Anderson, Franks & McShea, 2001). Individuals may also organize themselves within functional units. Robson and Traniello (1999) suggest that while "studies of collective behavior minimise the role of individual behavioral variance," specialization within temporal and physical castes may be more widespread than first thought. Three categories of key individuals have been identified including (i) Catalysts who "stimulate greater activity in group members," (ii) Performers who "alone complete a task," and (iii) Organizers who "are required for a task to be completed, but … do not carry out the work themselves."

According to this social insect terminology, we classify our multi-robot system as a robot swarm composed of one functional unit performing the team task of collective construction. One robot has been nominated as an organizer since it does not contribute physically to wall building but it does perform a crucial subtask within the team: namely, providing a template. Builder robots constitute the remainder of the team and undertake the other subtasks that include foraging for, and then depositing, building material. Having clarified the terminology regarding the system, we now turn our attention towards the specifics of its operation.

A light source was mounted on a mobile organizer robot and arranged such that it emitted light only in a light beam projected over an angle of 30 degrees. The situation is depicted in Figure 2. Within this light beam the light intensity was a monotonically decreasing function of distance. With this setup the robotic swarm did not have to rely on an externally provided template but rather was able to create its own. Mounting the light source on a mobile robot also allowed the constraint that the template was fixed in space and time to be lifted. Since the organizer robot was mobile

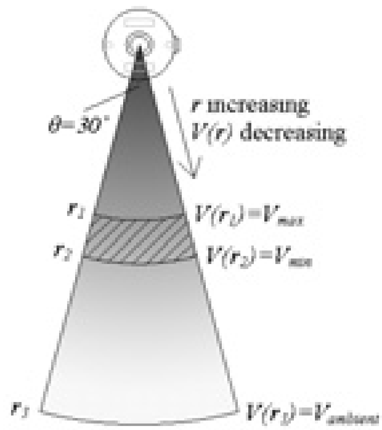


Figure 2 A depiction of the light beam created by a light source mounted on the organizer robot. The hatched region indicates the deposition window.

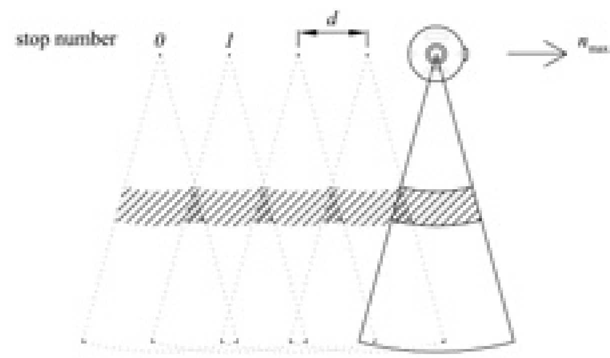


Figure 3 The organizer robot moves in a straight line from stops 0 to (n_{max} – 1), spending the latency time at each, before moving to the n_{max}th stop at which point the trial is deemed complete. Stops are separated by a distance, *d*.

and free to move, the light pattern that it generated has been classed as a spatio-temporal varying template. In a similar manner to the previous study, builder robots were programmed to deposit building blocks when they were within a window of intensity (equivalent to a window of light sensor voltages $V_{min} \leq V \leq V_{max}$) called the deposition window. This is the hatched region of Figure 2. Throughout this report the term *beam* is used to refer to that portion of the light beam defined by $V_{ambient} \leq V \leq V_{max}$.

By varying the organizer robot's motion, it was expected that builder robots could be made to deposit building blocks at certain locations in space. This was verified in an experiment in which the light-producing organizer robot drove a discrete number of steps (namely 15) in a straight line (Figure 3). The distance between stops was chosen to be d = 0.097 m so that there was a large overlap between the deposition windows. The time of arrival, A_n, at some stop number, n, was given by

$$A_n = nL + nT \quad (1)$$

where L is the latency time (time spent at each stop) and T is the transition time (time spent moving between stops). As can be seen in Figure 3, the path traced out by the deposition window falls along a straight line so that, provided the latency time (L) was chosen to be sufficiently high, a loose straight wall was able to be constructed. This work verified the concept of a spatio-temporal varying template and provides the means for the construction of more complex structures.

Before investigating how more complex structures might be built and how templates might be used to solve the generalized 2D collective construction problem, there is one significant shortcoming in this system that needed addressing. Namely, the latency time, L, must be found manually through trial and error. It is clear therefore, that an alternative and autonomous approach for determining an appropriate latency time must be found.

The choice of latency time is related to the rate of deposits, which is likely to depend on many interrelated factors. These include swarm size, block availability, test area dimensions, beam size and orientation, deposition window size and position, robot behaviors, and obstacles in the environment including the structure itself. One alternative could be to attempt to model all of the variables in the system in order to predict the rate of deposits and required latency time. However, given the inherent complexity of the system and the uncertainties associated with it, this approach may not be feasible.

To add generality to this work the robotic swarm should be able to operate in different and possibly changing environments. Such environments may require the organizer robot to select a different latency time for each stop. Ideally an adaptable choice of latency time could be adopted based on the changing characteristics of the environment. This solution may be possible if feedback about the current state of the environment could be provided to the organizer robot. The use of feedback mechanisms are widespread in the social insects and this serves as inspiration for the design of the distributed feedback system detailed in this paper.

1.4 Feedback Mechanisms in Social Insects

Social insect colonies consisting of a few to tens of millions of individuals (Jeanne, 1999) operate in dynamically changing environments. By adapting to changes in the environment, the needs of the colony are able to be met. Individuals make decisions based on information they receive directly from other individuals or indirectly, through the environment. It is through the combined and often different actions of these individuals that the colony is able to balance competing demands in an efficient and reliable way. It would seem, therefore, that the success of these colonies lies in the effective use of information and in the design of effective "information pathways" (Pratt, 1998). Often information is used in a way that facilitates the closure of feedback loops. Feedback provides the means of regulating processes within the colony.

Facilitated by information pathways, feedback can be positive or negative, fast or slow (Theraulaz et al., 1999; Wilson & Hölldobler, 1988). Feedback allows colonies to respond to changes in the environment just as an engineered control system is able to respond to input changes (DiStefano, Stubberud, & Williams, 1995; Dutton, Thompson, & Barraclough, 1997). Positive feedback "is a mechanism that promotes change in a system; moreover, it drives change in the same direction as a perturbation" (Anderson, 2002). On the other hand, negative feedback helps to stabilise a collective pattern and counterbalances positive feedback (Theraulaz et al., 1999).

One example of feedback mechanisms at work is in the regulation of worker numbers collecting and unloading nectar in a honeybee colony (Seeley, 1989). Here the aim is to match the colony's rate of nectar collecting with its capacity for nectar processing (Seeley & Tovey, 1994). It has been shown that the time spent searching for a food-storer bee by a returning forager looking to off-load nectar is an accurate indicator of the relative rates of nectar collecting and processing. If the search time is long, foragers perform tremble dances causing additional bees to become food storers. If, on the other hand, the search time is short, foragers perform waggle dances thereby recruiting more foragers (Anderson & Ratnieks, 1999b). Search times as well as wait times are queuing delays that are also used by other species of social insects to assist task partitioning (Anderson & Ratnieks, 1999a). Such cues appear to be an effective means of collecting information.

Often information transferred between individuals is, as stated by Anderson and Ratnieks (1999b), "the key factor that binds individuals together as an adaptive unit." This information takes form in cues and signals and "flows among the parts of the whole" (Franks, 1999). A signal is a "deliberate act of communication that has been shaped by natural selection" whereas a cue is a "structure or behavior which conveys information … incidentally" (Anderson & Ratnieks, 1999b). In the previous example, search time to find a food storer bee is a cue and the waggle and tremble dances performed by forager bees are signals. Further examples abound where social insects use cues and/or signals to regulate processes. These include nest construction by the paper wasp, *Polistes* (Downing & Jeanne, 1988), house-hunting by honeybees and the ant *L. albipennis* (Franks, Pratt, Mallon, Britton, & Sumpter, 2002), recruitment to food supplies by ants (Cassill, 2003), and the construction of pillars by termites (Bruinsma, 1979). In each case the use of information (in the form of cues and signals) facilitates the closure of feedback loops that regulate a process often critical to the survival of the colony.

"A colony's survival and reproduction depend on how well it can match its efforts to the challenges of a variable environment" (Gordon, 1996). As has just been discussed, feedback loops play a critical role in this adaptive ability. Feedback is often distributed, in the sense that many workers may have to act in order to bring about a change at the colony level. In some cases individuals sample sources of information several times to increase accuracy, incurring only minimal additional cost for doing so (Anderson & Ratnieks, 1999a). In other cases individuals "tap into socially pooled information that effectively has already been collated" (Franks, 1999).

There exists, though, a fundamental compromise in decision making between speed and accuracy (Franks et al., 2002). Unfortunately this will always be the case when colonies have to operate in varying environments without the certainty of global information or common knowledge. The most that can be hoped for is that the colony is able to make the best choice it can given its limited resources and lack of control over external disturbances. From the examples given, it is clear that insect societies have employed simple yet sophisticated ways of adapting to changes in the environment whilst always seeking to meet the needs of the colony.

Despite the fact that individuals may work at cross purposes (Pratt, 1998; Wilson, 1975) with incorrect decisions often being made at the level of the individual, a favorable outcome for the colony is able to emerge. It is a process that has been likened to a genetically prescribed republic (Wilson & Hölldobler, 1988) where individuals effectively vote through their choice of action and campaign to influence the decisions taken by other individuals.

From the studies on feedback mechanisms in social insect colonies several important ideas emerge. Namely,

(i) many individuals may influence a colony's response to changes; that is, the feedback is distributed;
(ii) individuals may make poor decisions; however, collective decisions are well judged;
(iii) individuals may modify their actions based on different cues and signals;
(iv) actions undertaken by individuals, directly (through signals) or indirectly (through cues) modify the actions of other individuals;
(v) there is a fundamental compromise in decision making between speed and accuracy.

Mindful of these ideas, a distributed feedback mechanism for a robotic swarm was designed and is detailed in this paper. Here the purpose of feedback is to allow for an adaptive choice of latency time to be made such that the system is able to work in different and possibly changing environments. In the next section we discuss the experimental procedure for the trials that were conducted. Following this, we will return to investigate the design of the feedback mechanism.

2 Experimental Procedure

2.1 Hardware

The mobile robots used in the experiments detailed in this paper form a 5-member robotic swarm called *The Robotermites* (Figure 4). The robots were designed following a minimalist philosophy. They each possessed a forward-facing array of infra-red proximity sensors that covered approximately 180º of the base perimeter and a light-sensor ring (containing 24 light sensors with a coverage of 360º) that could give the direction of greatest or least light intensity. Additionally, builder robots also carried an absolute light sensor (a light-dependent resistor covered by a diffusive Ping-Pong ball), a flash bulb for producing high intensity bursts of light and a gripper for picking up building blocks. The gripper was equipped with a sensor to indicate whether a block was currently being held. The organizer robot did not have these additional features but instead car-

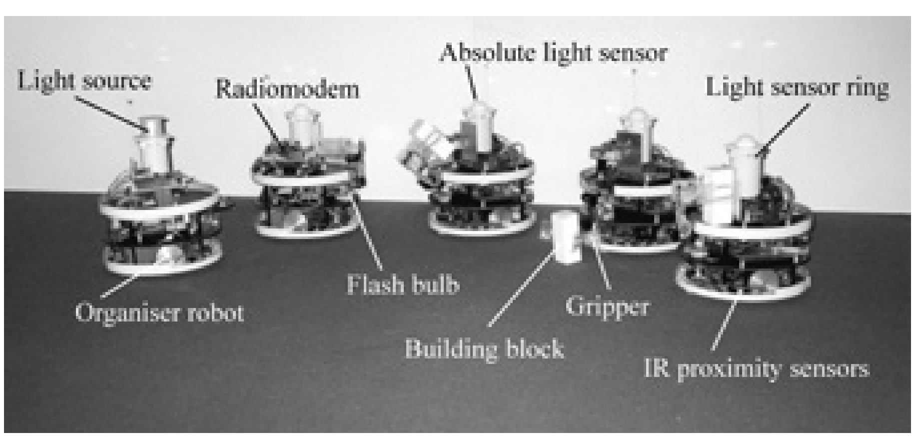


Figure 4 The robotic swarm called *The Robotermites*. The 5-member swarm consists of an organizer robot and 4 builder robots. A light source mounted on top of the organizer robot creates a light-field template.

ried a light source capable of radiating light in an arc of 30º (as in Figure 2). Each robot was controlled by an onboard microcontroller that was interfaced to each sensor and actuator. To facilitate data-logging and initialization, each robot also had a radiomodem which allowed information to be sent to and from an off-field computer.

2.2 Setup

Building blocks made from polystyrene (approximately 10 × 3.8 × 3.8 cm), weighted to reduce the likelihood of them falling over, were placed in two layers around the perimeter of a test area (Figure 5). At the start of each experiment the robots were arranged in the test area approximately as shown in Figure 5. This starting configuration was arbitrarily chosen and so as not to bias the results, it was kept the same for all trials. Each robot began the trial in a state of foraging for a block. A low-light level camera mounted overhead with a wide-angle lens allowed somewhat distorted video footage of the trials to be recorded and, from this, images were extracted (the appendix shows images of the walls constructed during the trials). The room lights were switched off for each trial and during calibration.

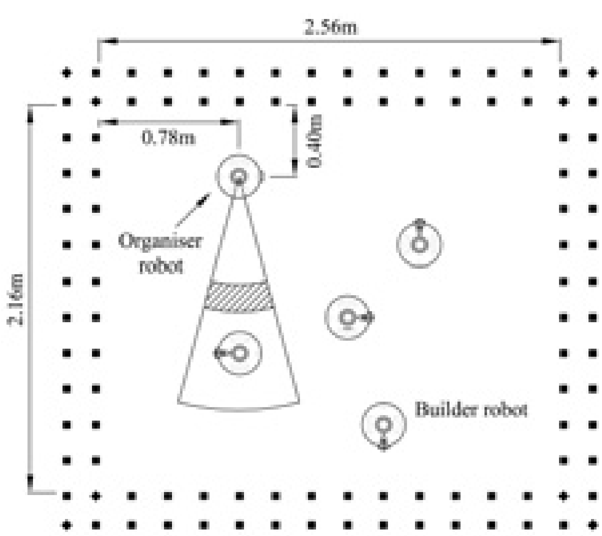


Figure 5 The experimental set-up showing the approximate placement of the robots at the beginning of each trial. Two layers of building blocks form the perimeter of the test area. In this figure, the organizer robot's direction of movement is to the right.

2.3 Calibration

Each robot required calibration. The calibration involved placing the organizer robot in the darkened test area with its light source switched on. Each builder robot was then placed in the organizer robot's beam at the distances r_1, r_2, and r_3 (Figure 2). The distances chosen for all trials in this paper were $r_1 = 0.60$ m, $r_2 = 0.75$ m, and $r_3 = 1.3$ m. At each of these locations the light level sensed by each builder robot's absolute light sensor was recorded for use.

2.4 Experimental Trials

The light-levels obtained during calibration were stored in the memory of each builder robot. Each trial was initiated by an observer via the off-field computer. The off-field computer also served as a data logger, time-stamping the completion of important events by individual robots. During trials, blocks taken by robots from the outermost layer were replaced manually by an observer and where the robots pushed the blocks aside creating a gap in the outer layer, blocks were also added by an observer to prevent the robots from leaving the test area. In all trials the organizer robot moved along an approximately straight line. The situation is similar to that depicted in Figure 3. Unlike previous work (Stewart & Russell, 2004), the distance between stops was chosen so that the deposition windows did not overlap markedly. In all trials the distance between stops was 0.31 m.

3 Latency Time

As was mentioned earlier (Section 1.3.3) the choice of an appropriate latency time is difficult. The latency time must be sufficient to allow enough blocks to be deposited in each deposition window so that a full wall is constructed. There are potentially many factors affecting the rate of deposits (some of which were mentioned in Section 1.3.3). In previous work, a choice of latency time was made through trial and error to give a good compromise between total trial time and the number of blocks deposited. It was suggested that "by increasing latency time the total number of blocks deposited would increase until some certain limit determined by the size of the deposition window is reached" (Stewart & Russell, 2004). The situation resulting from this

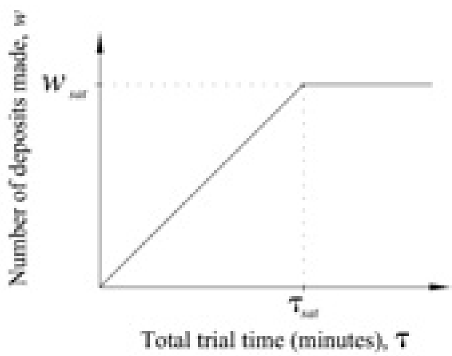


Figure 6 A model that shows how the total number of deposits made increases with total trial time until some certain limit is reached.

hypothesis is represented by the model, (2), and is shown in graphical form in Figure 6.

Equation 2 describes the total number of deposits made during a trial, w, as a function of τ, the total trial time:

$$w(\tau) = \left\{ \begin{array}{ll} \frac{w_{sat}}{\tau_{sat}} \cdot \tau & 0 \le \tau \le \tau_{sat} \\ w_{sat} & \tau > \tau_{sat} \end{array} \right\} \quad (2)$$

The total trial time, τ, is the time of arrival at the last stop, n_{max}. It can be calculated from (1) as

$$\tau = A_{n_{max}} = n_{max}L + n_{max}T \quad (3)$$

In most cases the latency time, L, is much greater than the transition time, T, and hence the total trial time can be thought of as being roughly proportional to latency time.

3.1 Effect of Varying Latency Time (Trials 1–9)

In order to confirm the model, a series of trials were devised to investigate the effect of latency time on the total number of deposits made. The robots were each programmed with a simple rule set (Tables 1 and 3) that governed which of its elementary behaviors to perform. The rule set for the builder robots is relatively simple. The main objective is for builder robots to deposit building blocks when they are in the deposition window ($V_{min} \le V \le V_{max}$). Rule 4 (Table 3) stipulates that a builder robot may produce a flash of high intensity light under certain circumstances. In later trials the flashes provide feedback to the organizer robot. However, in the current set of trials the organizer robot does not detect these flashes. Rule 4 was, however, executed by builder robots in these trials to ensure later trials were commensurate. In Tables 1–3 there are rules that require the robot to drive forward. The distance traveled when driving forward is dependant on the situation and has not been stated unless it was deemed important. Sensors are sampled to give the information required in these rules. Note that in Tables 1–3, the rules are ordered with those appearing earlier receiving higher precedence.

Details of some of the more complex robot behaviors used in Table 3 are given below.

(i) Attempt to pick up object {A} In order to try and pick up an object (assumed to be a block), a robot must first align itself so that the block is located beneath the gripper. To do this the robot samples

Table 1 The rule set used by the organizer robot in Trials 1–9.

1. **If** (at the 6^{th} stop) **Then** (turn off light bulb)
2. **If** (robot has spent L mins at current stop) **Then** (drive forwards 0.31m)
3. Wait for L minutes to elapse

Table 2 The modified rule set used by the organizer robot in Trials 10–21.

1. **If** (at the 6^{th} stop) **Then** (turn off light bulb)
2. **If** (*flash_counter_max* flashes have been detected at the current stop) **Then** (drive forwards 0.31m)
3. Count flash signals

Table 3 The rule set used by the builder robots in all trials.

1. **If** (not holding block and not in beam and object detected) **Then** (attempt to pick up object {**A**})
2. **If** (holding block and in deposition window) **Then** (deposit block {**B**} behind another block or at inner window limit and reset frustration counter)
3. **If** (holding block and have just arrived in beam) **Then** (drive further into beam avoiding obstacles {**D**} if detected)
4. **If** (frustration counter equals 2) **Then** (flash light and reset frustration counter)
5. **If** (holding block and in beam and obstacle detected) **Then** (increment frustration counter, turn randomly ±135º and then drive forwards)
6. **If** (holding block and in beam) **Then** (drive up beam towards light {**C**})
7. **If** (obstacle detected) **Then** avoid obstacle {**D**}
8. Drive forwards

its array of 14 infra-red proximity sensors and rotates in a direction that turns the gripper towards the strongest stimulus. It does this until the block is centrally located with one (or both) of the two centrally located sensors stimulated and all other sensors not stimulated. The robot then knows that the block is correctly positioned and is ready to be picked up.

In some cases, when multiple blocks are detected, the robot may not be able to align itself successfully. This can cause cyclic behavior where the robot first rotates left towards one stimulus, and then right, towards another. In other cases, when the object is actually another robot, the stimulus may disappear altogether. To resolve such situations the robot is limited in the number of attempts it makes.

Once aligned with a block, the robot reverses, lowers and opens its gripper, and then drives forwards before closing its gripper to grasp the block. The robot then reverses, raises its gripper and checks (with a magnetic finger separation sensor) whether or not a block is actually being held. In situations where the robot tries to pick up another robot this check invariably gives a negative response, meaning the robot needs to continue searching for a block.

(ii) Deposit {B} Depositing a block is quite straightforward. When a robot needs to make a deposit, potentially behind another block, it reverses a fixed distance first before lowering its gripper. When this is done, the gripper is opened to release the block which comes to rest on the ground. The robot then reverses, closes and then raises its gripper before rotating 180º, ready to drive away in search of another block.

(iii) Drive up beam towards light {C} Each robot is equipped with a 24 element light-sensor ring which gives a 360º field of view. During a sensing cycle, the intensity of light incident on each sensor in the ring is measured (in terms of a voltage) by the robot. By performing a sensing cycle and then processing the data, the robot is able to determine the direction corresponding to highest intensity. To drive up the light beam towards the light, a robot repeats the sequence: (i) Perform sensing cycle, (ii) orient in direction of highest intensity and, (iii) drive forwards a small distance. In this way, the robot is able to continually adjust its path to ensure it is heading up the beam towards the light.

(iv) Avoid obstacle {D} A robot uses its proximity sensors to monitor its immediate surroundings. To avoid an obstacle, the robot rotates on the spot, in a direction away from the object, until the object is no longer detected. In some atypical situations when there are multiple obstacles nearby, a robot may rotate back and forth on the spot as it tries to avoid one object and then the other. To avoid constant oscillation, the robot is limited in the number of times (namely three) it is allowed to move back and forth. After this it rotates in a direction that is towards the detected obstacle and

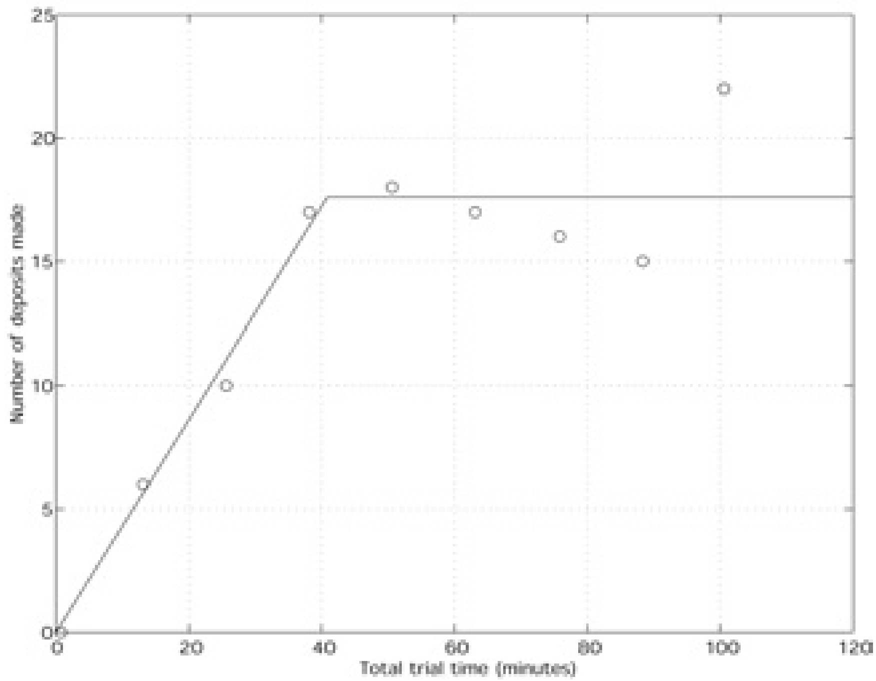


Figure 7 Results from the trials investigating the effect of latency time on the number of deposits made. The model, (2), has been fitted to the data (τ_{sat} = 40.9 minutes, w_{sat} = 17.6 deposits made).

continues to rotate in this same direction until the stimulus is no longer detected.

As can be seen from the organizer robot's rule set (Table 1) the organizer robot was programmed to move in a straight line spending L minutes at each stop until reaching the last stop ($n_{max} = 5$) at which point the trial was deemed over. The latency time, L, was varied in 2.5 minute intervals from 0 to 20 minutes resulting in a total of 9 trials. The total number of deposits made in each trial was recorded. Least-squares, nonlinear curve-fitting techniques were then used to fit the model, (2), to the raw data obtained from the experiments. The model parameters were found to be w_{sat} = 17.6 deposits made and τ_{sat} = 40.9 minutes.

The experimental results and the fitted model are shown in Figure 7 and the walls built during these trials (1–9) are given in the appendix. As can be seen from the plot, despite the fact that there is some noticeable variation in the total number of deposits for trials with a total trial time greater than τ_{sat} = 40.9 minutes, the results do support the proposed model. Explanations for this variation are detailed in Section 5. Ideally, many more trials would have been performed. Unfortunately though, as a consequence of using real robots that experience wear and tear, it is not possible to repeat experiments many times whilst maintaining controlled conditions.

It became clear from the trials that for latency times, $L \lessdot \frac{\tau_{sat}}{n_{max}} = \frac{\tau_{sat}}{5}$, the organizer robot was, on average, not spending enough time at each stop to allow each deposition window to be filled, nor the construction of a complete wall. Conversely, for $L \gtrdot \frac{\tau_{sat}}{5}$ the organizer robot was, on average, spending too much time at each stop such that the total trial time was too long. Ideally then, it seemed that a latency time of $L \approx \frac{\tau_{sat}}{5}$ would result, on average, in the construction of a complete wall (w_{sat} deposits) in the shortest possible time. We define this latency time to be the optimum latency time. From the trials, the optimum latency time was found to be approximately 8.2 minutes with the point of optimality in Figure 7 at (40.9, 17.6). It must be noted, however, that for a different experimental setup the optimum latency time is likely to be a different value. It has been suggested earlier that the use of a feedback mechanism might allow an adaptable choice of latency time to be made. Such a feedback mechanism would ideally result in the system operating close to the point of optimality. In the following section a distributed feedback mechanism designed for this purpose is introduced.

4 Distributed Feedback

Operation at the point of optimality requires that the organizer robot should move on to its next stop when the current deposition window has been filled with blocks. Therefore, the organizer robot should have some means of assessing window fullness. While the organizer robot does not have any direct contact with the deposition window, the builder robots do. In moving up the beam (rule 6, Table 3) they attempt to access the deposition window in order to deposit a block (rule 2, Table 3). On many occasions though, blocks already in the window prevent them from making this access (rule 5, Table 3). This inability to deposit a block may serve as a local cue as to the deposition window's fullness. However, because the deposition window is spatially large this cue may not always be an accurate indication. To increase accuracy, multiple samples could be taken. Builder robots frustrated at their inability to deposit could then signal to the organizer robot (using a burst of high intensity light) and the organizer robot, on receiving multiple flash signals (indicating a number of failed deposition attempts), could move.

To test this idea, the organizer robot's rule set was modified from the earlier trials to accommodate this feedback (Table 2). The builder robots maintained the same rule set as earlier (Table 3) except that rule 4 had added relevance, since the organizer robot was now able to respond to flashes made by builder robots. A builder robot becomes frustrated when it is making its way up the beam (rule 6, Table 3) and encounters an obstacle (rule 5, Table 3). Every time a robot becomes frustrated a frustration counter residing in its memory is incremented. If a robot is frustrated on two consecutive attempts, then it displays its frustration by making a flash of high intensity light (rule 4, Table 3). Its frustration counter is then reset. It is also reset when a successful deposit is made (rule 2, Table 3). The different states of the frustration counter are shown in Figure 8.

The organizer robot was programmed to use a sensor in its light sensor ring to look for flashes (rule 3, Table 2). On detecting a preset number of flashes (*flash_count_max*), the organizer robot was programmed to move on to the next stop (rule 2, Table 2). The different states of a flash counter (with *flash_count_max* = 3), that resided in the memory of the organizer robot, are shown in Figure 9b. This approach differs from earlier trials (Section 3) in which the organizer robot moved only when a certain time (equal to a fixed latency time) had elapsed (Figure 9a).

When the organizer robot moves it creates a fresh deposition window devoid of any blocks. Thus, the organizer robot's movement effectively applies negative feedback to the number of blocks currently in the deposition window. The feedback loop for this new system is depicted in Figure 10. Deposits are seen as an input to the system. It should be noted that in this system the feedback occurs at discrete times. The information passing between a source and a recipient may be unreliable, inaccurate or lost and this is indicated in the diagram (Figure 10) by the phrase *loose coupling*. We will regard the feedback as distributed since any one of the builder robots has the opportunity to contribute to the closing of the loop via the organizer robot. That is, the *flash_count_max* flashes required for the organizer robot to move can be provided by one or more of the robots.

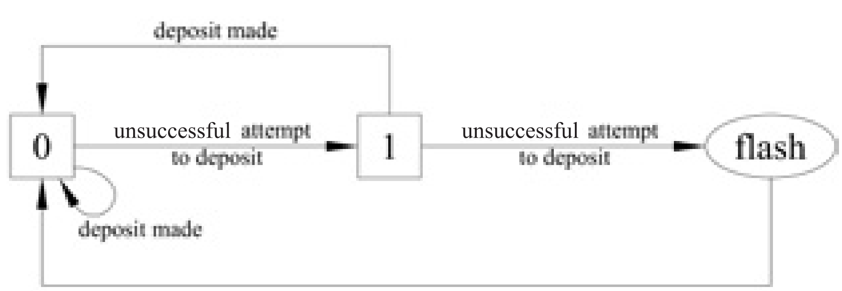


Figure 8 The different states of the frustration counter and the actions leading to state transitions.

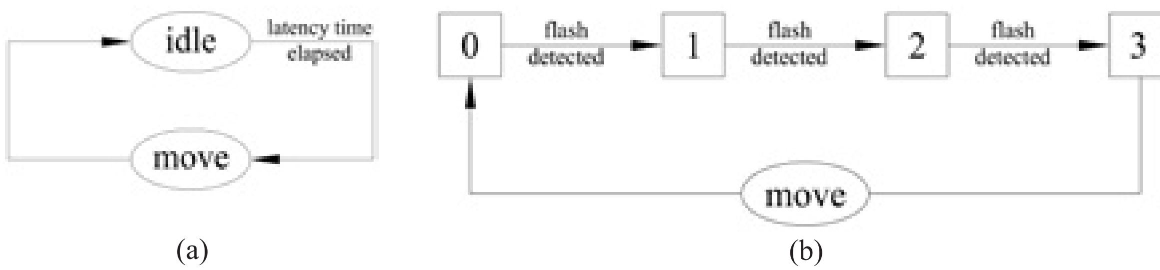


Figure 9 (a) Without feedback. The organizer robot moves to the next stop only when the latency time has elapsed. (b) With feedback. The organizer robot moves to the next stop when a certain number of flashes (*flash_count_max*) have been detected. The different states of the flash counter (with *flash_count_max* = 3) and the events leading to state transitions are shown.

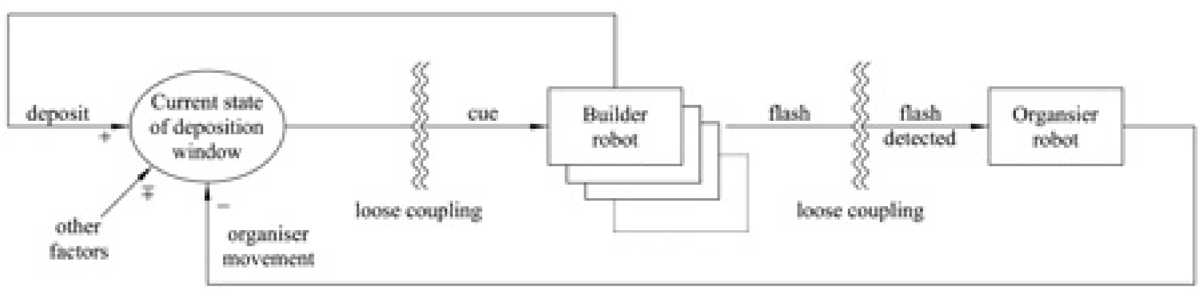


Figure 10 The distributed feedback system. Information from cues and signals flows throughout the system and closes the feedback loop.

As has been mentioned, an individual's assessment of the window's fullness may not be accurate. A robot may become frustrated prematurely by encountering another robot in the beam or by attempting to enter a portion of the deposition window that contains blocks even though the window as a whole may not be full. To increase the likelihood of a correct assessment being made, each robot was programmed to make two attempts before flashing (Figure 8 and rule 4, Table 3). This number was fixed for all trials.

The organizer robot was programmed to wait for *flash_counter_max* flashes before moving. Multiple flashes were required so that an isolated case of frustration would not lead to premature movement of the deposition window. To investigate how the value of *flash_counter_max* affected the quality of the information gathered and hence the quality of the wall, a series of trials were undertaken and are detailed in the following section. These trials also validate the proposed design for the feedback system.

4.1 Quality Versus Speed Trade-Off (Trials 10–19)

In order to determine an appropriate value for *flash_counter_max* a series of trials was undertaken. In these trials *flash_counter_max* was varied from 1 to 5 to determine its effect on the total trial time as well as the total number of deposits made. Two trials were performed for each value of *flash_counter_max* and the results averaged. A graph showing the averaged points (dark print) as well as the raw data points (light print) is given in Figure 11. In the discussions that follow, the phrase *n flash feedback* refers to a feedback trial with *flash_counter_max* = *n*. The walls built by the robots in Trials 10–19 appear in the appendix.

From Figure 11, a clear trend is evident. Increasing the number of flashes waited for increases both the number of deposits made and the total trial time, up to a limit of four flashes. Beyond four flashes

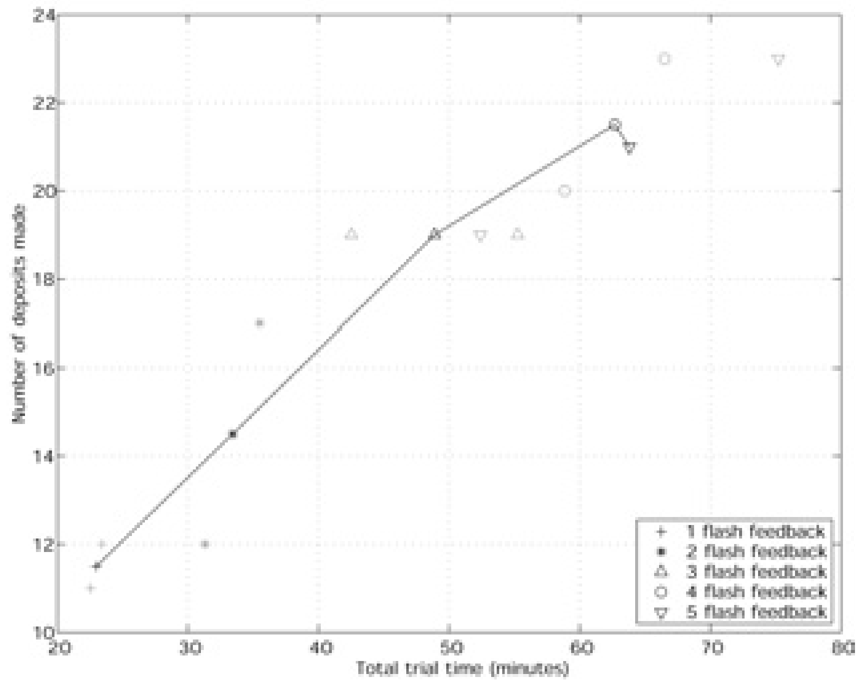


Figure 11 The effect that *flash_counter_max* has on the total trial time and the number of deposits made. Each of the points appearing in dark print has been averaged over two trials. The raw data points, from which the averages were made, are shown in a lighter shade.

there appears to be no further gain in the number of deposits made.

If the averaged points are compared with the curve obtained earlier (Figure 7) it becomes evident that 3 flash feedback operates closest (in terms of total number of deposits made) to the point of optimality. Using only 1–2 flash feedback results in the organizer robot moving when some windows are still partially empty. On the other hand, when 5 flash feedback was used, the time spent waiting for the 5th flash signal was effectively wasted. There is thus an obvious trade-off between speed (in terms of total trial time) and accuracy (in terms of the total number of deposits made) analogous to that discussed in Section 1.4.

This series of experiments confirmed the validity of the feedback system and showed that for this particular setup a value of *flash_counter_max* = 3 resulted in the system operating close to the point of optimality. It should be noted that this chosen value might not be ideal for all situations. In a swarm of larger size where there is likely to be more crowding and hence increased frustration, or where the size selected for the window is physically larger, a greater value for *flash_counter_max* may be necessary to ensure that the window is filled.

4.2 Information Flow

To better illustrate the feedback mechanism at work it is useful to look at how information flowed throughout the system in an actual trial. Trial 15 with 3 flash feedback provides a typical example which we will now focus on in some detail. The final wall structure constructed during this trial is shown in the appendix and repeated below in Figure 12.

During this trial the data logger was used to timestamp the completion of important events by each robot. Figure 13 shows the time of occurrence of these events. The stars on each horizontal line indicate when the corresponding event was completed by a robot. The data on each horizontal line are expanded upon further in Figures 14 and 15 which show blocks picked up, deposits made, frustration of robots, flashes made by robots and movement made by the organizer robot. These plots not only depict the data more clearly, they also provide additional informa-

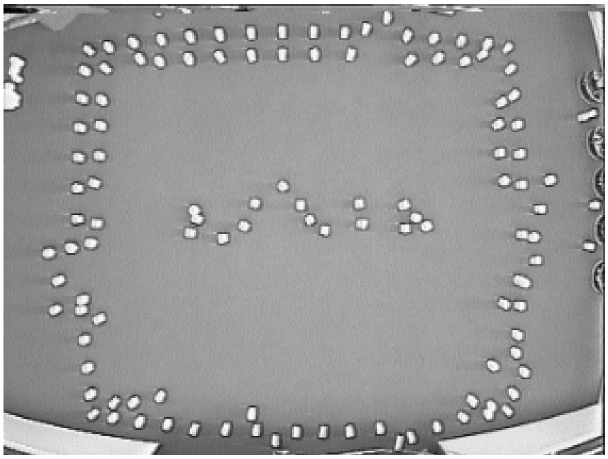


Figure 12 The final wall structure constructed in Trial 15.

tion about which robot actually completed the recorded event.

A breakdown of the percentage of trial time spent by each builder robot undertaking (i) foraging for building material, (ii) looking for the beam, (iii) traveling in the beam, and (iv) making a deposit, is given in Figure 16. In the pie charts it is seen that a relatively small percentage of time is spent foraging for building material with most of the time actually spent searching for the beam.

At the start of the experiment each robot acquired a block very quickly. This accounts for the high initial gradient for the block "pickups" curve in Figure 14. After this start-up period the robots generally acquired a block relatively soon after they had made a deposit, as suggested by the pie charts (Figure 16). This is also evidenced in Figure 14 where there tends to be a correspondence of robot activity in the two curves (at least after the start-up period).

Figure 15 indicates the times of frustration experienced by each robot. On making a failed attempt to deposit, a robot will turn randomly ±135° (rule 5, Table 3). This may lead to it moving outside the beam and hence wandering off (rule 8, Table 3) or may lead it to making another approach up the beam (rule 6, Table 3). In the latter case robots may become frustrated multiple times in a row, leading to flashes being produced and resulting in movement of the organizer robot. The distributed nature of the feedback is highlighted by Figures 14 and 15 which show robots contributing to the filling of the window (which influences a cue) or getting frustrated and making a flash signal.

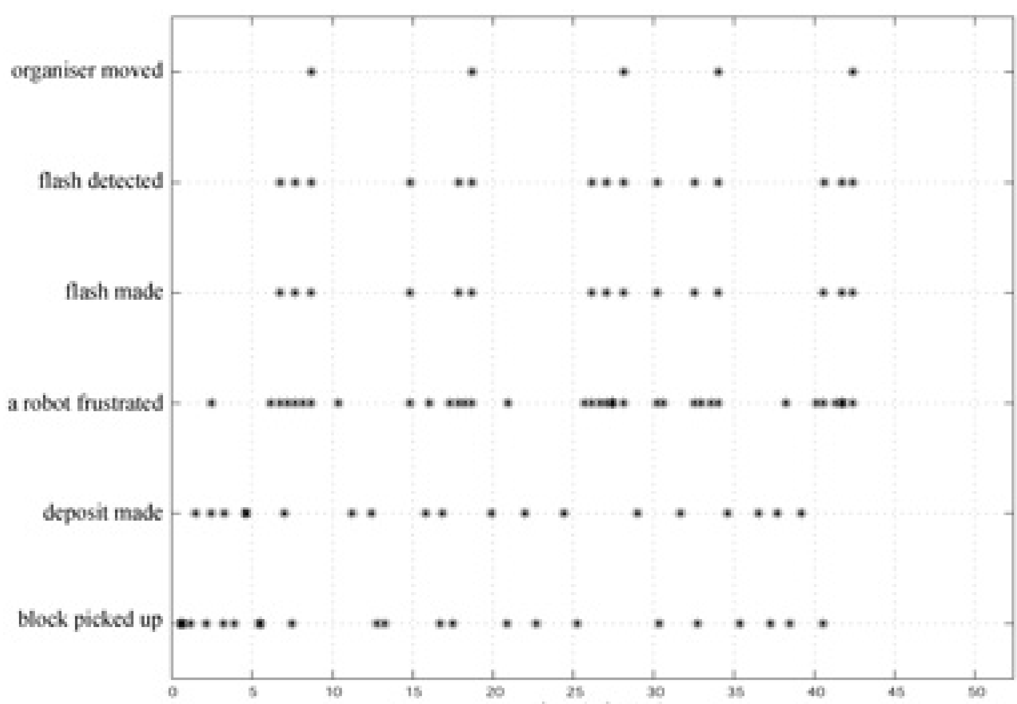


Figure 13 The occurrence of important events during Trial 15.

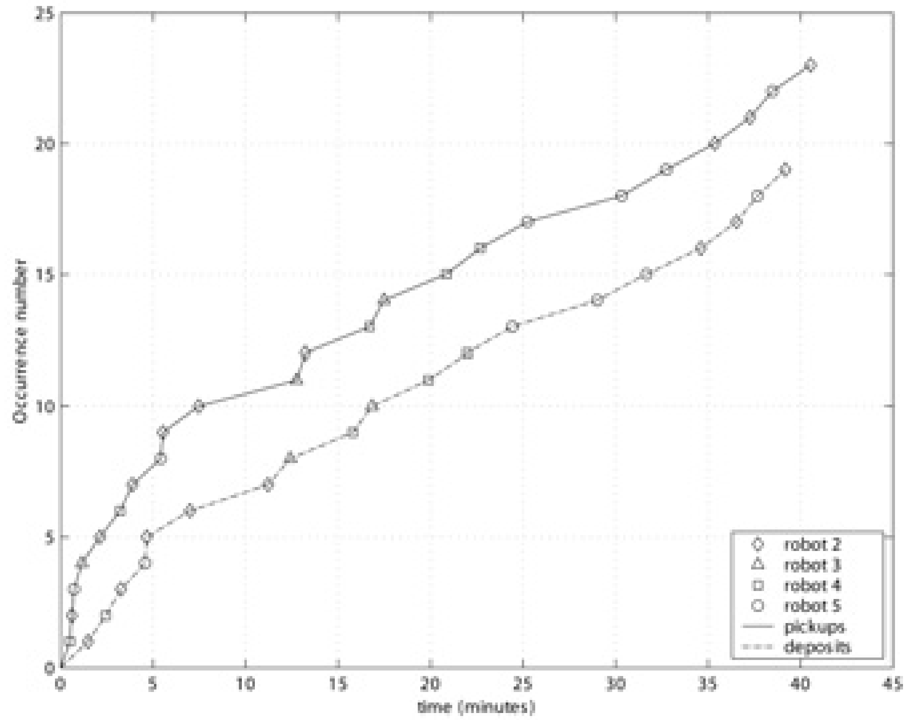


Figure 14 The times that blocks were picked up and deposited by different robots during Trial 15.

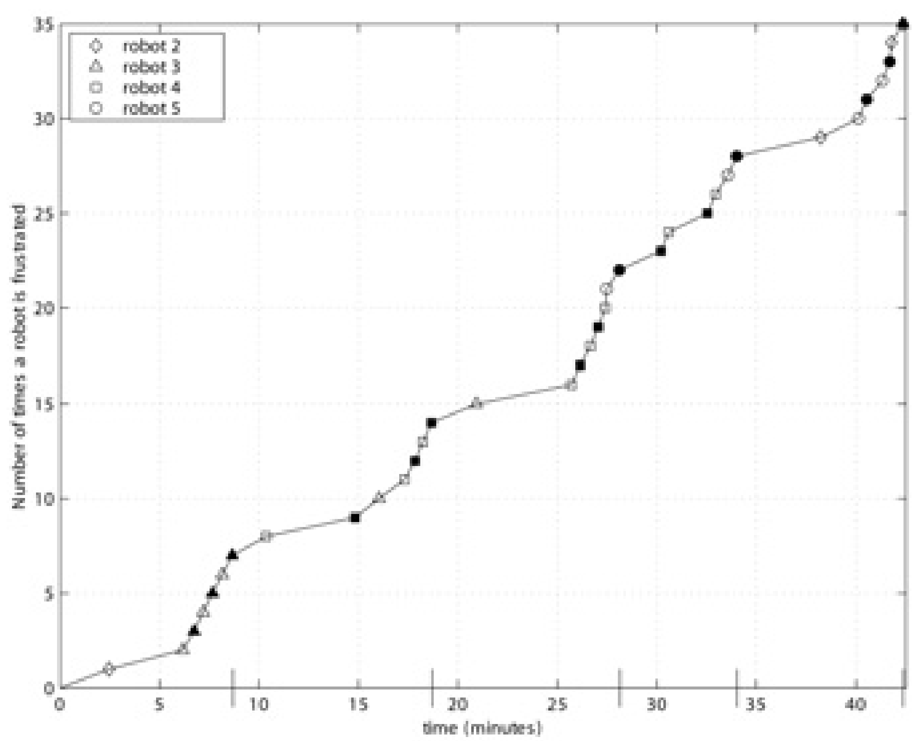


Figure 15 The times that different robots became frustrated during Trial 15. Filled-in markers indicate when a robot was frustrated and produced a flash. The times when the organizer robot moved have been marked on the *x*-axis.

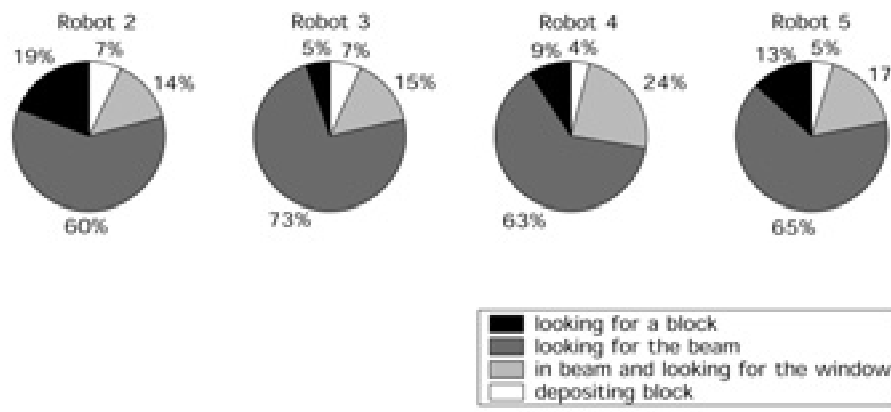


Figure 16 A breakdown of the percentage of the total trial time that various activities were being undertaken by each robot.

In this particular trial all flashes were detected by the organizer robot.

The effect of cues and signals on the actions of individuals becomes clear on close inspection of Figure 13. From this diagram we can see that a cyclic pattern emerges from the individuals' actions. Blocks are deposited and start to fill the deposition window. As the window begins to fill, some robots may start to experience frustration. The rate of deposits then decreases and a period of intense frustration takes over. This is a result of individuals experiencing great difficulty in depositing blocks despite multiple attempts from different approach angles. This is also clear from Figure 15 where frustration events tend to cluster together. The frustration leads to flash signals being produced and detected and then finally the organizer robot moving. The movement effectively zeros the number of blocks in the window and serves to allow blocks to be deposited again and the average frustration level to decline.

Despite the fact that attempts made to deposit can be sporadic, order is able to emerge from the system. Individuals assess the fullness of the deposition window using a cue and either make a deposit or pass relevant information on to the organizer robot using a signal. Information is thus seen to flow throughout the system. The rate of movement by the organizer robot is relatively constant (Figure 15). This is because the environment is not changing rapidly and affecting the average rate of deposits. In the next section we look at situations where the environment is changed radically during a trial.

4.3 Perturbation Experiments

Thus far it has been shown that for a particular system setup there exists an optimal trial completion time and that by using feedback the system can be made to work close to this optimum. In the previous trials robots operated in an environment that did not vary greatly. That is, factors influencing the rate of deposit attempts did not vary significantly throughout the course of each trial. In this section a series of perturbation experiments are detailed that investigate the system's ability to adapt to changes in the environment. The perturbations made were chosen because they were expected to have a direct and noticeable affect on the rate of deposits.

4.3.1 Removal of Robots (Trial 20) A feedback trial (*flash_count_max* = 3) was initiated with the complete swarm of 5 robots. The robots commenced building as per usual completing 3 deposition windows. At this point in the experiment, after the third movement of the organizer robot had been made, an observer perturbed the system by manually removing two of the builder robots so that the total swarm size was reduced to 3.

The effect that this action had on the system can be seen in Figures 17 and 18. The shaded region in each figure indicates the time during which the system was operating with only 3 robots. In Figure 17 the frequency of each event decreases during the shaded period. This can also be seen in Figure 18 where the time between

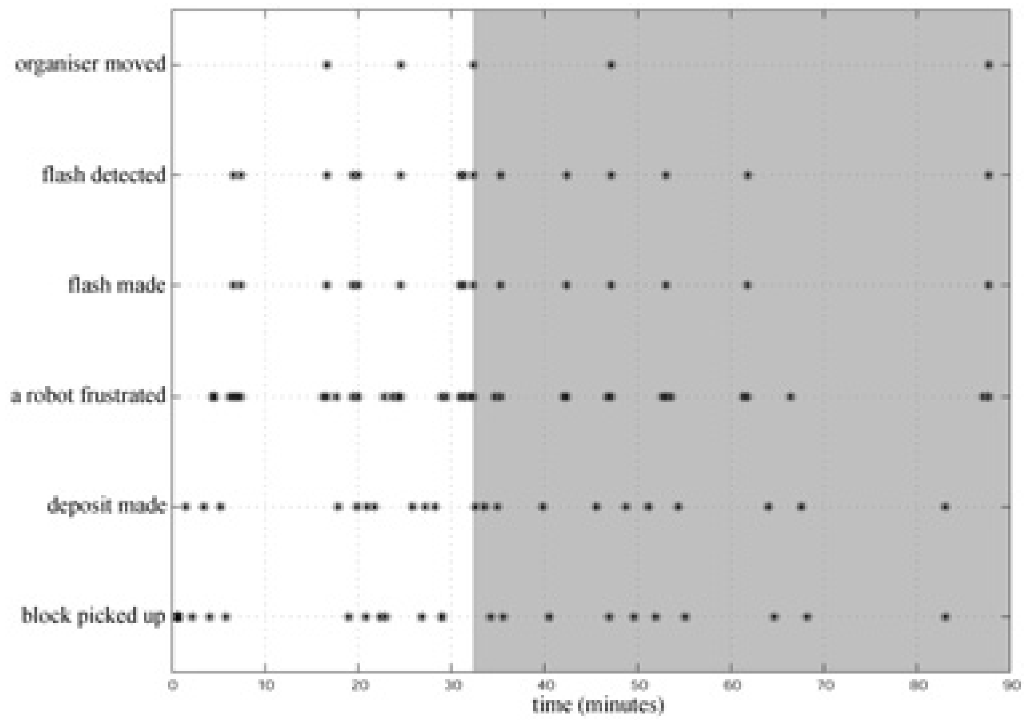


Figure 17 The occurrence of important events during Trial 20. The shaded portion indicates approximately the period of time during the trial when the swarm size was reduced to 3.

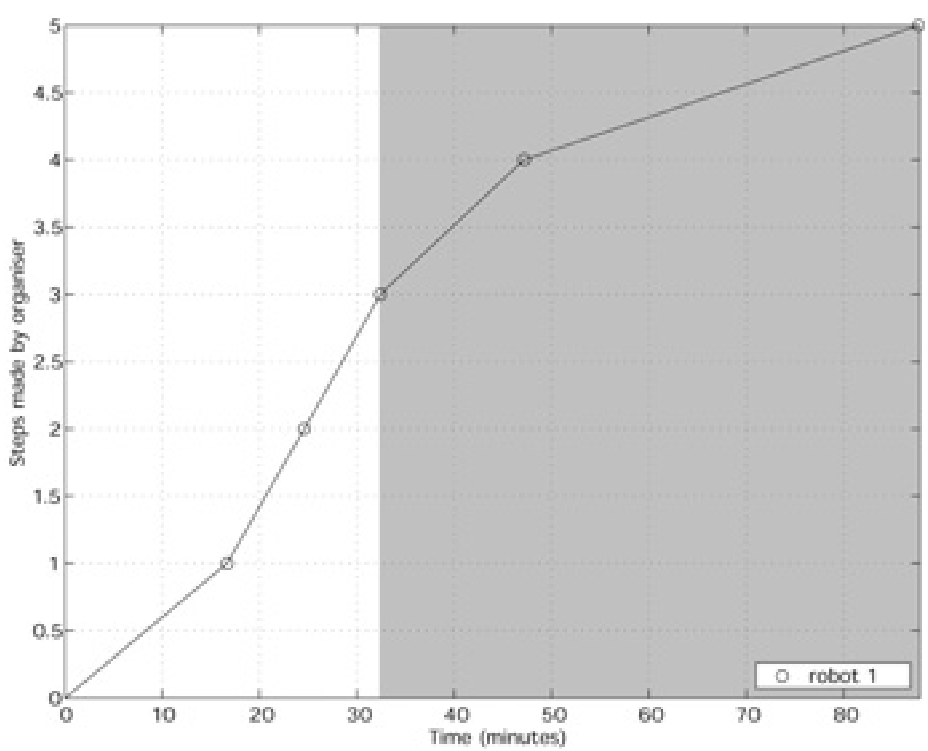


Figure 18 The times when the organizer robot moved to the next stop in Trial 20. The shaded portion indicates the period of time during the trial when the swarm size was reduced to 3.

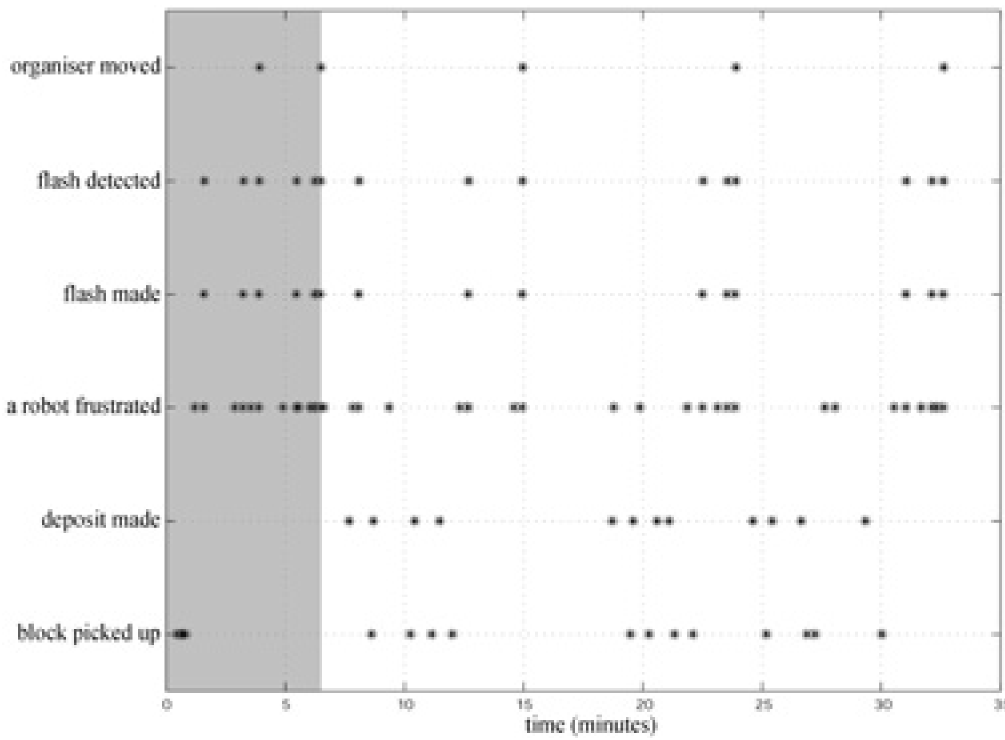


Figure 19 The occurrence of important events during Trial 21. The shaded portion indicates approximately the period of time during the trial when there was an obstacle occupying the first two deposition windows.

movements (approximately equivalent to latency time) during 3-robot operation is, on average, longer than during the period where 5 robots were operating. This slowing down of the organizer robot permitted the remainder of the wall to be completed successfully despite the reduced swarm size (see Trial 20 in the appendix).

From this trial it is clear that the feedback mechanism allowed the system to adapt to a change in the environment (in this case the removal of robots) without the need to quantify or be directly informed of the change. This adaption was possible because the organizer robot was able to change its latency time to accommodate the change in conditions.

4.3.2 Obstacle (Trial 21) To further illustrate the adaptive ability of the system when using feedback a second trial was undertaken. In this trial an artificial obstacle was added into the test area prior to the commencement of the trial (Figure 21a). This obstacle occupied an area where the first and second deposition windows would be so that robots would be unable to deposit blocks in these windows.

Again, the system was able to respond to the perturbation. Since the robots were unable to deposit any blocks where the obstacle was located, they quickly became frustrated, causing the organizer robot to move on to the next stop relatively quickly. The shaded region in Figures 19 and 20 highlights this period of time. Figure 19 shows that during this period no deposits were made and instead there was great frustration. Once the organizer robot had moved past the obstacle, the system resumed its typical operation where the robots had to fill the deposition windows themselves. Figure 20 shows how the organizer robot moved relatively quickly past the obstacle, adapting its rate of movement to compensate for variations in the environment. The final wall structure for this trial is shown in Figure 21b (and also Trial 21 in the appendix) where it can be seen that the obstacle forms part of the wall.

The perturbation experiments demonstrate that the system is able to adapt to changes in the environment. Such adaption would not be possible without the use of feedback. Feedback allows the latency time to vary

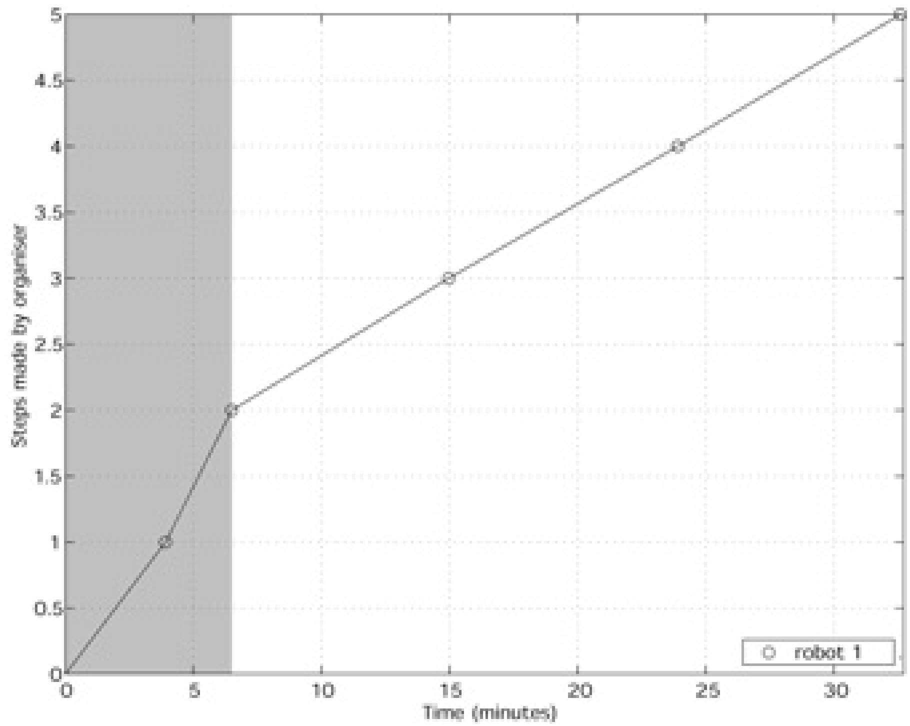
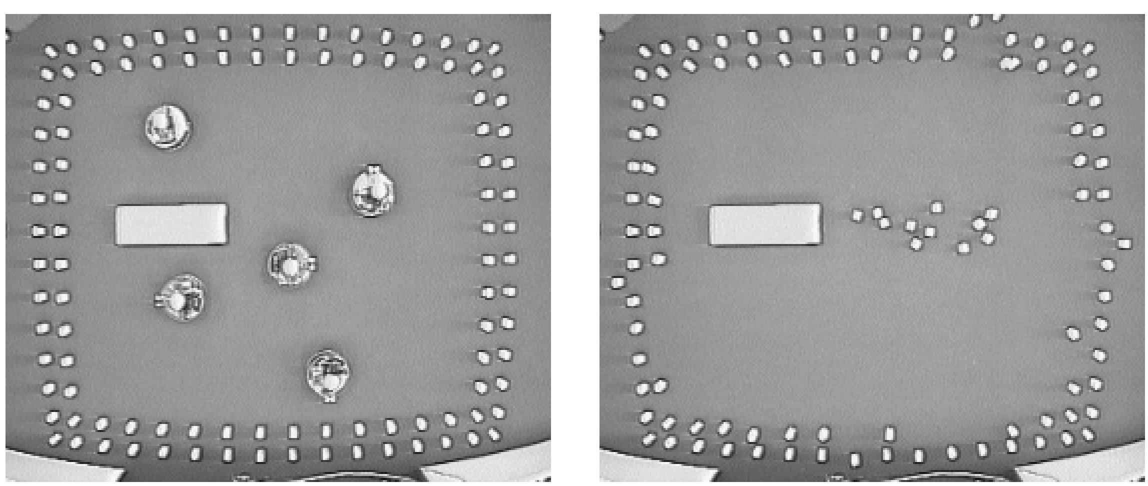


Figure 20 The times when the organizer robot moved to the next stop in Trial 21. The shaded portion indicates the period of time during the trial when there was an obstacle occupying the first two deposition windows.



(a) (b)

Figure 21 Overhead images of the test area. (a) At the beginning of the trial and (b) after the completion of the trial (with the robots removed). An obstacle (which appears as a large white rectangle) has been incorporated into the final wall structure.

in response to perturbations affecting the deposit rate and ensures the wall quality is, on average, consistent along its length. When the deposit rate dropped (due to the removal of 2 robots) the system slowed to compensate and when the effective deposit rate increased (due to an obstacle), the system sped up.

5 Factors Affecting Result Variability and Assumptions

In the analysis of the trials undertaken in this paper, the variable, *number of deposits made*, was often used. It was hoped that this variable would provide an indication of the number of blocks deposited in the deposition windows and hence incorporated into the final wall structure. It was also intended that the total trial time would be an accurate indication of the speed of the system. However, over the 21 trials that were performed many different situations were observed that had an effect on these two performance criteria as well as on the shape and consistency of the final wall structure. These often unexpected phenomena go some way in explaining the variability observed in the results that were obtained. Each trial generally had a number of these atypical happenings that influenced the results. The following list outlines many of these unexpected actions that were observed during the trials detailed in this report as well as in other test trials:

- A robot that drove in the path of a robot depositing a block may have been unfortunate enough to have the block deposited on top of it.
- Instead of depositing blocks at the edge of the window (rule 2, Table 3), sometimes premature deposits were made. This was often triggered by a block in an adjacent window or by another robot near by (rule 2, Table 3).
- A robot depositing a block and being unable to reverse a sufficient distance (due to another robot being in the way) sometimes closed its gripper and picked up the same block it had just deposited.
- Blocks were sometimes taken from the wall by a foraging robot. This was because blocks in the wall were not distinguished uniquely as being part of a structure (rule 1, Table 3).
- Robots also tried to pick up other robots thinking that they were a building block. This was again because all obstacles were treated as being potential blocks. On occasions this behavior actually caused damage to the proximity sensors of the robot thought to be a block.
- Robots with damaged proximity sensors were constantly under the impression that they were in the presence of an obstacle. In trying to execute their obstacle avoidance behavior they were seen to turn continuously on the spot until the end of the trial.
- Failure of the sensor indicating whether or not a block was being held led to builder robots undertaking the deposit action without the presence of a real block.
- On some occasions blocks were picked up poorly causing some sensors in the light sensor ring to be obscured. This reduced the robot's ability to track up the beam towards the light source.
- On many occasions robots deposited blocks and also became frustrated outside the deposition window. This was probably due to errors in sensor readings, partial obscuring of the absolute light sensor, and by robots taking readings in the penumbra of the light source.
- On some rare occasions flashes were erroneously detected by the organizer robot. It is thought that this was due to light from the organizer robot's light source reflecting off a closely situated builder robot and then being detected by the organizer robot.
- Flashes produced by builder robots were not always detected by the organizer robot. This usually happened when the light produced by the flash was obstructed by another robot.
- The flash bulb onboard each builder robot did not flash in some situations due to hardware failure and the builder robot did not detect such situations.
- Blocks were often knocked by turning or reversing robots. On numerous occasions this caused significant damage to the wall structure including the compression of blocks or the creation of large gaps.
- In most cases the robots approached the deposition window by driving up the gradient of the beam (rule 6, Table 3). In some cases though, builder robots came upon the deposition window from another angle. After making a deposit these robots would often reverse into blocks already in the wall causing damage.
- The organizer robot's path was, on some rare occasions, blocked by builder robots.
- Robots often became jammed, sometimes taking a while to free themselves. This was particularly the case in situations involving the organizer robot. Because it was stationary most of the time it was unable to aid the other robot in becoming unstuck.

- There were some small bumps on the carpeted surface of the test area. On several occasions robots were seen to become stuck on these bumps before freeing themselves.
- Blocks were placed in different configurations in each deposition window. Deposition windows with more compact block configurations thus had a higher count for the number of deposits made than windows with poor block placement.

Despite all of these factors that influenced the results, the system was mostly fault tolerant. In all the cases detailed in this report a wall was successfully constructed (see the appendix for pictures of the walls that were constructed in each trial). However, some trials (not documented here) were aborted due to major hardware failures that disrupted the controlled conditions of the experiment, such as the organizer's light bulb blowing, a wheel motor on board one of the builder robots failing and failure of the communication link with the data logger. As it stands, this system has a single point of failure in the organizer robot. By giving each robot the ability to become an organizer, a failed organizer could be replaced and this would mean the robustness of the system as a whole could be improved. Robustness could also be enhanced through the use of a larger swarm, with multiple organizer and builder robots all working concurrently. At this level, individual organizer robots would be dispensable.

It was felt that small fluctuations caused by many of the above factors would not be significant enough to cause a shift in the average rate of deposit attempts. This assumption of an essentially unchanging environment is made in the formulation of the model introduced in Section 3. In Trial 15, which involved the use of feedback, the organizer robot was seen to move at a fairly constant rate (Figure 15) indicating the validity of this assumption. Other factors would affect the latency time more significantly than those mentioned above. These have already been mentioned in Section 1.3.3 and include swarm size, block availability, test area dimensions, beam size and orientation, deposition window size and position, robot behaviors, and obstacles in the environment including the structure itself. In the perturbation experiments the assumption of an unchanging environment was purposefully violated by introducing large-scale environmental changes in order to investigate the adaptive ability of the system.

For the building of structures in real applications, it is likely that structural quality will be of some importance. For the analyses in this paper, the number of deposits made was a performance metric that was used to provide a quantitative measure of the success in building a wall structure. From this, a complete wall was taken to mean a structure for which w_{sat} deposits had been made. This metric did not account for the actual placement of blocks or whether or not there were any gaps in the wall. In this system, the robots had only very simple sensing and actuator capabilities and as such did not place blocks precisely. The aim in this work was to build a loose wall structure that could form a framework that could be reinforced at a later stage. For real applications in which deficiencies in wall structure may be of importance, it may be desirable to introduce other metrics to measure wall completeness, or as in Wilson, Melhuish, Sendova-Franks, and Scholes (2004), to find the best achievable structure.

6 Discussion

The feedback system detailed in this paper illustrates a form of multi-agent adaption that does not require the use of radio communications, shared memory, planning or performance evaluation. Instead the objectives of the swarm were achieved using an implementation that relied on signals and cues found in the environment. These techniques were inspired by behavior observed in social insects.

In collective construction tasks the aim is to assemble building material at various locations in space. This can be achieved through the use of a spatio-temporal varying template which acts to bring a certain amount of spatial coordination to the swarm. Quantitative information is distributed spatially within the template (in the form of a light-intensity gradient) and is used by the robots to determine where they should deposit building material. This building material itself then serves to pass on information (regarding window fullness) to builder robots.

Upon entering a deposition window, block-carrying builder robots move towards the light source until they cross over the deposition window's inner limit or detect another block (rule 2, Table 3). At this point they make a deposit. The case where a block is deposited as a result of another block being detected is an example of stigmergy. Here, the builders are stimulated to build "by the very performances they have achieved" (Grassé, 1959). However, in this system an overriding template moderates this stigmergic action by preventing robots from making deposits at all when the perceived light intensity falls outside a certain fixed range. When builder robots are unable to make a deposit they become frustrated and can produce a flash (rule 4, Table 3). Here the actions of the builder robots are again affected by their past actions through the state of the deposition window. However, in this case, instead of being stimulated to do further work they are stimulated to become frustrated and flash. Because previous actions are seen to affect future actions, this process is an example of sematectonic communication (see Section 1.2).

It is possible to classify this system in more general terms by making reference to known pattern generation mechanisms. Specifically, following the classification scheme of Cazamine et al. (2001), the pattern generating capabilities of this system are the result of a *leader*, following a *recipe*. Because the leader (organizer robot) is carrying a light source, a spatio-temporal varying *template* is created that imposes order on the system. At each step in the recipe, the leader drives to a new location and the transition between steps in the recipe is facilitated through a feedback system. This process of using a recipe differs from qualitative stigmergy in that instead of the previous building itself creating the stimulus for future building (e.g., Theraulaz & Bonabeau, 1995), an organizer robot creates the stimulus. This robot alone determines the location of the next deposition window by following its recipe.

7 Future Work

Through a simple extension to the current template-generating technique, a solution to the generalized 2D collective construction problem can be found. Currently, the distance of the deposition window from the organizer robot is fixed. However, variable deposition window placement can be achieved, simply by varying the light source intensity. This means builder robots detect the same intensity but at a different location relative to the organizer robot. By systematic source intensity variation and linear motion by the organizer robot, it is theoretically possible to build a loose planar structure of any given design. Indeed, this claim has recently been supported in a number of trials in which more complex structures have been built (Stewart & Russell, 2005). In that work, the desired 2D structure was decomposed into a recipe (or building program), which the organizer robot followed. Each step in the recipe contained a linear movement value and/or a new light source intensity value. The feedback system facilitated the transition between steps in the recipe demonstrating its applicability in solving more complex construction tasks.

With a solution to the generalized construction problem demonstrated, it is useful to consider directions for further research. Instead of pre-programming the entire recipe for the organizer robot, the recipe steps could be written by the organizer robot "on-the-fly" according to some perceived environmental stimulii. For example, an organizer robot capable of detecting a chemical spill at some location (Kowadlo & Russell, 2004) could then organize builder robots to create a circular wall around the spill with a radius and thickness sufficient to contain it. Building adaptive structures is likely to be beneficial in situations where a collective construction task of a general nature is required but the environment is too uncertain to pre-program the entire recipe in advance. To cater for these situations it may be useful to specify only loose and qualitative constraints. For example, a robot swarm could be instructed to build a structure with three rooms, leaving the size and location of rooms as variables. The actual recipe implementation could then incorporate these constraints whilst allowing the structure to be fitted to the environment in an appropriate manner.

In this article a feedback system was designed to regulate one particular process: namely, construction. The extent to which feedback loops can be used to regulate multiple processes in large robot swarms operating in unpredictable environments is unclear. It is of particular interest to consider the extent to which a robot swarm and its environment can be regarded as an engineering control system. If such an analogy can be made, a new approach to the design, understanding, and control of swarms may be possible. Similar to an engineering control system (Dutton et al., 1997), it may be possible to design a distributed controller that, when added, optimizes the system in terms of response time and its ability to accurately track changes in the environment. This controller could potentially be implemented by simply adding some "catalyst" robots programmed to carry out simple behaviors in response to local conditions.

8 Conclusion

The work detailed in this article focused on the implementation of a feedback system for collective construction. A spatio-temporal varying template, providing quantitative information, was presented as a means of facilitating the creation of complex spatial structures by a robotic swarm. Previous work revealed the need for a mechanism that could regulate the building process in an environment that may be constantly changing. To this end, a feedback system was introduced that was inspired by the exploits of social insects.

In developing and investigating this system, a total of 21 experimental trials were undertaken in which a swarm was assigned a wall building task. Firstly, it was shown that there exists an optimal latency time that allows the construction of a complete wall in the shortest possible time. By introducing feedback into the system it was then shown how the system could regulate itself to achieve operation close to this point of optimality. The feedback system was facilitated by direct and indirect transfer of information between individuals, using signals and cues. By analyzing one trial in detail, the distributed nature of the feedback became apparent showing all robots had the capacity to help close the feedback loop. From the analysis it also became clear how information flowed throughout the system. Individual actions were sporadic but the feedback brought about cohesion amongst the swarm and its environment.

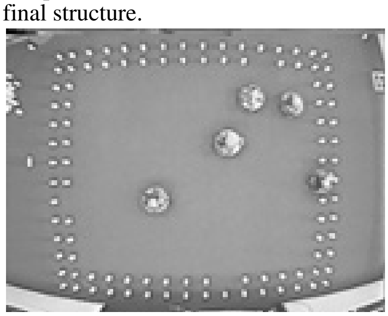
The ability of the swarm to cope with changes in the environment was then investigated in perturbation trials. In the first of these trials, the swarm size was reduced from 5 to 3 part way through the experiment. This had the effect of reducing the rate of deposit attempts. Despite this change, the system was able to adapt by slowing the organizer robot's rate of movement. In the second trial an obstacle was introduced into the test area in a location that served to inhibit blocks being placed in the first two deposition windows. Again the system was able to adapt, causing the organizer robot to move past the obstacle relatively quickly. It was shown how feedback allowed the system to adapt by changing the actions of individuals. These perturbation trials illustrated the need for feedback in systems that are operating in variable environments. Feedback allowed the swarm to respond to environmental changes without the need to model or predict the changes directly. This demonstrates a flexibility that is likely to be a key in developing practical robotic applications.

Appendix: Walls Constructed During the Trials

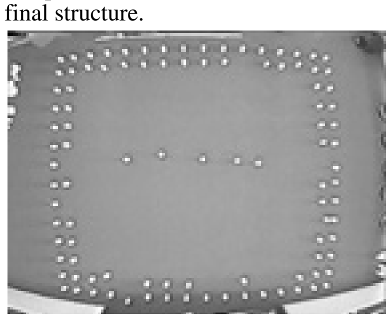
Trials 1–9: Effect of varying latency time
Trial 10–19: Quality versus speed trade-off
Trial 20–21: Perturbation experiments

The following images show the configuration of the blocks in the test area after the completion of each trial. In each image (excluding Trial 1) the robots have been removed from the test area so that the wall structures built can be seen more clearly. In each case the wall was constructed from left to right.

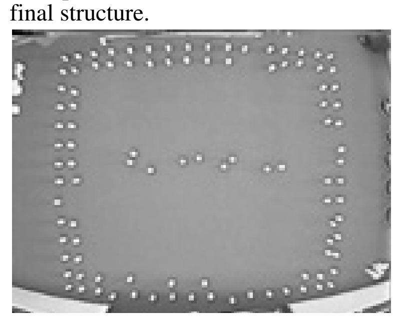
Trial 1.
No feedback, $L = 0$ minutes.
Total trial time = 0.7 minutes.
0 deposits made, 0 blocks in final structure.



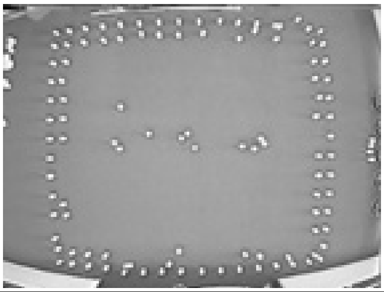
Trial 2.
No feedback. $L = 2.5$ minutes.
Total trial time = 13.1 minutes.
6 deposits made, 5 blocks in final structure.



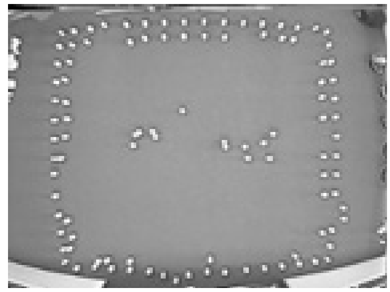
Trial 3.
No feedback. $L = 5$ minutes.
Total trial time = 25.6 minutes.
10 deposits made, 9 blocks in final structure.



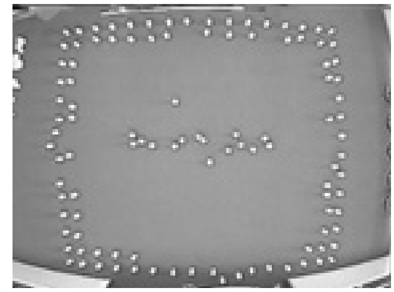
Trial 4.
No feedback. L = 7.5 minutes.
Total trial time = 38.1 minutes.
17 deposits made, 11 blocks in final structure.



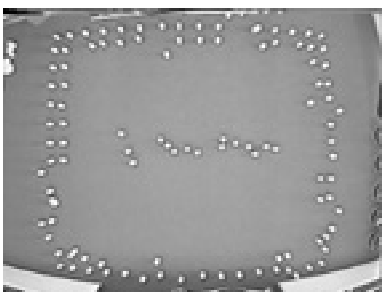
Trial 5.
No feedback. L = 10 minutes.
Total trial time = 50.6 minutes.
18 deposits made, 13 blocks in final structure.



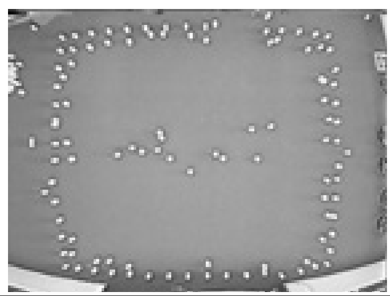
Trial 6.
No feedback. L = 12.5 minutes.
Total trial time = 63.1 minutes.
17 deposits made, 16 blocks in final structure.



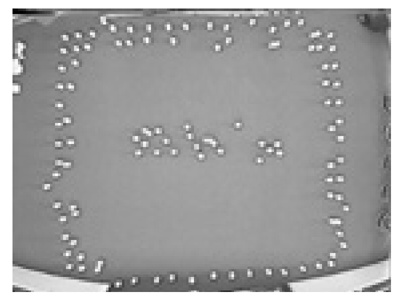
Trial 7.
No feedback. L = 15 minutes.
Total trial time = 75.8 minutes.
16 deposits made, 15 blocks in final structure.



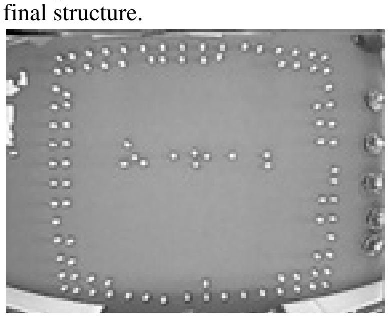
Trial 8.
No feedback. L = 17.5 minutes.
Total trial time = 88.3 minutes.
15 deposits made, 13 blocks in final structure.



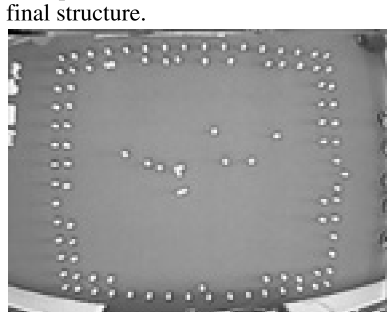
Trial 9.
No feedback. L = 20 minutes.
Total trial time = 100.6 minutes.
22 deposits made, 22 blocks in final structure.



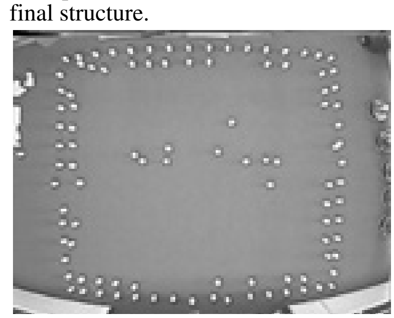
Trial 10.
1 flash feedback.
Total trial time = 22.6 minutes.
11 deposits made, 11 blocks in final structure.



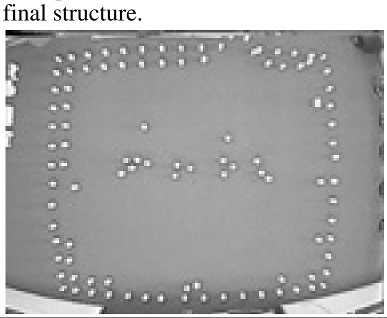
Trial 11.
1 flash feedback.
Total trial time = 23.4 minutes.
12 deposits made, 10 blocks in final structure.



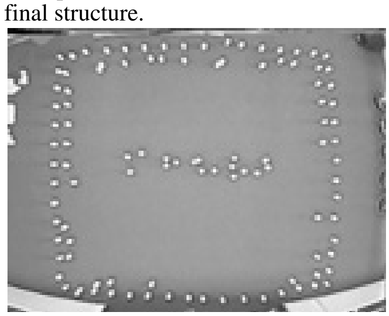
Trial 12.
2 flash feedback.
Total trial time = 31.3 minutes.
12 deposits made, 10 blocks in final structure.



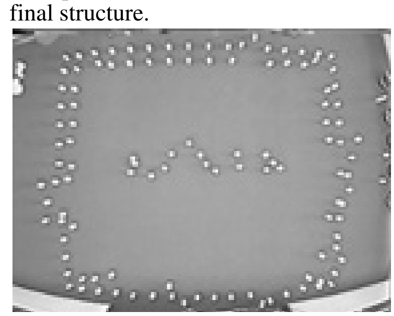
Trial 13.
2 flash feedback.
Total trial time = 35.5 minutes.
17 deposits made, 16 blocks in final structure.

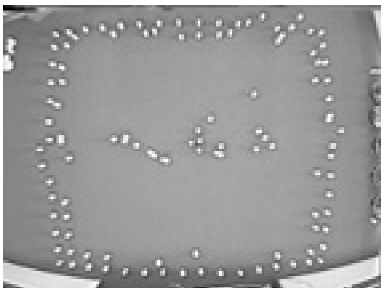
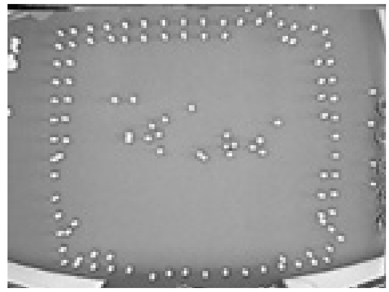
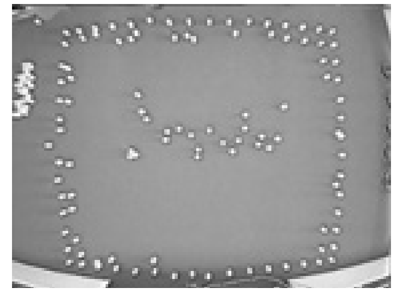
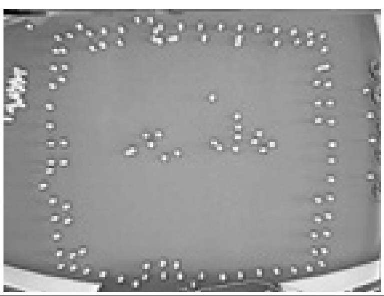
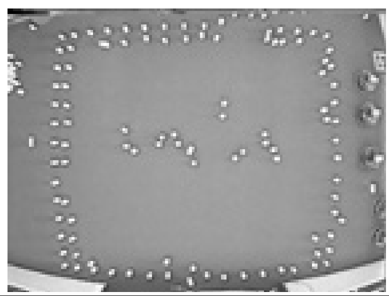
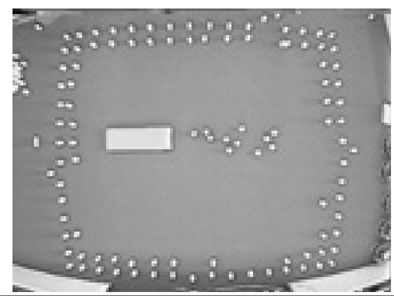


Trial 14.
3 flash feedback.
Total trial time = 55.3 minutes.
19 deposits made, 17 blocks in final structure.



Trial 15.
3 flash feedback.
Total trial time = 42.5 minutes.
19 deposits made, 16 blocks in final structure.



Trial 16. 4 flash feedback. Total trial time = 66.5 minutes. 23 deposits made, 20 blocks in final structure. 	**Trial 17.** 4 flash feedback. Total trial time = 58.9 minutes. 20 deposits made, 20 blocks in final structure. 	**Trial 18.** 5 flash feedback. Total trial time = 75.2 minutes. 23 deposits made, 22 blocks in final structure. 
Trial 19. 5 flash feedback. Total trial time = 52.4 19 deposits made, 18 blocks in final structure. 	**Trial 20.** Perturbation: Removal of 2 robots. Total trial time = 87.8 minutes. 21 deposits made, 18 blocks in final structure. 	**Trial 21.** Perturbation: Obstacle. Total trial time = 32.7 minutes. 12 deposits made, 11 blocks in final structure. 

Note

1 Melhuish has introduced the all encompassing term autostruosis to describe the formation of macroscopic structures where the distinction between robots and the environmental building material is blurred (Melhuish, 2000).

Acknowledgments

The work described in this paper was supported by the Australian Research Council funded Centre for Perceptive and Intelligent Machines in Complex Environments. The work is also supported by a grant from Monash University's Faculty of Engineering Small Grants Scheme. Robert Stewart would like to acknowledge the support of an Australian Postgraduate Award. The help and advice of academic and support staff within the Department of ECSE, Monash University, has been appreciated.

References

Anderson, C. (2002). Self-organization in relation to several similar concepts: Are the boundaries to self-organization indistinct? *Biological Bulletin*, *202*, 247–255.

Anderson, C., Franks, N. R., & McShea, D. W. (2001). The complexity and hierarchical structure of tasks in insect societies. *Animal Behaviour*, *62*, 643–651.

Anderson, C., & McShea, D. W. (2001). Intermediate-level parts in insect societies: Adaptive structures that ants build away from the nest. *Insectes Sociaux*, *48*, 291–301.

Anderson, C., & Ratnieks, F. L. W. (1999a). Task partitioning in foraging: General principles, efficiency and information reliability of queueing delays. In C. Detrain, J.-L. Deneubourg, & J. Pasteels (Eds.), *Information processing in social insects* (pp. 31–50). Basel: Birkhäuser.

Anderson, C., & Ratnieks, F. L. W. (1999b). Worker allocation in insect societies: Coordination of nectar foragers and nectar receivers in honey bee (*Apis mellifera*) colonies. *Behavioral Ecology and Sociobiology*, *46*, 73–81.

Beckers, R., Holland, O. E., & Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics. In R. Brooks & P. Maes (Eds.), *Artificial life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems* (pp. 181–189). Cambridge, MA: MIT Press.

Bonabeau, E. (1997). From classical models of morphogenesis to agent-based models of pattern formation. *Artificial Life*, *3*, 191–211.

Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm intelligence: From natural to artificial systems*. New York: Oxford University Press.

Bonabeau, E., Theraulaz, G., & Cogne, F. (1998). *The design of complex architectures by simple agents* (Working Paper No. 98-01-005). Santa Fe Institute.

Born, M., & Wolf, E. (1964). *Principles of optics* (2nd ed.). Oxford: Pergamon.

Brooks, R. A., Maes, P., Matarić, M. J., & More, G. (1990). Lunar base construction robots. In *Proceedings of 1990 IEEE International Conference on Intelligent Robots and Systems (IROS'90)* (pp. 389–392). IEEE.

Bruinsma, O. H. (1979). *An analysis of building behaviour of the termite* Macrotermes subhyalinus *(rambur)*. Unpublished doctoral dissertation, Landbouwhogeschool te Wageningen, The Netherlands.

Camazine, S., Deneubourg, J.-L., Franks, N. R., Sneyd, J., Theraulaz, G., & Bonabeau, E. (2001). *Self-organization in biological systems*. Princeton, NJ: Princeton University Press.

Cassill, D. (2003). Rules of supply and demand regulate recruitment to food in an ant society. *Behavioral Ecology and Sociobiology*, *54* (5), 441–450.

DiStefano, J. J., III., Stubberud, A. R., & Williams, I. J. (1990). *Schaum's outline of theory and problems of feedback and control systems*. New York: McGraw-Hill.

Dorigo, M., & Şahin, E. (2004). Guest editorial. *Autonomous Robots*, *17*, 111–113.

Downing, H. A., & Jeanne, R. L. (1988). Nest construction by the paper wasp, *Polistes*: A test of stigmergy theory. *Animal Behaviour*, *36*, 1729–1739.

Dutton, K., Thompson, S., & Barraclough, B. (1997). *The art of control engineering*. Harlow: Addison Wesley.

Franks, N. R. (1999). Information flow in the social domain: How individuals decide what to do next. In C. Detrain, J.-L. Deneubourg, & J. Pasteels (Eds.), *Information processing in social insects* (pp. 101–110). Basel: Birkhäuser.

Franks, N. R., & Deneubourg, J.-L. (1997). Self-organizing nest construction in ants: Individual worker behaviour and the nest's dynamics. *Animal Behaviour*, *54*, 779–796.

Franks, N. R., Pratt, S. C., Mallon, E. B., Britton, N. F., & Sumpter, D. J. T. (2002). Information flow, opinion polling and collective intelligence in house-hunting social insects. *Philosophical Transactions of the Royal Society of London, B 357*, 1567–1583.

Franks, N. R., Wilby, A., Silverman, B. W., & Tofts, C. (1992). Self-organizing nest construction in ants: Sophisticated building by blind bulldozing. *Animal Behaviour*, *44*, 357–375.

Gordon, D. M. (1996). The organization of work in social insect colonies. *Nature*, *380*, 121–124.

Grassé, P.-P. (1959). La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes Natalensis* et *Cubitermes Sp*. La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes Sociaux*, *6*, 41–80.

Holland, O. E. (1996). The use of social insect based control methods with multiple robot systems. In *Proceedings of UKACC International Conference on Control '96* (pp. 1260–1265). London, UK: IEE.

Holland, O. E., & Melhuish, C. (1999). Stigmergy, self-organization, and sorting in collective robotics. *Artificial Life*, *5*, 173–202.

Howsman, T. G., O'Neil, D., & Craft, M. A. (2004). A stigmergic cooperative multi-robot control architecture. In J. Pollack, M. Bedau, P. Husbands, T. Ikegami, & R. A. Watson (Eds.), *Artificial life IX: Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems* (pp. 88–93). Cambridge, MA: MIT Press.

Jeanne, R. L. (1999). Group size, productivity, and information flow in social wasps. In C. Detrain, J.-L. Deneubourg, & J. Pasteels (Eds.), *Information processing in social insects*. Basel: Birkhäuser.

Jones, C., & Matarić, M. J. (2003). From local to global behavior in intelligent self-assembly. In *Proceedings of 2003 IEEE International Conference on Robotics and Automation (ICRA'03)* (pp. 721–726). IEEE.

Kowadlo, G., & Russell, R. A. (2004). To näively smell as no robot has smelt before. In *Proceedings of 2004 IEEE Conference on Robotics, Automation and Mechatronics (RAM'04)* (pp. 893–903). IEEE.

Krieger, M. J. B., Billeter, J.-B., & Keller, L. (2000). Ant-like task allocation and recruitment in cooperative robots. *Nature*, *406*, 992–995.

Kube, C. R., & Zhang, H. (1997). Task modelling in collective robotics. *Autonomous Robots*, *4*, 53–72.

Kurabayashi, D. (1999). Toward realization of collective intelligence and emergent robotics. In *Proceedings of 1999 IEEE International Conference on Systems, Man, and Cybernetics (SMC'99)* (Vol. 4, pp 748–753). IEEE.

Ladley, D., & Bullock, S. (2005). The role of logistic constraints in termite construction of chambers and tunnels. *Journal of Theoretical Biology*, *234*, 551–564.

Li, G., & Zhang, H. (2005). A rectangular partition algorithm for planar self-assembly. In *Proceedings of 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'05)* (pp. 2324–2329). IEEE.

Melhuish, C. (2000). Autostruosis: Construction without explicit planning in microrobots—a social insect approach. In T. Gomi (Ed.), *Evolutionary Robotics III: From Intelligent Robots to Artificial Life* (Vol. 3). Kanata, ON: AAI Books.

Melhuish, C., Welsby, J., & Edwards, C. (1999). Using templates for defensive wall building with autonomous mobile ant-like robots. In U. Nehmzow & C. Melhuish (Eds.), *Proceedings of Towards Intelligent Mobile Robots (TIMR'99)*.

Parker, C. A. C., Zhang, H., & Kube, C. R. (2003). Blind bulldozing: Multiple robot nest construction. In *Proceedings* *of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'03)* (pp. 2010–2015). IEEE.

Pratt, S. C. (1998). Decentralized control of drone comb construction in honey bee colonies. *Behavioral Ecology and Sociobiology*, *42*, 193–205.

Robson, S. K., & Traniello, J. F. A. (1999). Key individuals and the organization of labor in ants. In C. Detrain, J.-L. Deneubourg, & J. Pasteels (Eds.), *Information processing in social insects*. Basel: Birkhäuser, 239–259.

Seeley, T. D. (1989). Social foraging in honey bees: How nectar foragers assess their colony's nutritional status. *Behavioral Ecology and Sociobiology*, *24*, 181–199.

Seeley, T. D., & Tovey, C. A. (1994). Why search time to find a food-storer bee accurately indicates the relatives rates of nectar collecting and nectar processing in honey bee colonies. *Animal Behaviour*, *47*, 311–316.

Spears, W. M., Spears, D. F., Hamann, J. C., & Heil, R. (2004). Distributed, physics-based control of swarms of vehicles. *Autonomous Robots*, *17*, 137–162.

Stewart, R. L., & Russell, R. A. (2003). Emergent structures built by a minimalist autonomous robot using a swarm-inspired template mechanism. In H.A. Abbass & J. Wiles (Eds.), *Proceedings of The First Australian Conference on Artificial Life (ACAL'03)* (pp. 216–230). Canberra, Australia: UNSW.

Stewart, R. L., & Russell, R. A. (2004). Building a loose wall structure with a robotic swarm using a spatio-temporal varying template. In *Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04)* (pp. 712–716). IEEE.

Stewart, R. L., & Russell, R. A. (2005). A generalised technique for building 2d structures with robot swarms. In H. A. Abbass, T. Bossomaier, & J. Wiles (Eds.), *Recent advances in artificial life—advances in natural computation*, *Vol. 3*. Singapore: World Scientific, 265–277.

Theraulaz, G., & Bonabeau, E. (1995). Modelling the collective building of complex architectures in social insects with lattice swarms. *Journal of Theoretical Biology*, *177*, 381–400.

Theraulaz, G., Bonabeau, E., & Deneubourg, J.-L. (1999). The mechanisms and rules of coordinated building in social insects. In C. Detrain, J.-L. Deneubourg, & J. Pasteels (Eds.), *Information processing in social insects* (pp. 309–330). Basel: Birkhäuser.

Theraulaz, G., Gautrais, J., Camazine, S., & Deneubourg, J.-L. (2003). The formation of spatial patterns in social insects: From simple behaviours to complex structures. *Philosophical Transactions of the Royal Society of London, A*, *361*, 1263–1282.

Wawerla, J., Sukhatme, G. S., & Matarić, M. J. (2002). Collective construction with multiple robots. In *Proceedings of 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)* (pp. 2696–2701). IEEE.

Werfel, J. K. (2004). Building blocks for multi-robot construction. In R. Alroni (Ed.) *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS'04)*.

Werfel, J. K., Bar-Yam, Y., & Nagpal, R. (2005). Building patterned structures with robot swarms. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI'05)* (pp. 1495–1502).

Wilson, E. O. (1975). *Sociobiology: The new synthesis*. Cambridge, MA: The Belknap Press of Harvard University Press.

Wilson, E. O., & Hölldobler, B. (1988). Dense heterarchies and mass communication as the basis of organization in ant colonies. *Trends in Ecology and Evolution*, *3*, 65–68.

Wilson, M., Melhuish, C., Sendova-Franks, A. B., & Scholes, S. (2004). Algorithms for building annular structures with minimalist robots inspired by brood sorting in ant colonies. *Autonomous Robots*, *17*, 115–136.

About the Authors

Robert Stewart completed a B.E. (Hons.) in the field of electrical and computer systems engineering at Monash University in 2001. He is currently pursuing a Ph.D. in robotics at the Intelligent Robotics Research Centre (Monash University) and is studying the building of structures with robotic swarms.

R. Andrew Russell received his B. Eng. and Ph.D. degrees in 1972 and 1976, respectively, from the University of Liverpool, UK. He is currently Mechatronics Course Director at Monash University, in Australia. His research interests include many aspects of mechatronics and intelligent robotics including humanoid robotics, robot tactile sensing, olfactory sensing for robots, the design of robotic mechanisms and robot learning. He has written books on robot tactile sensing and odor sensing for mobile robots as well as authoring over 100 other publications. *Address*: Centre for Perceptive and Intelligent Machines in Complex Environments: Intelligent Robotics, Monash University, Clayton campus, Wellington Road, Clayton, Victoria 3800, Australia. *E-mail*: andy.russell@eng.monash.edu.au