AMORPHOUS COMPUTING AND SWARM INTELLIGENCE

A SEMINAR REPORT

Submitted by

JEEBHA B PRASAD

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN
COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF ENGINEERING

COCHIN UNIVERSITY OF SCIENCE & TECHNOLOGY,

KOCHI-682022

NOVEMBER 2008

DIVISION OF COMPUTER SCIENCE & ENGINEERING

SCHOOL OF ENGINEERING

COCHIN UNIVERSITY OF SCIENCE & TECHNOLOGY

KOCHI-682022

BONAFIDE CERTIFICATE

Certified that this seminar report "**AMORPHOUS COMPUTING AND SWARM INTELLIGENCE** "is the bonafide work of **JEEBHA B PRASAD** who carried out the seminar under my supervision.

Mr. Pramod Pavithran	**Dr. David Peter**
SEMINAR GUIDE	**HEAD OF THE DIVISION**
Reader	Division of Computer Engineering
Division of Computer Engineering	School Of Engineering
School Of Engineering	Cochin University Of Science
Cochin University of Science	Technology
& Technology	

ACKNOWLEDGEMENT

First and foremost I thank almighty for his blessings. I sincerely express my gratitude to my seminar guide, Mr. Pramod Pavithran, Reader, CUSAT, for his proper guidance and valuable suggestions. Am equally indebted to Dr. David Peter, the HOD, Computer Engineering division and other faculty members for giving me such an opportunity to learn and present this seminar. If not for the above mentioned persons my seminar would never have been completed successfully. I once again extend my sincere thanks to all of them

JEEBHA B PRASAD

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS, ABBREVIATIONS AND NOMENCALATURE

1. ACO	Ant Colony Optimization
2. ACS	Ant Colony System
3. ABC	Ant Based Control
4. ANTS	Autonomous Nano Technology Swarms
5. TSP	Traveling Salesman Problem
6. TRL	Technology Readiness Level
7. MEMS	Micro-Electro-Mechanical Systems

ABSTRACT

Amorphous computing consists of a multitude of interacting computers with modest computing power and memory, and modules for intercommunication. These collections of devices are known as swarms. The desired coherent global behavior of the computer is achieved from the local interactions between the individual agents. The global behavior of these vast numbers of unreliable agents is resilient to a small fraction of misbehaving agents and noisy and intimidating environment. This makes them highly useful for sensor networks, MEMS, internet nodes, etc.

The ideas for amorphous computing have been derived from swarm behavior of social organisms like the ants, bees and bacteria. A certain level of intelligence, exceeding those of the individual agents, results from the swarm behavior. Swarm Intelligence may be derived from the randomness, repulsion and unpredictability of the agents, thereby resulting in diverse solutions to the problem. There are no known criteria to evaluate swarm intelligence performance. Swarm Intelligence relies upon stigmergic principles in order to solve complex problems using only simple agents.

1. INTRODUCTION

1. 1.**S**WARM **I**NTELLIGENCE

During the course of the last 20 years, researchers have discovered the variety of interesting insect and animal behaviors in nature. A flock of birds sweeps across the sky. A group of ants forages for food. A school of fish swims, turns, flees together, etc. We call this kind of aggregate motion **S**warm behavior. Recently, biologists and computer scientists have studied how to model biological swarms to understand how such social animals interact, achieve goals, and evolve. Furthermore, engineers are increasingly interested in this kind of swarm behavior since the resulting swarm intelligence can be applied in optimization (e.g. in telecommunication systems) , robotics track patterns in transportation systems, and military applications .

A high-level view of a swarm suggests that the N agents in the swarm are cooperating to achieve some purposeful behavior and achieve some goal. This apparent collective intelligence seems to emerge from what are often large groups of relatively simple agents. The agents use simple local rules to govern their actions and via the interactions of the entire group, the swarm achieves its objectives. A type of self-organization emerges from the collection of actions of the group.

Swarm intelligence is the emergent collective intelligence of groups of simple autonomous agents. Here, an autonomous agent is a subsystem that interacts with its environment, which probably consists of other agents, but acts relatively independently from all other agents. The autonomous agent does not follow commands from a leader, or some global plan . For example, for a bird to participate in a flock, it only adjusts its movements to coordinate with the movements of its flock mates, typically its neighbors that are close to it in the flock. A bird in a flock simply tries to stay close to its neighbours, but avoid collisions with them. Each bird does not take commands from any leader bird since there is no lead bird. Any bird can fly in the front, center or back of the swarm. Swarm behavior helps birds take advantage of several things including protection from predators (especially for birds in the middle of the flock), and searching for food (as each bird is essentially exploiting the eyes of every other bird).

1.2. **B**IOLOGICAL BASIS AND **A**RTIFICIAL LIFE

Researchers try to examine how collections of animals, such as flocks, herds and schools, move in a way that appears to be orchestrated. A flock of birds moves like a well choreographed dance troupe. They veer to the left in unison, and then suddenly they may all dart to the right and swoop down toward the ground. How can they coordinate their actions so well? In 1987, Reynolds created a **b**oid model, which is a distributed behavioral model, to simulate on a computer the motion of a flock of birds . Each boid is implemented as an independent actor that navigates according to its own perception of the dynamic environment. A boid must observe the following rules. First, the “avoidance rule" says that a boid must move away from boids that are too close, so as to reduce the chance of in-air collisions. Second, the “copy rule" says a boid must go in the general direction that the flock is moving by averaging the other boids' velocities and directions. Third, the “center rule" says that a boid should minimize exposure to the flock's exterior by moving toward the perceived center of the flock. Flake added a fourth rule, “view," that indicates that a boid should move laterally away from any boid the blocks its view. This boid model seems reasonable if we consider it from another point of view, that of it acting according to attraction and repulsion between neighbours in a flock. The repulsion relationship results in the avoidance of collisions and attraction makes the flock keep shape, i.e., copying movements of neighbours can be seen as a kind of attraction. The center rule plays a role in both attraction and repulsion. The swarm behaviour of the simulated flock is the result of the dense interaction of the relatively simple behaviours of the individual boids. To summarize, the flock is more than a set of birds; the sum of the actions results in coherent behaviour.

One of the swarm-based robotic implementations of cooperative transport is inspired by cooperative prey retrieval in social insects. A single ant finds a prey item which it cannot move alone. The ant tells this to its nest mate by direct contact or trail-laying. Then a group of ants collectively carries the large prey back. Although this scenario seems to be well understood in biology, the mechanisms underlying cooperative transport remain unclear. Roboticists have attempted to model this cooperative transport. For instance, Kube and Zhang introduce a simulation model including stagnation recovery with the method of task modeling. The collective behavior of their system appears to be very similar to that of real ants.

1.3.**A**MORPHOUS **C**OMPUTING

Amorphous computing represents an analog approach to swarm system design.

In amorphous computing systems, a colony of cells cooperates to form a multi-cellular organism under the direction of a program (loosely called a genetic program) that is shared by all members of the colony.

The objective of amorphous computing is the creation of algorithms and techniques for the understanding of programming materials. Essentially, amorphous computing seems to incorporate the biological mechanisms of individual cells into systems that exhibit the expressive power of digital logic circuits. Stigmergy in such systems can be either marker-based or sematectonic and be either scalar or vector in extent. An amorphous computing medium is a system of irregularly placed, asynchronous, locally interacting computing elements. The medium is modelled as a collection of “computational particles” sprinkled irregularly on a surface or mixed throughout a volume. In essence, the computational assembly forms an ad hoc network.

Research into self-healing structures, circuit formation, programmable self-assembly and selforganizing communication networks are a small sample of the work undertaken.

1.4. **E**VALUATION OF **S**WARM **I**NTELLIGENT **S**YSTEM

Although many studies on swarm intelligence have been presented, there are no general criteria to evaluate a swarm intelligent system's performance. They

proposed measures of fault tolerance and local superiority as indices. They compared two swarm intelligent systems via simulation with respect to these two indices. There is a significant need for more analytical studies.

1.4.1 **S**TABILITY OF SWARMS

1.4.1.1 BIOLOGICAL MODELS

In biology, researchers proposed “continuum models" for swarm behavior based on nonlocal interactions. The model consists of integro-differential advection-diffusion equations, with convolution terms that describe long range attraction and repulsion. They found that if density dependence in the repulsion term is of a higher order than in the attraction term, then the swarm has a constant interior density with sharp edges as observed in biological examples. They did linear stability analysis for the edges of the swarm.

2. PRINCIPLES OF SWARM INTELLIGENCE

2.1 **O**VERVIEW

The objective of this engagement is to provide a comprehensive assessment of the state of the art in Swarm Intelligence; specifically the role of **stigmergy** in distributed problem solving. In order to do this, working definitions have to be provided along with the essential properties of systems that are swarm-capable; i.e. problem solving is an emergent property of a system of simple agents.

The principle of stigmergy implies the interaction of simple agents through a common medium with no central control. This principle implies that querying individual agents tells one little or nothing about the emergent properties of the system. Consequently, simulation is often used to understand the emergent dynamics of stigmergic systems. Stigmergic systems are typically stochastic in nature; individual actions being chosen probabilistically from a limited behavioural repertoire. Actions performed by individual agents change the nature of the environment; for example a volatile chemical called a pheromone is deposited. This chemical signal is sensed by other agents and results in modified probabilistic choice of future actions.

The advantages of such a system are clear. Being a system in which multiple actions of agents are required for a solution to emerge, the activity of an individual agent is not as important. That is, stigmergic systems are resilient to the failure of individual agents and, more importantly still react extremely well to dynamically changing environments. Optimal use of resources is often a significant consideration in designing algorithms. Another stigmergic system -- the raid army ant model – efficiently and effectively forages for food using pheromone-based signalling. In a raid army ant system, agents develop a foraging front that covers a wide path, leading to extremely effective food finding. This model has military value in that it could potentially be exploited as a series of mechanisms for searching for land mines, a problem that, tragically, is all too common in parts of the world.

A third stigmergic model of military interest is that of flocking or aggregation. Here, large numbers of simple agents can be made to move through a space filled with obstacles (and potentially threats) without recourse to central control. The environmental signals here are the position and velocities of the agents themselves.

The utility of this model is that tanks could potentially be made to move across a terrain taking into account only tanks that are close by. A similar use of the model might be the self-organization of a squadron of flying drones.

2.2.EMERGENT PROBLEM SOLVING

2.2.1.OVERVIEW

Emergent problem solving is a characteristic of swarm systems. Emergent problem solving is a class of problem solving where the behavior of individual agents is not goal directed; i.e. by looking at the behavior of single agents little or no information on the problem being solved can be inferred.

2.2.2 SWARM PROBLEM SOLVING

Swarm problem solving is a bottom-up approach to controlling and optimizing distributed systems. It is a mindset rather than a technology that is inspired by the behavior of social insects that has evolved over millions of years.

Peterson suggests that swarms calculate faster and organize better. Swarm systems are characterized by simple agents interacting through the environment using signals that are spatially (and temporally) distributed. By simple we mean that the agents possess limited cognition and memory; sometimes no memory at all. Furthermore, the behavior of individual agents is characterized by a small number of rules. In this document we consider the complexity (or simplicity) of an agent to be a function of the number of rules that are required to explain its behavior.

2.2.3.ADVANTAGES AND DISADVANTAGES

There are several advantages:

A. Agents are not goal directed; they react rather than plan extensively.

B. Agents are simple, with minimal behavior and memory.

C. Control is decentralized; there is no global information in the system.

D. Failure of individual agents is tolerated; emergent behavior is robust with respect to individual failure.

E. Agents can react to dynamically changing environments.

F. Direct agent interaction is not required.

Flexible	The colony respond to internal perturbations and external challenges
Robust	Tasks are completed even if some individuals fail
Scalable	From a few individuals to millions
Decentralized	There is no central control(ler) in the colony
Self organized	Paths to solutions are emergent rather than predefined

Table 2.1: Advantages of Swarm Systems

There are certain disadvantages:

A. Collective behavior cannot be inferred from individual agent behaviour. This implies that observing single agents will not necessarily allow swarm-defeating behavior to be chosen. (This can be viewed as an advantage too from an aggressive point of view).

B. Individual behavior looks like noise as action choice is stochastic.

C. Designing swarm-based systems is hard. There are almost no analytical mechanisms for design.

D. Parameters can have a dramatic effect on the emergence (or not) of collective behavior.

Behavior	Difficult to predict collective behavior from individual rules.
Knowledge	Interrogate one of the participants, it won’t tell you anything about the function of the group.
Sensitivity	Small changes in rules lead to different group-level behavior.
Action	Individual behavior looks like noise: how do you detect threats

Table 2.2: Disadvantages of Swarm systems

2.2.4. **C**REATING **S**WARMING SYSTEMS

A swarm-based system can be generated using the following principles:

1. Agents are independent, they are autonomous. They are not simply functions as in the case of a conventional object oriented system.
2. Agents should be small, with simple behaviors. They should be situated and capable of dealing with noise. In fact, noise is a desirable characteristic.
3. Decentralized – do not rely on global information. This makes things a lot more reliable.
4. Agents should be behaviorally diverse – typically stochastic.
5. Allow information to leak out of the system; i.e. introduce disorder at some rate.
6. Agents must share information – locally is preferable.
7. Planning and execution occur concurrently – the system is reactive.

The principles outlined above come from Parunak . More recently, the importance of gradient creation and maintenance has been stressed and that digital pheromones can be made to react in the environment, thereby creating new signals of use to other swarm agents.

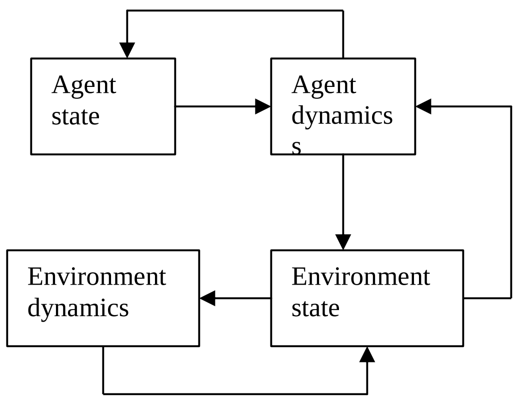


Figure 2.1 : Agent Environment Interaction

The above figure summarizes the interactions between agent and environment. Agent state along with environment state drives agent dynamics; i.e. agent action selection. Agent action selection changes environment state through the creation or modification of signals. Environment state is used as input to environment dynamics. The dynamics of the environment causes changes to occur in environment state. What is important in the above figure is that agent state is hidden – only the agent has access to it. Environment state is visible to the agent but has to be stored by the agent if it is to be reused at some later point in time when the agent has (presumably) moved to a different location.

2.2.5.**T**OOLS FOR INVESTIGATING SWARM SYSTEMS

As mentioned in a previous section, predicting the emergent behavior of swarm systems based upon the behavior of individual agents is generally not analytically tractable. Consequently, agent-based simulation is used to investigate the properties of these systems. There are two tools useful for such investigations.

2.2.5.1.**N**ETLOGO[1]

NetLogo is a simple agent simulation environment based upon StarLogo, an environment by Resnick and described in his book entitled, “Turtles, Termites and Traffic Jams”. Users program using agents and patches (the environment). In NetLogo, the environment has active properties and is ideal in its support of stigmergy as agents can easily modify or sense information of the local patch or patches within some neighborhood. Unlike conventional programming languages, the programmer does not have control over agent execution and cannot assume uninterrupted execution of agent behavior. A fairly sophisticated user interface is provided and new interface components can be introduced using a drag-and-drop mechanism. Interaction with model variables is easily achieved through form-based interfaces. The user codes in NetLogo’s own language, which is simple and type-free (i.e. dynamically bound).

2.2.5.2. **R**EPAST[2]

Repast is a more sophisticated Java-based simulation environment that forces the developer to provide Java classes in order to create an application.

[1] Freely available from ‘http://ccl.northwestern.edu/netlogo/’.

[2] Available from ‘http://repast.sourceforge.net/’

The Recursive Porous Agent Simulation Toolkit (Repast) is one of several agent modeling toolkits that are available. Repast borrows many concepts from the Swarm agent-based modeling toolkit . Repast is differentiated from Swarm since Repast has multiple pure implementations in several languages and built-in adaptive features such as genetic algorithms and regression. Repast is at the moment the most suitable simulation framework for the applied modeling of social interventions based on theories and data" . Of particular interest is the built-in support for genetic algorithms (which can be used to evolve controllers for robot swarms, for example) and sophisticated modeling neighbourhoods. Repast is widely used for social simulation and models in crowd dynamics, economics and policy making among others have been constructed.

3. APPLICATION OF SWARM INTELLIGENCE

3.1. **A**NT **C**OLONY **O**PTIMIZATION

Ant algorithms (also known as Ant Colony Optimization) are a class of metaheuristic search algorithms that have been successfully applied to solving NP hard problems . Ant algorithms are biologically inspired from the behaviour of colonies of real ants, and in particular how they forage for food. One of the main ideas behind this approach is that the ants can communicate with one another through indirect means (stigmergy) by making modifications to the concentration of highly volatile chemicals called pheromones in their immediate environment.

The Traveling Salesman Problem (TSP) is an NP complete problem addressed by the optimization community having been the target of considerable research . The TSP is recognized as an easily understood, hard optimization problem of finding the shortest circuit of a set of cities starting from one city, visiting each other city exactly once, and returning to the start city again. The TSP is often used to test new, promising optimization heuristics. Formally, the TSP is the problem of finding the shortest Hamiltonian circuit of a set of nodes. There are two classes of TSP problem: symmetric TSP, and asymmetric TSP (ATSP). The difference between the two classes is that with symmetric TSP the distance between two cities is the same regardless of the direction you travel; with ATSP this is not necessarily the case.

Ant Colony Optimization has been successfully applied to both classes of TSP with good, and often excellent, results. The ACO algorithm skeleton for TSP is as follows :

Procedure ACO algorithm for TSPs

1.Set parameters, initialize pheromone trails

2.while (termination condition not met) do

3.ConstructSolutions

4.ApplyLocalSearch % optional

5.UpdateTrails

6.end

end ACO algorithm for TSPs

ALGORITHM

Expanding upon the algorithm above, an ACO consists of two main sections: initialization and a main loop. The main loop runs for a user-defined number of iterations. These are described below:

Initialization

• Any initial parameters are loaded.

• Each of the roads is set with an initial pheromone value.

• Each ant is individually placed on a random city.

Main loop begins

Construct Solution

• Each ant constructs a tour by successively applying the probabilistic choice function and randomly selecting a city it has not yet visited until each city has been visited exactly once.

$$p_{ij}^{k}(t) = \frac{\left[\tau_{ij}(t)\right]^{\alpha} \cdot \left[\eta_{ij}\right]^{\beta}}{\sum_{l \in N_i^k} \left[\tau_{il}(t)\right]^{\alpha} \cdot \left[\eta_{il}\right]^{\beta}}$$

• The probabilistic function, $P^{k}_{ij}(t)$, is designed to favour the selection of a road that has a high pheromone value, tŶand high visibility value, ?Ŷwhich is given by, l/d_{ij}, where d_{ij} is the distance to the city. The pheromone scaling factor, a“ and visibility scaling factor, ßÇare parameters used to tune the relative importance of pheromone and road length in selecting the next city.

Apply Local Search

• Not used in Ant System, but is used in several variations of the TSP problem where 2-opt or 3-opt local optimizers are used.

Best Tour check

• For each ant, calculate the length of the ant's tour and compare to the best tour's length. If there is an improvement, update it.

Update Trails

• Evaporate a fixed proportion of the pheromone on each road.

• For each ant perform the "ant-cycle" pheromone update.

• Reinforce the best tour with a set number of "elitist ants" performing the "antcycle" pheromone update.

In the original investigation of Ant System algorithms, there were three versions of Ant System that differed in how and when they laid pheromone. They are:

• “Ant-density” updates the pheromone on a road traveled with a fixed amount after every step.

• “Ant-quantity” updates the pheromone on a road traveled with an amount proportional to the inverse of the length of the road after every step.

• “Ant-cycle” first completed the tour and then updates each road used with an amount proportional to the inverse of the total length of the tour.

Of the three approaches “Ant-cycle” was found to produce the best results and subsequently receives the most attention. It will be used for the remainder of this paper.

Main Loop Ends

Output

• The best tour found is returned as the output of the problem.

3.2 .ROUTING

Routing has been a significant area of research for swarm intelligence. Starting with Schonderwoerd in 1997, and Di Caro in 1998, the exploitation of the foraging behaviour of ants has been shown to significantly improve the quality of routing in networks. Most recently, research into ad hoc network routing has been active; with Di Caro (AntHocNet) having provided the most compelling research.

Ad hoc networks consist of autonomous self-organized nodes. Nodes use a wireless medium for communication, thus two nodes can communicate directly if and only if they are within each other’s transmission radius. Examples are sensor networks (attached to a monitoring station), rooftop networks (for wireless Internet access), and conference and rescue scenarios for ad hoc networks, possibly mobile. In a routing task, a message is sent from a source to a destination node in a given network. Two nodes normally communicate via other nodes in a multi-hop fashion. Swarm intelligence follows the behaviour of cooperative ants in order to solve hard static and dynamic optimization problems. Ants leave pheromone trails at nodes or edges which increases the likelihood of other ants to follow these trails. Routing paths are then found dynamically on the fly, using this so called notion of stigmergy.The ideas coming from existing swarm intelligence based routing in communication networks are incorporated into the wireless domain, with some new techniques which are typical for the wireless domain (such as flooding, use of position, monitoring traffic at neighbouring nodes) being incorporated.

3.2.1 General principles of routing

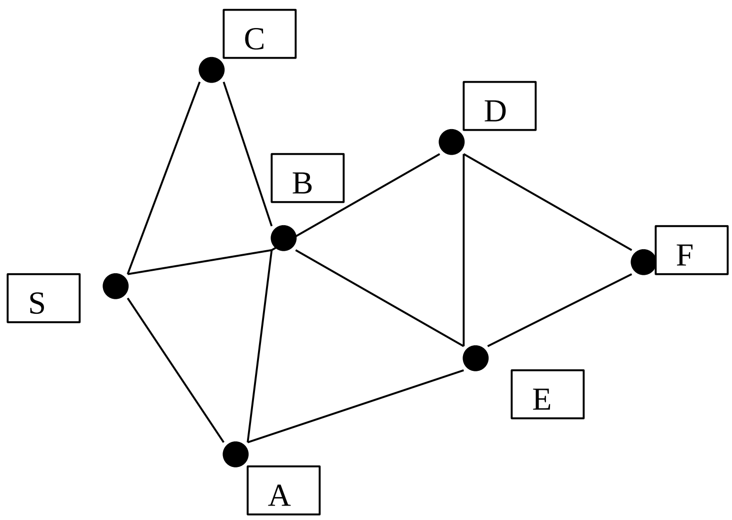


Figure 3.1 :Routing network

	A	**B**	**C**	**D**	**E**	**F**
A	**0.9**	**0.1**	**0.1**	**0.4**	**0.5**	**0.5**
B	**0.1**	**0.8**	**0.2**	**0.6**	**0.4**	**0.4**
C	**0.0**	**0.1**	**0.7**	**0.0**	**0.1**	**0.1**

Table 3.1: Routing Table for node S

Each node in the network has a routing table which helps it determine where to send the next packet or ant. These routing tables have the neighbours of the node as rows, and all of the other nodes in the network as columns. In Figure , we see an example of a network, and in Figure we see the routing table for node *S* in this network.

An ant or message going from node S to node F, for example, would consider the cells in column F to determine the next hop. Ants and messages can determine the next hop in a variety of ways. The next hop can be determined uniformly; which means that any one of the neighbours has an equally likely probability of being chosen. It can be chosen probabilistically, that is, the values in the routing table in column F are taken as the likelihoods of being chosen. Taking the highest value in the column of F could be another way of choosing the next hop. It could also be chosen randomly, which means choosing uniformly if there is no pheromone present, and taking the highest value if there is. There is also an exploratory way of choosing the next hop, which means taking a route with a value of 0 if one exists.

There are a few swarm intelligence (ant-based) routing algorithms developed for wired networks, and the most well known of which are **A**nt**N**et **[DD]** and **A**nt-**B**ased **C**ontrol (**ABC**) . The fundamental principle behind both AntNet and ABC is similar – they use ants as exploration agents. These ants are used for traversing the network node to node and updating routing metrics. A routing table is built based on the probability distribution functions derived from the trip times of the routes discovered by the ants. The approaches used in AntNet and ABC are, however, dissimilar – in AntNet, there are forward and backward ants, whereas in ABC, there is only one kind of ant. Another difference between AntNet and ABC is in the routing front. In ABC, the probabilities of the routing tables are updated as the ants visit the nodes, and are based on the life of the ant at the time of the visit; while in AntNet, the probabilities are only updated when the backward ant visits a node.

3.3.**C**OLLECTIVE **R**OBOTICS

3.3.1. **I**NTRODUCTION

Collective, or swarm-based, robotics is a relatively new field. One of the earliest researchers in the field was Kube who demonstrated that simple robots with no inter-robot communication could collectively push heavy objects and cluster objects in a manner similar to ants. His robots were homogeneous.

Martinoli provides a very good introduction to the problems of creating swarms of robots that exhibit complex distributed collective problem solving strategies. More recently, March 2005, the Swarm Bots project lead by Marco Dorigo completed its 3.5 year investigation into the creation of teams of small robots using stigmergy.

3.3.2 **A**UTONOMOUS **N**ANO**T**ECHNOLOGY **S**WARMS

NASA's autonomous nanotechnology swarms (ANTS) creates communities of intelligent teams of agents where redundancy is built in. The ANTS architecture uses a biologically inspired approach, with ants as primary inspiration. It is the most sophisticated of all of the stigmergic systems currently in design. Swarms of up to 1000 nodes will be deployed on deep space missions to study asteroids, with sub-swarms of 100 nodes being independently tasked with given mission parameters. Several classes of swarm unit have been defined with measurement (imaging, for example), communication and leadership characteristics. A generic worker class has also been designed. The ANTS project timeline extends beyond 2030 when the first missions are envisaged. However, several important engineering concepts have already been developed. In the ANTS system, the basic physical structure is a tetrahedron that flexes, changing shape causing a tumbling motion thereby allowing movement over a surface. Tetrahedral structures are used at all levels of the ANTS design, the designers arguing that this structure is one of the most stable naturally-occurring structures. The ANTS system consists of small, spatially distributed units of autonomous, redundant components. These components exhibit high plasticity and are organized as hierarchical (multilevel, dense heterarchy) and inspired by the success of social insect colonies. The ANTS system uses hybrid reasoning – symbolic and neural network systems – for achieving high levels of autonomous decision making.

3.3.3 **S**WARM **B**OTS

The main scientific objective of the recently completed Swarm-bots project was to study a novel approach to the design and implementation of self-organising and self-assembling artifacts. This novel approach used as theoretical roots recent studies in swarm intelligence, that is, studies of the self-organizing and self-assembling capabilities shown by social insects and other animal societies employing stigmergic principles extensively.

The main tangible objective of the project was the demonstration of the approach by means of the construction of at least one of such artifact. A swarm-bot was constructed. That is, an artifact composed of a number of simpler, insect-like, robots (s-bots), built out of relatively cheap components, capable of self-assembling and self-organizing to adapt to its environment. Three distinct components were developed: s-bots (hardware), simulation (software), and swarm-intelligence-based control mechanisms (software). A set of hardware s-bots that can self assemble into a shape-changing swarm-bot were developed that were capable of accomplishing a small number of tasks. Tasks completed were dynamic shape formation and shape changing and navigation on rough terrain. In both cases, teaming is crucial as a single sbot cannot accomplish the task and the cooperative effort performed by the s-bots aggregated in a swarm-bot is necessary.

3.4.MECHATRONICS

Mechatronics is the discipline of building reconfigurable robots. An excellent resource on the subject can be found at Colorado State. Robots are made out of modules, which could crudely be described as intelligent Lego bricks. Plugging the bricks (or modules) together in particular ways allows a mechatronic robot to more or less effectively solve a problem such as moving over terrain of a given class; e.g. swamp or very rocky. In the mechatronic domain stigmergy is represented as perception of self. While the Swarm-bot project can be thought of as fitting into this category, mechatronic research focuses on the assembly, re-assembly and reconfiguration of simpler units. Continuing with the comparison with the Swarm Bot project, mechatronic research is concerned with the construction of an s-bot rather than the swarm-bot. Stigmergy in this area is typically sematectonic – the robot/module configurations being used to drive the configuration process.

4. FUTURE RESEARCH AND TECHNOLOGY ASSESSMENTS

4.1.INTRODUCTION

This section deals with potential future research that might facilitate the introduction of stigmergic principles into military and security systems. The tables below provide an extended assessment over the timeframe 2005-2030.

TECHNOLOGY FORECAST

Stigmergy

	Timeframe 2005	2010	2015	2020	2025	2030
a. Sensor Networks Hardware						
b. Sensor Networks Algorithms						
c. Digital Pherome Applications						
d. Smart Robots						
e. Smart Materials						
f. Smart Applications						

Technology Readiness
(Colour codes defined above) *

Definitions:

a. Sensor Network Hardware
Sensors capable of deployment in harsh military environments are the goal
The sensors should be deployable by ground troops, or by UAVs

b. Sensor Network Algorithms
Algorithms refer to routing protocols. Stability has to be proven.
Algorithms for information dispersal.

c. Digital Pheromone Applications
Tracking and target surveillance applications with superior performance to traditional solutions.

d. Smart Robots
Robots that are capable of threat assessment and self-deployment.
Robots should support reconfigurability and replication.

e. Smart Materials
Materials with large numbers of embedded processors capable of autonomous computation.

f. Smart Applications
Optimization applications for military logistical problems

Table 4.1:Future assessments

A wide range of technology levels have been observed, from TRL 1 through to TRL 8. Digital Pheromone target tracking system should be considered the most mature technology at TRL 7/8 having flown in operational experimental conditions. The Swarm-bots project –arguably the most exciting project from a robotics perspective – is assessed at TRL 4/5.Mechatronic research is generally at TRL 4. The algorithms derived from the Ant Colony Optimization metaheuristic ("Smart Algorithms") should be rated at TRL 2/3 (only because physical systems are not generated in this environment). The MANET routing algorithm research should be rated at TRL 3. Routing algorithms for sensor networks would also attract a TRL rating of 3. Sensor technology achieves the rating of TRL 5/6.

**** Technology Readiness Levels***

Code	***Colour***	***Definition***
1		Basic principles observed and reported
2		Technology concept and/or application formulated
3		Analytical and experimental critical function and/or characteristic
4		Component and/or breadboard validation in laboratory environment
5		Component and/or breadboard validation in relevant environment
6		System/subsystem model or prototype demonstration in a relevant environment
7		System prototype demonstration in a operational environment
8		Actual system completed and 'flight qualified' through test and demonstration
9		Actual system 'flight proven' through successful mission operations

Table 4.2:Technology Readiness Assessment

5. CONCLUSION

Very little theory exists for swarm-based systems. No robust systems should be deployed before we understand fundamental properties of stigmergic systems.

First, sensor network simulation tools need to be constructed; theoretical analysis should occur in parallel possibly providing bounds on performance when analytical closed-form solutions can not be easily obtained. The existing body of sensor network routing algorithms are either incompletely specified or analyzed; considerable work remains to be done. Scenario generators should be built in order to evaluate the effectiveness of the sensor network – the environment – in conjunction with agents whose behaviour is stigmergically-driven. In order to achieve this, an extensible, reusable agent framework should be developed that captures the patterns documented in this report, suitably augmented with existing intelligent agent algorithms for military applications. Research into the problem of combining stigmergic signals – sensor fusion – also needs to be conducted. Furthermore, other stigmergic patterns should be captured and added as research in theoretical biology provides insight into other social insect behaviours.

Second, technologies for wide-spread cost-effective sensor networks need to be developed.

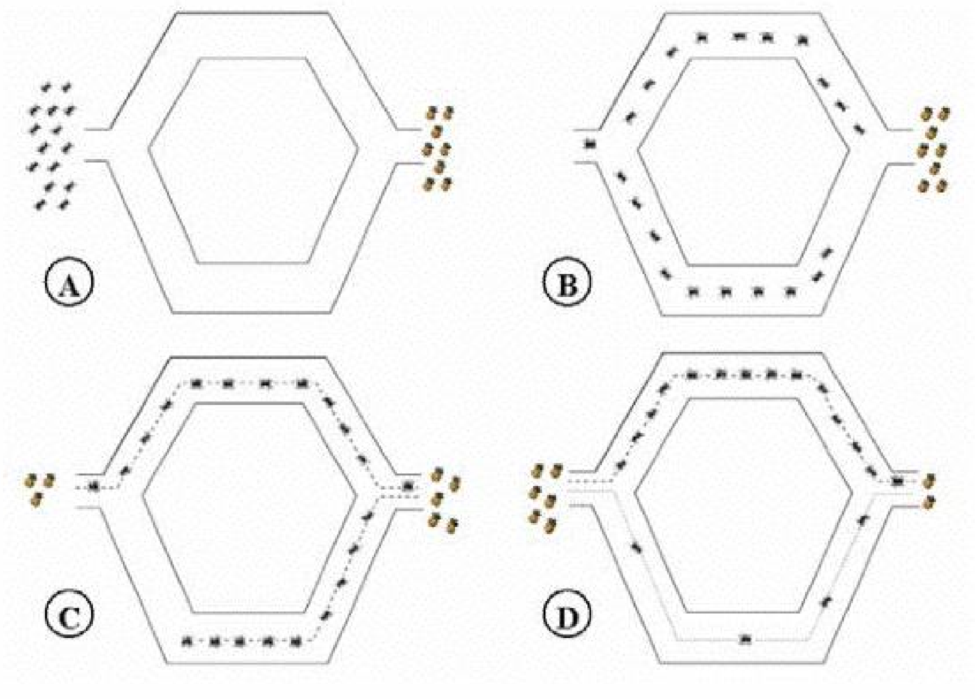
Third, intelligent materials research needs to be undertaken. Sensors woven into the fabric of clothing are relevant here. Also the work on Amorphous Computing may be of interest as it provides the potential for materials capable of self repair. Self-repairing materials have obvious applications in the autonomous repair of unmanned autonomous vehicles, for example.

Finally, research into reconfigurable and self-reproducing robots should be supported. The goal should be to understand, fabricate and deploy modules in the battlefield setting that can be used as building blocks for the repair and reproduction of unmanned autonomous vehicles in situation. Owing to the importance of sensor networks in the battlefield of the future, an in depth review of routing algorithms for ad hoc networks has been provided. Swarm Bots project shows significant promise for the engineering of future robot swarms.

The most mature military systems using stigmergic principles – rated at TRL 8 – have been described and demonstrate conclusively that marker-based stigmergy ensures very good information fusion and processing in a battlefield scenario. Related work – referenced but not described – indicates that the systems evaluated are stable, can be effectively simulated and scale to large number of unmanned autonomous vehicles.
To conclude, the body of work on swarm intelligence found in the literature and social insects observed in nature, indicate that robust, scalable and engineering solutions can be created. What remains is the problem of developing a detailed research agenda and then funding it.

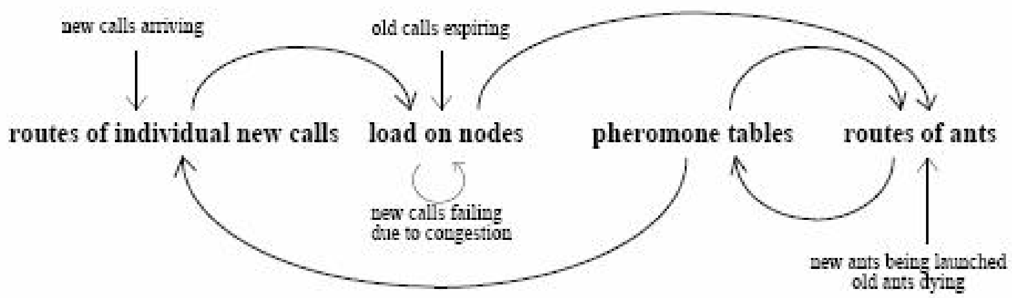
APPENDIX 1

ACO DIAGRAM



APPENDIX 2

ROUTING



APPENDIX 3

TYPICAL ANT ROBOT

BIBLIOGRAPHY

1.Blum C., Theoretical and Practical Aspects of Ant Colony Optimization, IOS Press, ISBN
1586034322, November 2004.
2. Bonabeau E., Dorigo M. and Theraulaz G., Swarm Intelligence, Oxford University Press
US, ISBN 0195131584, September 1999.
3.Dorigo M. and Stutzle T., Ant Colony Optimization, MIT Press, ISBN 0262042193, July
2004.
4.Harold Abelson, Gerald Sussmann, their 1999 MIT memorandum, "Amorphous Computing"
(reprinted by the Communications for the Association of Computing Machinery earlier this year).
5.Kennedy J. and Eberhart R. C., Swarm Intelligence, Morgan Kaufman, ISBN
1558605959, 2001.