Swarm Intelligence

Franco Zambonelli

March 2010

Outline

- Part 1: What is Swarm Intelligence
  - Basic Definitions
  - Individual Intelligence vs. Swarm Intelligence
- Part 2: Examples and Applications of Swarm Intelligence
  - Firefly synchronization
  - Slime-mold aggregation
  - Ant foraging and sorting
  - Termites nest building
- Conclusions and Open Issues

Part 1

- What is swarm intelligence

What is Swarm Intelligence

- Definition
  - Intelligent behaviors from a large number (i.e., a swarm) of simple individuals
  - Collectively doing something seemingly “intelligent” or “useful”
  - Where no one of the individual can claim intelligence
- So the intelligence is not in the composition of simple intelligences
  - Rather, intelligence “emerges” as a consequences of the interactions
  - Is a property of the system, not of its components
  - It is the system in its whole that does something intelligence

Where is Swarm Intelligence?

- Actually, there are a number of systems which seems to exhibit swarm intelligence
  - **Animal colonies** and specifically
  - **Insect colonies** like ants, termites, and bees
  - **Bacteria** (e.g., the Dictostelyum), which appear able to act in a finalized way
  - **The Brain**: intelligence and mind arises form the interaction of simple neurons
  - **The Cell**: homeostasis and the capability of adapting and reproducing arise form protein interactions
- Therefore
  - Swarm intelligence seems not to be an “accident” but rather a property of a variety of systems
  - Definitely, evolution has played an important role in this

The Concept of "Emergence"

- The behavior of swarm intelligent systems if often said to be an "emergent behavior"
  - It does not arise from a rationale choice
  - It does not arise from an engineering finalized analysis
  - No one and nothing in the system says: I will do that because this will lead to a specific behavior of the system
  - So, intelligence seems to magically "emerge"
- Clearly, **emergence is in the eye of the observer**
  - The individual in the system have no global perspective
  - They are not aware of what's globally happening
  - They are not aware they are doing something intelligent

Self-Organization

- The “intelligence” is observed in terms of some high-level global scale organized behavior
  - Spontaneously emerged in the system
  - Spanning outside the typical local capability of sensing/effecting of individuals
- From Local to Global
  - Typically, in swarm systems, interactions and capability are local
  - Nevertheless, the behavior observed has some sorts of global organization
- From order to disorder
  - Typically, a system may start in a very disordered state
  - And evolve in time to reach “order”, i.e., some global observable patterns of organization in structure and/or in activities

Swarms and Adaptivity

- Swarm intelligence is often very adaptive
  - It can achieve self-organization independently of the contingencies of the environment
  - It can re-shape the global behavior to react to environmental dynamics
  - Without losing global organization, but rather having the system re-organize itself
- In other words, swarm intelligent systems are adaptable to the context
  - They perceive somewhat changes in the context
  - And re-shape their behavior accordingly

Mechanisms of Swarm Intelligence

- We can characterize a swarm system in terms of
  - Individuals, interactions, environment
- Activities of individuals
  - Perceive local properties of the environment
  - React on the basis of simple perception-reaction models (e.g., reactive agents)
  - Affect somewhat the properties of the environment
  - Move in the environment
  - Very often, act based on some stochastic parameter
- Interaction mechanisms
  - Typically, interactions are not direct communications
  - But are indirect forms of interactions
  - I “smell” the environment (“stigmergy”)
  - I “see” what the others are doing (“behavioral interactions”)
- Environmental mechanisms
  - The environment in which individuals live may contribute with its own activities
  - May possess its own properties
  - May have active processes to dynamically vary this properties

The Role of Feedbacks

- The capability of self-organization in swarm system includes contrasting phenomena of feedbacks
- **Positive feedback**: re-enforcement or activation
  - The behavior of an individual (or its effect in the environment)
  - May solicits other individuals to do the same
  - So that several individual starts behaving in a seemingly organized way
- **Negative feedback**: control or inhibition
  - The behavior of some individuals, or processes in the environment
  - May avoid that all individuals converge to the same behavior or to the same state
  - And avoid the system to reach a stable equilibrium
- There is a continuous tension between re-inforcement and inhibition
  - And this is what actually happens in most known phenomena of self-organization, e.g., cellular automata, markets, complex networks, etc.

The Role of Randomness

- Often, swarm intelligent behavior rely on stochastic choices by its individuals
  - Individuals have a certain probability to behave in specific ways
  - The behavior of individuals is a balanced between a simple perception-reaction model and a random model
- This enables
  - To reach self-organization form any initial configuration
  - To escape (as known from operations research) from local minima (e.g. to reach the optimal self-organization)
  - To react to changed situations in the environment
- The latter point is very important:
  - Together with feedbacks, it enables the system to exhibit adaptivity
  - No configuration is the final one
  - Randomness enables exploring always some alternative
- For instance, when the conditions change and a reached form of self-organization is no longer satisfying
  - Randomness ensure the better solution will be found
  - Positive feedback ensure that will become the new configuration

The Role of Mass

- For swarm intelligent system to achieve their global self-organization
  - In a robust and adaptive way
- It is necessary to have large masses of agents
  - To explore a search space in full
  - To tolerage local faults (single components do not matter!)
  - To enable adaptivity
    - While a large percentage of components are doing their work, the other may be “lost” in searching alternate solutions

Swarm Systems vs. Multiagent Systems

- Actually, swarms are ensembles of simple agents, i.e., multi-simple_agent systems
  - Components are autonomous, i.e., they act based on local decisions
  - Components are situated in an environment
  - They interact with each other (via the mediation of the environment – stigmergy)

- However, the basic philosophy is somewhat different from that of multiagent systems
  - The accent is more on the ensemble than on the rationality of agents
    - Agents may be irrational or probabilistic
  - There is much more emphasis on the role of the environment
    - Not simply a way to get information
    - But a way to coordinate with each other
    - And the environmental processes counts

- So, given that most modern distributed systems can be assimilated, modeled, as agents, swarm intelligence may have some relevance to them

Why is Swarm Intelligence of Use in Modern Distributed Systems?

- Swarm intelligence systems obtain “useful behaviors” with
  - Very simple components
  - Distributed in an environment
  - Interacting in a local way
  - Nevertheless exhibiting global organization of activities
  - And being capable of adaptation
- When transporting this into modern distributed systems, we would have
  - Simple computational components (thus suitable for implementation in a resource effective way)
  - Interacting with local components (thus avoiding high communication costs)
  - And achieving global application goals in a very effective
  - And adaptable way
- These are indeed valuable properties to enforce in modern distributed systems!
- Still, no free lunch! There is a price to pay in
  - losing certainty of outcome (the final configuration cannot be exaclty predicted)
  - Wasting resources in a mass of agents doing nothing apparently useful

Swarm vs. Individual Intelligence

- Not only "stupid" animals exhibit swarm intelligence
- Sometimes, even mammals or humans do it
  - Castori does water walls on river
  - Buffalos "flock" in the wild
  - Wolves surround a prey
  - Humans forms global self-organized patterns when walking
- This implies that sometimes
  - The power of interactions overcomes the power of individuals
  - Whatever the reason an individual act in a specific way
  - What matter is its interactive behavior, i.e., the way it act and interact in the system
- There is also a role for stochastic variables
  - Since swarm intelligence in "stupid" animals is subject to probabilistic choices by animals
  - The capability for an "intelligent" animals to do rational actions different from those of the group may be perceived as a sort of probabilistic behavior

Part 2

- Examples and Applications of Swarm Intelligence

Examples of Swarm Intelligence

- Let us analyze several examples of swarm intelligent systems
  - And unfold the exploited mechanism

- In particular, we analyze the following phenomena:
  - Firefly synchronization
  - Ant sorting
  - Ant foraging
  - Termites nest building
  - Birds’ and Fishes’ Flocking

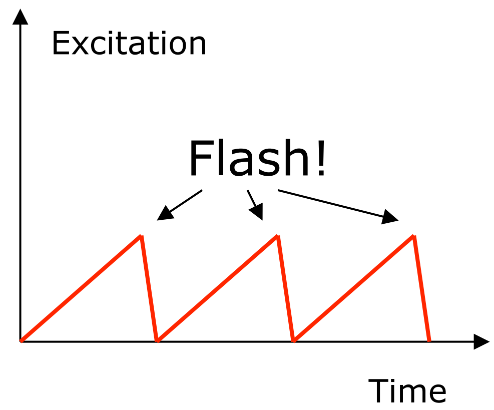
- Most examples are simulated using the “*NetLogo*” simulation system, version 2.1

Firefly Synchronization

- Some species of firefly (e.g., in North America) blink in a synchronized way
  - How can they agree to that
  - Where there is not explicit agreement
  - And there is not explicit leader?
- Related problems in science and nature
  - Synchronized clapping in humans
  - Heart beating
  - Firing neurons in brain
  - Synchronous pendulum

Mechanisms of Firefly Synchronization

- Each firefly flashes with its own frequency
  - It is like a sort of triangular excitation function that, once reaching an *excitation threshold*, make the firefly flash and them get to zero
- But it perceives the local flashing of nearby firefly (or perceive flashing with a distance decreasing intensity)
- Feedback mechanisms
  - If the flashing of other fireflies exceed a given *luminosity threshold* (reinforcement feeedback)
  - And the firefly that perceive it still not already excited enough (control feedback!)
  - It reset its excitation to zero, as if it had just flashed



- See the NetLogo Simulation

Firefly Synchronization: The Algorithm

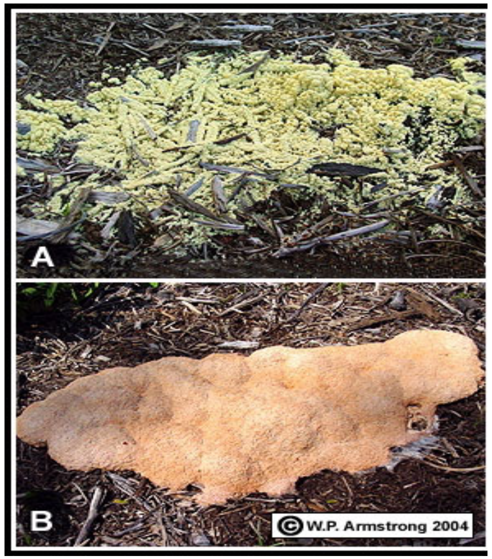
```
Int excitation =0;
Int perceived;
Const excitationthreshold = ET;
Const sensibilitythreshold = ST;
Const noactthreshold = NAT;


DO FORALL (t)
      excitation++;
      perceived = perceive();

      if(excitation<ET && perceived > ST)
            flash();
            excitation=0;
```

Firefly Synchronization: Applications

- To achieve a sort of cost effective global synchronization of activities in a distributed system
  - Where each component (e.g., a peer or a sensor in a network) has its own clock
  - And where clocks, even if a priori synchronized, may have drifts
- Important for modern distributed systems
- In "large" networks", the process may
  - Take a long time to converge
  - Have some mis-synchrony due to delays in signal propagation
- However:
  - The convergence process may be much faster in a small world
  - And the mis-synchrony may be notably contained!

Aggregation of Slime-Mold Cells (Dictyostelium & Company)

- Such cells, sorts of amoebas, spend most of their lives as individuals
  - Wandering around in fluids
  - Or, in the case of their fungi companion, wandering on grasses and trees
- When food is scarce, they aggregate into a sort of unicellular organism
  - With more movement capability
  - With more chances to get food
- How can they agree to get together?



Mechanisms of Aggregation

- In specific environmental conditions
  - Each cell start emitting a sort of chemical pheromone
  - And start “smelling” such pheromones, being attracted in the direction of the greatest scent
- The trivial results is that
  - Clusters start forming
  - The largest a cluster, the more pheromones it emits (positive reinforcing feedback)
- Pheromones diffuse in the environment
  - Giving the chance to far cells to sense the pheromone gradients
- Phoeromones evaporates in time
  - Limiting the dimension of clusters and the spatial extent from which cells can be aggregated

- See the NetLogo simulations

Slime-Mold Aggregation: The Algorithm

```
DO FORALL (t)
      if(food_is_enough)
            move(random_direction);
      else
            emit_pheromone();
            move(direction_of_scent_gradient);
```

Slime-Mold Aggregation: Applications

- Per se, the phenomena has several application

- Coalition formation algorithms
    - Group of agents needed to group together to solve a complex problem that a single agent per se is not able to form
    - The algorithm can be used to recruit agents able to perform the task
- Community formation
  - By spreading different types of pheromones depending on interest, it is possible to dynamically form networks of agents grouped by “interest”, i.e., communities

Ant Sorting

- Ant nests are very organized
  - Ants put in different places eggs, larvae, deaths, foods
  - Still, no ant knows the “map” of the nest
  - No a priori decision has been made on where to put different kinds of items
  - The ants do not directly communicate with each other!
- So, how can such kind of very organized behavior emerge?
- Similar behaviors in bees, and in most social insects

Mechanisms of Ant Sorting

- Here's how ant sort items in their nests
  - Wander randomly around the nest (a sort of Brownian motion, where the direction is instantaneously chosen at random)
  - Sense ("smell") nearby objects
    - The ant has a very short term memory and can remember what it has seen in the past few steps
  - If the ant is not carrying anything, get an object with a certain probability
    - The probability of getting something decreases if it has encountered similar object earlier
  - If the ant is carrying something, drop it with a certain probability
    - The probability increases if the ant has encountered similar objects earlier

- Eventually, all objects of same type will end up being in the same pile

- See the NetLogo simulation

Ant Sorting: The Algorithm

```
Int memory[10]; // remember 10 steps
Boolean carry_object;
Int k1 // probability of getting an object
Int k2 // probability of dropping an object

DO FORALL (t)
        move(choose_random_direction());
        perceived_item = perceive();
        memory[t%10] = perceived_item;
       if (!carry_object)
             // probabilistic action
        // based on f, the number of object of same kind
        // stored in memory
               get = probability(k1/(k1+f));
              // get the object
               if (get) carry_object=true;
       if (carry_object)
              //probabilitic action
               get = probability(f/(k2+f));
               // drop the object
               if (get) carry_object=false;
```

Mechanisms of Ant Sorting

- Ants random motion
  - The fact that ants wander in a sort of Brownian motion will make them explore the whole nest, after or before
  - If there are items around, the ant will get to them
  - Here, the presence of probabilistic behavior is very important!
- Items pile
  - The greater a pile of items (i.e., the more the ant has wandered around it), the less the probability an ant will get something from it (**negative feedback**)
  - The greater a pile of items (i.e., the more the ant has wandered around it), the greater the probability that an ant will drop a similar item there (**positive feedback**)
  - The co-presence of positive and negative feedback is very clear here!

Ant Sorting: Why it Work (2)

- Eventually
  - Large piles will get larger
  - Small piles or individual items will be get by ants and will be moved by a larger pile
  - The key parameter is the “memory” of the ants, which influence the capability of ants of evaluating the size of piles
    - If the has low memory, the sorting will end up in a partial one, with several medium-size piles co-existing
- In general, the result is a set of large piles all with the same types of items
- Adaptability
  - The system works independently of the nest structure, independently of the initial position of items, independently of the actual number of ants
  - The system continue work even if we manually re-arrange things

Ant Sorting: Applications

- The capability of sorting items in a distributed setting may have some useful applications
  - As a way to group files by characteristics in file system search
  - As a way to implement “by subject” searches in Web search engines
  - As a way to distributed and search files in P2P systems
- For instance, with reference to P2P systems
  - We could have sorts of digital “Java” ants that move randomly in the P2P network
  - Get files and transfer them (or simply get references to these files) to other places in the network
  - The result is that all files will be grouped “by category”,

Ant Foraging

- When ants go out of the nest looking for food
  - How can they find the food?
  - How can ants altogether get to the same source of food?
  - How can they all avoid obstacles?
  - How can they all get back to the nest?
- What is found is that, surprisingly
  - Most of the ants find food
  - Most of them will find the shortest path from food to nest and viceversa
  - They avoid obstacles, and find the shortest path even in the presence of obstacles
  - When a new shortest path is created, they will find it
  - When food finishes, the colony start looking for alternate sources

See also the movies (courtesy of Marco Dorigo)

Ant Foraging: How it Works (1)

- Ants go out of the nest and start wander around (Brownian motion)
  - Independently form each other
  - Eventually, they will explore all the space around the nest
  - Turning around obstacles
  - The more ants, the more fast and larger will be such exploration (the mass counts!)
- Eventually, one ant will find some food
  - It will pick up the food
  - And will start trying to get back to the nest, randomly, until the nest is reached and the food dropped out
- Ants who carry food
  - Will deposit a chemical substance (pheromone) that diffuse around
- All ants, independently of whether they carry food or now, tends to be attracted by pheromones
  - In general all ants wander selecting the direction randomly
  - But the scent of pheromones tends to give higher probability to choose a direction towards increasing pheromones concentration (climbing the hill of pheromone gradients)

Ant Foraging: How it Works (2)

- Pheromone spread by ants diffuse
  - This ensure that nearby ants will start sensing pheromones and will be attracted by it
  - Thus, when an ant has found some food, other ants that are looking for food can
    - Follow the pheromone path left by that ants and leading to food
    - And eventually reach food
- How can ants get home?
  - From food to nest, ants will leave a path
  - Thus, once an ant has reached home, there will exists a path from home that other ants will follow to get home
  - And other ants exiting for food will follow to find food
    - But how can this path be made the shortest one?
    - What happens when food finishes or when an obstacle appear?

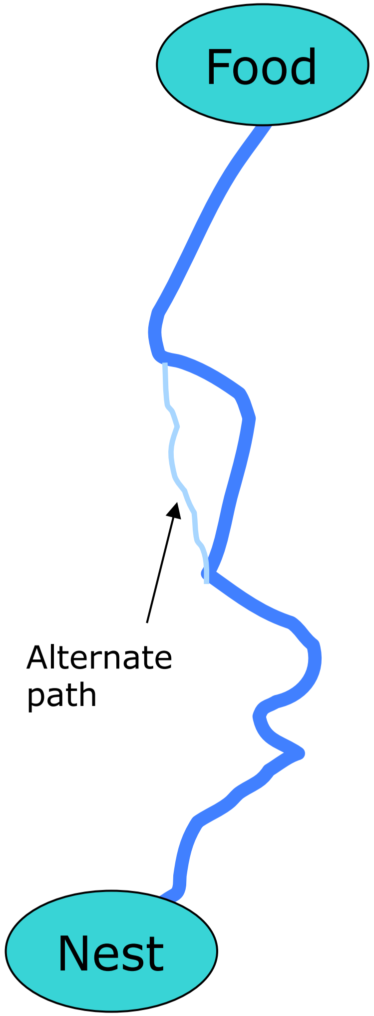
Mechanisms of Ant Foraging

- Pheromone spread by ants evaporate
  - The pheromone path does not last forever
  - It gradually lose its strength as time passes
- This is very important because
  - When food is finished, no more ant will leave pheromone nearby
    - The path evaporates
    - And ants will wander to other directions to find new food sources
  - When an obstacles appear
    - The ants will be forced to circumvent it, creating a new path
    - The old one will evaporate
    - And not ants will any longer be directed towards the obstacles
- Pheromone evaporation is thus very important to promote adaptivity!

- In addition, pheromone evaporation is (together with randomness) what enables finding shortest paths!

Finding Shortest Paths

- When an ant tries to get home with food
  - It wanders randomly
  - And leave a path of pheromones to be followed by other ants
  - Thus, eventually, there is a path form food to home
- However, ants select a path only with high probability, not always
  - Thus, ants with food will create alternate, possibly short-cut path around the main path to be followed by other ants
- Due to evaporation
  - The ants that follow the longer path, will take a longer time, and their track will evaporate sooner
    - Less probability to be followed
  - The ants that follow the shorter track will leave a more “fresh” track of pheromones
    - More probability to be followed
- Eventually, the process converges to a shortest path



Positive and Negative Feedbacks in Ant Foraging

- Mechanisms of positive feedback (re-enforcement) and negative feedback (control) feedbacks are available in ant foraging
- Positive Feedback
  - Pheromones attract more ants
  - Get them to food
  - And make them leave pheromones at their turn
  - This enables the creation of stable path
  - Without such re-enforcement, path would not be stable enough to attract several ants
- Negative feedback
  - Pheromone evaporation ensure that no path will last forever
  - This ensure that useless path will disappear and that the system can adapt to changed situation (forgetting is as important as remembering!)
  - Also, together with randomness, ensure that all possible solutions, e.g., the whole search space, are explored in any case
  - E.g., that the shortest path is found

An Alternate Approach: Double Pheromone Types

- An alternative (not corresponding to real-world ants, but interesting) is that ants can emit two types of pheromones
  - Nest pheromones, when they have left the nest and look for food
  - Food pheromones, when they have found food
- This possibility ensure that
- Once an ant has found some food
  - It can start smelling the nest pheromones
  - It is expected that there will be a higher concentration of nest pheromones in the direction to home
  - So that the ant can more easily find a path back home

- This is more effective and it is the one which is usually implemented in systems mimicking ant foraging

Ant Foraging: The Algorithm

```
Boolean carry_food;
Int food_pheromone_gradient, nest_pheromone_gradient;

DO FORALL (t)
  if(!carry_food)
        emit_nest_pheromone();
       food_pheromone_gradient = sense_food();
        if(food_pheromone_gradient)
               move(choose_preferential_direction());
       else
               move(choose_random_direction());
  if(carry_food)
       emit_food_pheromone();
       nest_pheromone_gradient = sense_nest();
        if(nest_pheromone_gradient)
               move(choose_preferential_direction());
       else
               move(choose_random_direction());
  if(at_nest) carry_food=false;
  if(at_food) carry_food=false;
```

Ant Foraging Applications: Ant Optimization

- Optimization in Networks (Dorigo et al., 1997)
  - TSP problem: have digital ants “live” on the graph, and have them explore the graph looking for paths
  - By properly tuning the parameters, the ants will find (if not the optimal) very nearly optimal paths
  - With performances that can be better to those of traditional approaches
- The same approach can be mapped into other classical optimization problems

		Best	Average	Std.dev.
Ant simulation	AS	420	420.4	1.3
Tabu search	TS	420	420.6	1.5
Simulated annealing	SA	422	459.8	25.1

Ant Foraging Applications: Routing in Networks

- Routing in networks (White and Pagurek, 1997)
  - Other than the capability of finding shortest path, one could exploit the capability of dynamic adaptation to discover routing paths in networks
- Map components of ant foraging in components of a data network
  - Environment = network
  - Nest pheromones gradients = Routing tables
  - Ants with food (backward ants) = data packets
  - Ants without food (forward ants) = active controllers of the routing system, sorts of simple mobile agents, spread by nodes as normal data packets but only in charge of exploring the space and of creating pheromones trails towards home
  - Food source = sender of data packet
  - Nest = receiver of data packet
- Of course, parameters needs to be carefully evaluated via simulations
  - Tuning of pheromone evaporation and diffusion rate (they must be stable enough but must propagate quite quickly)
  - Tuning of number of explorer ants
- But it works!
  - Di Caro and Dorigo shows it can outperform Internet in the presence of congestions
  - And it is especially suited for mobile ad-hoc networks

Ant Foraging Applications: Pervasive Computing

- To implement swarm intelligence systems, there is in general need of something acting as environment
  - The Internet, a P2P network, a sensor network
- How can we use swarm intelligence for everyday environment to enforce pervasive computing?
- The Idea (Mamei and Zambonelli, 2005) is to deploy pheromones on RF-ID tags
  - RF-ID tags could be already in the environment
  - Or they could be spread by explorer robots
  - Mobile devices (e.g., PDA, laptops, smart phones) could "sense" the pheromones by reading RF-ID tags or could leave pheromones by "writing" them
- This could have a lot of applications
  - Finding objects and people
  - Orchestrating team movements
  - Act as sort of "environmental" memory
  - Or simply used as a mean to promote swarm coordination in swarms of mobile robots

Ant Foraging Applications: P2P Computing in Anthill

- Ant foraging – and the creation of pheromones trials, can be effective alternative to find information in P2P networks
- Anthill system (Montresor and Babaoglu, 2002)
  - An ant-based P2P system for service oriented P2P computing
- When a user needs a service (possibly composed of several sub-services) or some information in the P2P network
  - It launches a number of ants in the network looking for the services
  - Once a service is found, ants leave pheromone paths to it
- So, whenever other people has requested similar services, the pheromone trial may already exists
  - More requested services have re-enforced trials
  - Adaptivity in the case of service unavailability or faults

Ant Foraging Applications: Other Applications

- Dynamic manufacturing scheduling (Holvoet, 2004)
- Pattern detection in artificial vision (Parunak, 2000)
- Motion coordination for unmanned vehicles (Parunak, 2003)
- Extraction of information from large data warehouses

The Concept of Stigmergy

- Stigmergy = signs + actions (Grassé, 1957)
  - In any foraging, as well as in slime-mold aggregation and in ant sorting
  - Interactions between agents occur indirectly via the environment
- This is stigmergy in that
  - Agent have, via their actions, change the status the environment
  - This, in turn, affect the behavior of other agents
- In particular, in the examples we have seen
  - Pheromones in ant sorting and in slime-mold aggregation
  - Perception of environmental situation in ant sorting
  - Indirect perception (via the environment) of other fireflies behaviors in firefly synchronization

- This is something similar to what happens in shared dataspaces and tuple-based coordination in distributed systems
  - Stigmergy decouples interactions
  - Stigmergy, by definition, is context-awareness
  - Both of which are fundamental to enforce adaptivity

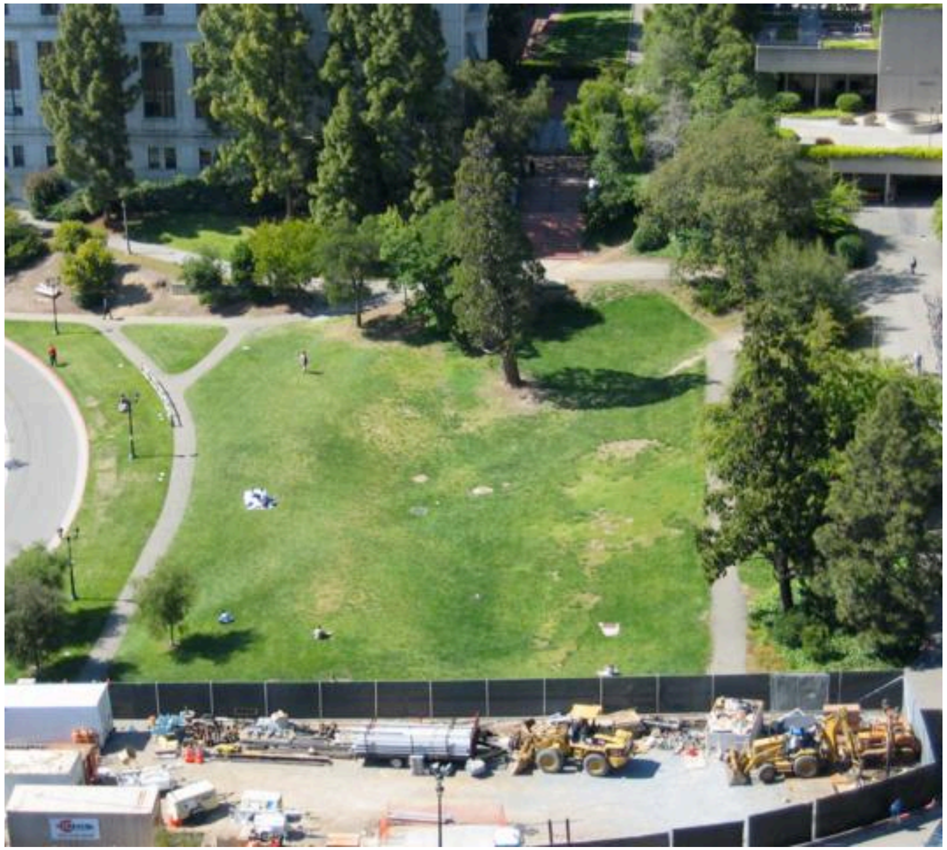
The Role of the Environment

- The role of the environment is not merely passive
- There are processes in the environment which are fundamental to support self-organization and adaptation
  - E.g., pheromone diffusion and evaporation
  - E.g. spreading of information (e.g., visual information in firefly synchronization)
- In other words, self-organization needs to be actively supported by the environment

- This somewhat resembles the concept of "programmable tuples space" (Omicini, 1998)
  - Which encode the laws of coordination
  - And provides for true decoupled and adaptive coordination

Swarming in Humans: Emergent Footpaths

- Imagine a park with stone footpaths
  - That should discourage you to walk on grass ;-)
  - But are traced with absurd geometries
- You will definitely start "cutting the corners"
  - This will make the grass signed
  - Inviting other people to walk in
  - Until a clear spontaneous footpath emerge, and it will be the shortest path
- If a path go to disuse
  - Grass will grow again (as in pheromone evaporation)

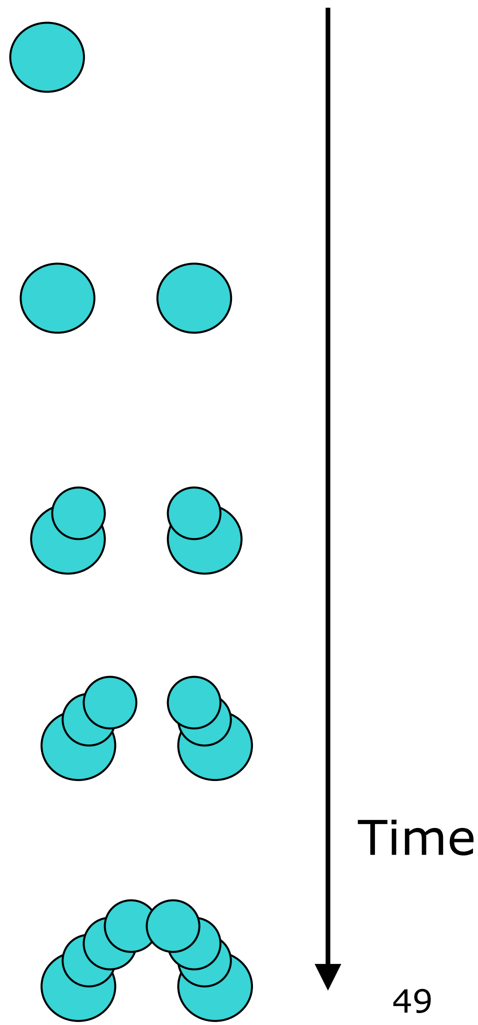


Another Interesting Example: Nest Building in Termites

- Not many actual applications in distributed systems (maybe in self-assembly?)
- But a very interesting example of swarm architecture
- A complex artifact built by "stupid" entities

Nest Building in Termites: How it Works

- Termite start getting and carrying pieces of mud
- And drop it marking it with pheromones
  - Either randomly in some place
  - Or where it sense a higher concentration of pheromones
  - i.e., above other pieces of mud, thus creating growing piles of mud
- When there are close piles of mud, termites are more attracted between the two
  - And deposit the mud on the pile, with a tendence of the pile to approach each other
  - Until an arch form
- And so on, recursively, the nest grows a floor over the other…



Other Examples

- A lot of other examples may be provided
  - Flocking in birds and fish, and other social animals
  - Wolves surrounding a prey and ants collectively carrying big objects
  - Morphogenesis and self-assembly (other than the simple model of slime-molds)
- With relevant applications in pervasive computing and robotics
- This can be better presented when talking about field-based coordination
  - A specific type of stigmergic interaction

Before Reaching the End: Swarms and the Mind

- So, from what said, we have clear in minds that swarms are not intelligent
  - They only seems intelligent
  - There is not a “collective mind” in ant colony…
- So, when we consider a system of simple electrical perception-reaction electrical components
  - Connected in some sorts of directed lattice (or small world network)
  - And exhibiting patterns of synchronization, of coordinated activity
  - We would never say that there is a collective mind there…but…
- Isn’t the above exactly a brain simply evolved by nature?
  - So, what is “intelligence” after all?
  - What is “mind”
  - Is mind only a specific form of “swarm intelligence”
  - So, if we have a “mind”, and we know we have, why couldn’t ant colonies have one?
- Sorry, this is getting philosophy and not computer science…

Conclusions and Open Issues

- Many natural systems exhibit seemingly intelligent behaviors
  - Global adaptive self-organized behavior
  - Achieving some global goals
  - Despite very simple components
- The lessons of these natural systems can be effectively transposed to distributed systems
  - To obtain a variety of applications
  - Robust, adaptive, and resource effective
- Other mechanisms exploited by natural systems, not analyzed in detail here, may be effectively exploited as the basis to enforce swarm intelligence
  - E.g., fields, chemical gradients, etc.
  - Will see some of these in other lectures
- Still, the problem of engineering a swarm system is open
  - General methodology to design swarm intelligent systems
  - Other than that of reverse engineering existing ones
  - Evolutionary approaches to evolve “intelligent” populations
  - These will be the subject of another lesson…