We are IntechOpen,
the world's leading publisher of Open Access books
Built by scientists, for scientists

4,400
Open access books available

117,000
International authors and editors

130M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS

BOOK CITATION INDEX

INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Meta-heuristic Techniques and Swarm Intelligence in Mobile Ad Hoc Networks

Floriano De Rango and Annalisa Socievole
DEIS Department, University of Calabria
Rende (Cs),
Italy

1. Introduction

The infrastructure-less and the dynamic nature of mobile ad hoc networks (MANETs) demands new set of networking strategies to be implemented in order to provide efficient end-to-end communication. MANETs employ the traditional TCP/IP structure to provide end-to-end communication between nodes. However, due to their mobility and the limited resource in wireless networks, each layer in the TCP/IP model requires redefinition or modifications to work efficiently in MANETs. One interesting research area in MANETs is routing. Routing is a challenging task and has received huge attention from researches. Due to the adaptive and dynamic nature of these networks, the *Swarm Intelligence* approach is considered a successful design paradigm to solve the routing problem. Swarm intelligence is a relatively new approach to problem solving that takes inspiration from the social behaviours of insects and of other animals. In particular, the collective behaviour of ants have inspired a number of methods and techniques among which the most studied and the most successful is the general purpose optimization technique known as *Ant Colony Optimization* (ACO) meta-heuristic. ACO takes inspiration from the foraging behaviour of some ant species. These ants deposit a chemical substance called *pheromone* on the ground in order to mark some favourable path that should be followed by other members of the colony. This behaviour has led to development of many different ant based routing protocols for MANETs. In this chapter, a description of swarm intelligence approach and ACO meta-heuristic is given, an overview of a wide range of ant based routing protocols in the literature is proposed and finally other applications related to ACO in MANETs and new directions are discussed.

2. The swarm intelligence approach

Swarm Intelligence (Bonabeau et. al, 1999) is a property of natural and artificial systems involving multiple individuals interacting with each other and the environment to solve complex problems exhibiting a collective intelligent behaviour. Examples of systems studied by swarm intelligence are colonies of ants and termites, schools of fish, flocks of birds, herds of land animals. Some human artifacts also fall into the domain of swarm intelligence, notably some multi-robot systems, and also certain computer programs written to solve optimization and data analysis problems.

Swarm intelligence has a multidisciplinary character. It is usual to divide swarm intelligence research into two areas according to the nature of the systems under analysis: in *natural* swarm intelligence research biological systems are studied while in *artificial* swarm intelligence human artifacts are studied. A different classification of swarm intelligence research can be given based on the goals that are pursued: it is possible to identify a *scientific* and an *engineering* stream. The goal of the scientific stream is to model swarm intelligence systems in order to understand the mechanisms allowing a system to behave in a coordinated way as a result of local individual-individual and individual-environment interactions. On the other hand, the goal of the engineering stream is to employ the biological behaviours in order to design systems able to solve problems of practical relevance.

The typical swarm intelligence system has the following properties:

- it is composed of many individuals;
- the individuals are either all identical or belong to a few typologies;
- the interactions among the individuals are based on simple behavioural rules that make use of local information exchanged directly or via the environment;
- the overall behaviour of the system results from the interactions of individuals with each other and with their environment.

The characterizing property of a swarm intelligence system (Tarasewich & MecMullen, 2002) is its capability to act in a coordinated way without the presence of a coordinator. In nature there are many examples of swarms performing some collective behaviour without any individual controlling the group. Wasps build nests with a highly complex internal structure that is well beyond the cognitive capabilities of a single wasp. Termites build nests whose dimensions can reach many meters of diameter and height. When compared to a single termite, which can measure as little as a few millimetres, these nests are huge. Schools of fish and flocks of birds are other examples of highly coordinated groups. Scientists have shown that these elegant behaviours can be understood as the result of a self-organized process where there is no leader and each individual bases its movement decisions solely on locally available information: the distance, the perceived speed, and the direction of movement of neighbours.

The most interesting swarm-level behaviours belongs to ants. What is fascinating is that ants are able to discover the shortest path to a food source and to share that information with another ants through *stigmergy* (Deneubourg et al., 1990; Dorigo et al., 1999). Stigmergy is a form of indirect communication used by ants in nature to coordinate their problem-solving activities. Ants realize stigmergetic communication by depositing on the ground a chemical substance called *pheromone* that induces changes in the environment which can be sensed by other ants. From the observation of real ant colonies, ant algorithms were inspired and applied to many different optimization problems.

The main advantages of the swarm intelligence approach compared with a classical approach are the following:

- flexibility: the group can quickly adapt to a changing environment;
- robustness: even when one ore more individuals fails, the group can still perform its tasks;
- self organisation: the group needs relatively little supervision or top down control.

These properties make swarm intelligence a successful design paradigm.

2.1 Ant foraging behaviour

The observation of ant's behaviour inspired the implementation of different optimization algorithms (Bonabeau et al., 2000). An ant colony is able to find the shortest path between

the nest and a food source using simple local decisions. Ants use a signalling communication system based on the deposition of pheromone over the path it follows, marking a trail. Pheromone is a hormone produced by ants that establishes a sort of indirect communication among them.

An ant foraging for food lay down pheromone over its route. When this ant finds a food source, it returns to the nest reinforcing its trail. Other ants in the proximities are attracted by this substance and have greater probability to start following this trail and thereby laying more pheromone on it. This process works as a positive feedback loop system because the higher the intensity of the pheromone over a trail, the higher the probability of an ant start travelling through it. The following example (see Fig. 1) will show how this process leads the colony to optimize a route:
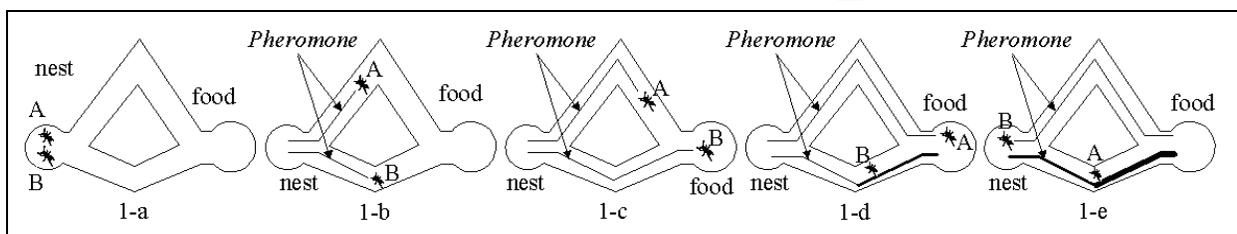

Fig. 1. Two ants exploring the shortest path

Suppose two ants, called A and B, were randomly searching for food when they found two different routes between the nest and the source. Since the route chosen by ant B is shorter, first ant B will reach food. Going back to the nest, ant B will choose the same path laying more pheromone over it. When ant A will also find the food, it will choose the path with the higher pheromone concentration to reach the nest. So, ant A will follow the same B's path to the nest. As the process continues, the pheromone concentration on this trail will increase while the longest route will be discarded because of the pheromone evaporation process.

When more paths are available from the nest to a food source, a colony of ants may be able to exploit the pheromone trails left by the individual ants to discover the shortest path from the nest to the food source and back.

2.2 ACO meta-heuristic

The ant colony foraging behaviour has attracted a lot of attention in combinatorial optimization problems, and has been reverse-engineered in the context of *Ant Colony Optimization* (ACO) meta-heuristic (Deneubourg et al., 1990). A *meta-heuristics* is a set of algorithmic concepts that can be used to define heuristic methods applicable to a wide set of different problems. In other words, a meta-heuristic is a general purpose algorithmic framework that can be applied to different optimization problems with relatively few modifications. Examples of meta-heuristics include simulated annealing (Cern'y, 1985), tabu search (Glover & Laguna, 1997), iterated local search (Lourenço et al., 2002), evolutionary computation (Dorigo et al. 2006), and ant colony optimization (Dorigo et al. 1996; Dorigo et al., 1999; Dorigo & Stützle, 2004).

In ACO, a number of artificial ants build solutions to an optimization problem and exchange information on the quality of these solutions via a communication scheme that is reminiscent of the one adopted by real ants.

The computational resources are allocated to a set of relatively simple agents (artificial ants) that communicate indirectly by stigmergy. Artificial ants have been enriched with some

capabilities which do not find a natural correspondence in order to make them more effective and efficient. In particular, the use of a colony of cooperating individuals, an (artificial) pheromone trail for local stigmergetic communication, a sequence of local moves to find shortest paths, and a stochastic decision policy using local information and are stemmed from real ants. The other features which do not find their counterpart in real ants are the following:

- artificial ants live in a *discrete world* and their moves consist of transitions between discrete states;
- artificial ants have an *internal state* containing the memory of the ant past actions;
- artificial ants deposit an amount of pheromone which is a function of the *quality of the solution* found;
- artificial ants timing in pheromone laying is problem dependent and often does not reflect real ants behaviour;
- to improve overall system efficiency, ACO algorithms can be enriched with *extra capabilities* like lookahead, local optimization, backtracking, and so on, that cannot be found in real ants.

In ACO algorithms a finite size colony of artificial ants with the above described characteristics collectively searches for good quality solutions to the optimization problem under consideration. The complexity of each ant is such that even a single ant is able to find a (probably poor quality) solution. High quality solutions are only found as the emergent result of the global cooperation among all the agents of the colony concurrently building different solutions.

The model of a combinatorial optimization problem is used to define the pheromone model of ACO. A pheromone value is associated with each possible *solution component* and the set of all possible solution components is denoted by *C*. In ACO, an artificial ant builds a solution by traversing a fully connected *construction graph $G_c(V,E)$*, where *V* is a set of vertices and *E* is a set of edges. This graph can be obtained from the set of solution components *C* in two ways: components may be represented either by vertices or by edges. Artificial ants move from vertex to vertex along the edges of the graph, incrementally building a *partial solution.* Additionally, ants deposit a certain amount of pheromone on the components; that is, either on the vertices or on the edges that they traverse. The amount Δ_τ of pheromone deposited may depend on the quality of the solution found. Subsequent ants use the pheromone information as a guide toward promising regions of the search space. The ACO meta-heuristic algorithms is the following:

Set parameters, initialize pheromone trails
SCHEDULE_ACTIVITIES
  ConstructAntSolutions
  ApplyLocalSearch {optional}
  UpdatePheromones
END_SCHEDULE_ACTIVITIES

After initialization, the meta-heuristic iterates over three phases: at each iteration, a number of solutions are constructed by the ants; these solutions are then improved through a local search (this step is optional), and finally the pheromone is updated.

The interest of the scientific community in ACO meta-heuristic has risen sharply. Different ACO algorithms have been proposed in the literature (Dorigo et al. 1996; Dorigo et al., 1999; Dorigo & Stützle, 2004). Although ACO has been applied in many combinatorial

optimization problems this chapter focuses on surveying ACO approaches in networks routing and load-balancing. In the following sections the most relevant ACO algorithms for routing and load balancing problems will be analyzed.

2.3 Approaches to mitigate stagnation

A major weakness of ACO algorithms is the *stagnation* in which all ants are taking the same position. Stagnation occurs when a network reaches its convergence (or equilibrium state) (Sim & Sun, 2003); an optimal path p_0 is chosen by all ants and this recursively increases an ant's preference for p_0. This may lead to the congestion of p_0 and to a dramatic reduction of the probability of selecting other paths. These two consequences are undesirable for a dynamic network since p_0, becoming congested, may become nonoptimal and disconnected due to network failure. Moreover, other nonoptimal paths may become optimal due to changes in network topology, and new or better paths may be discovered.

To alleviate the stagnation problem of ACO algorithms, different approaches have been proposed (Dorigo & Stützle, 2004) and can be categorized as follows:
- pheromone control;
- pheromone-heuristic control;
- privileged pheromone laying.

Pheromone control adopts several approaches to reduce the influences from past experience and encourages the exploration of new paths or paths that were previously nonoptimal: *evaporation*, *aging*, *limiting* and *smoothing pheromone*.

The approach called evaporation is typically used in conjunction with ACO in order to reduce the effect of past experience. Evaporation prevents pheromone concentration in optimal paths from being excessively high and preventing ants from exploring other (new or better) alternatives. In each iteration, the pheromone values $\tau_{i,j}$ in all edges (i,j) are discounted by an *evaporation factor* called *p.*

Additionally, past experience can also be reduced by controlling the amount of pheromone deposited for each ant according to its age. This approach is known as *aging*. In aging, an ant deposits lesser and lesser pheromone as it moves from a node to another one. Aging is based on the rationale that "old" ants are less successful in locating optimal paths since they may have taken longer time to reach their destinations. Both aging and evaporation prefer recent encouraging discoveries of new paths that were previously nonoptimal.

Limiting pheromone mitigate stagnation by limiting the amount of pheromone in every path. By placing an upper bound τ_{max} on the amount of pheromone for every edge (i,j), the preference for optimal paths over nonoptimal paths is reduced. A variant of such an approach is *pheromone smoothing*, in which the amount of pheromone along an edge is reinforced as follows:

$$\tau_{i,j}(t') = \tau_{i,j}(t) + \delta \cdot (\tau_{max} - \tau_{i,j}(t)) \qquad (1)$$

where δ is a constant between 0 and 1. It can be noticed that as $\tau_{i,j} \rightarrow \tau_{max}$, a smaller amount of pheromone is reinforced along an edge (i,j) .While evaporation adopts a uniform discount rate for every path, pheromone smoothing places a relatively greater reduction in the reinforcement of pheromone concentration on the optimal path(s). Consequently, pheromone smoothing seems to be more effective in preventing the generation of dominant paths.

Pheromone-heuristic control configures ants so that they do not solely rely on sensing pheromone for their routing preferences. This can be accomplished by configuring the probability function $P_{i,j}$ for an ant to choose an edge *(i,j)* using a combination of both pheromone concentration $\tau_{i,j}$ and heuristic function $\eta_{i,j}$. $\eta_{i,j}$ is function of the cost of edge which may include factors such as queue length, distance, and delay . $P_{i,j}$ at time t is given as follows:

$$P_{i,j}(t) = \frac{[\tau_{i,j}(t)]^a \cdot [\eta_{i,j}]^\beta}{\sum [\tau_{i,j}(t)]^a \cdot [\eta_{i,j}]^\beta} \qquad (2)$$

where *a* and *β* represent the respective adjustable weights of $\tau_{i,j}$ and $\eta_{i,j}$. The routing preferences of ants can be altered by selecting different values of *a* and *β*. If *a > β*, ants choose paths with more optimistic heuristic values.

By adopting the policy of *privileged pheromone laying*, a selected subset of ants to have the privilege to deposit extra or more pheromone on the best paths (in terms of trip time and length). This approach reduces the probability of ants reinforcing stagnant paths that are nonoptimal or congested.

3. ACO routing algorithms

ACO routing algorithms (Dorigo et al., 1999) are a subset ACO algorithms which model the behaviour of insect swarms to solve the routing problem.

ACO routing algorithms show a number of interesting properties compared to traditional routing algorithms. Firs of all, they are adaptive by means of continuous path sampling and probabilistic ant forwarding which leads an interrupted exploration of the routing capabilities. Moreover, they are robust because routing information is the result of the repeated sampling of paths. The use of sampling implies that routing information is based on direct measurements of the real network situation, which enhances its reliability.

In the following subsections, the main ACO algorithms solving the routing problem will be discussed. In order to illustrate the differences between them clearly, the example of the *travelling salesman problem* will be analyzed.

In the TSP (Dorigo & Gambardella, 1997) a set of locations (e.g. cities) and the distances between them are given. The problem consists of searching a closed tour of minimal length that visits each city once and only once. To apply ACO to the TSP, the graph is defined by associating the set of cities with the set of vertices of the construction graph. Since in the TSP it is possible to move from any given city to any other city, the construction graph is fully connected and the number of vertices is equal to the number of cities. The lengths of the edges between the vertices are proportional to the distances between the cities represented by these vertices and pheromone values and heuristic values are associated with the edges of the graph. Pheromone values are modified at runtime and represent the cumulated experience of the ant colony, while heuristic values are problem dependent values that, in the case of the TSP, are set to be the inverse of the lengths of the edges. The ants construct the solutions as follows. Each ant starts from a randomly selected city (vertex of the construction graph) and at each construction step it moves along the edges of the graph, keeping a memory of its path. In subsequent steps ant chooses among the edges that do not lead to vertices that it has already visited. A solution will be constructed once an ant has visited all the vertices of the graph. At each construction step, an ant probabilistically

chooses the edge to follow among those that lead to yet unvisited vertices. The probabilistic rule is biased by pheromone values and heuristic information: the higher the pheromone and the heuristic value associated to an edge, the higher the probability an ant will choose that particular edge. Once all the ants have completed their tour, the pheromone on the edges is updated. Each of the pheromone values is initially decreased by a certain percentage. Each edge then receives an amount of additional pheromone proportional to the quality of the solutions to which it belongs (there is one solution per ant). This procedure is repeatedly applied until a termination criterion is satisfied.

3.1 AS: Ant System
Ant system (AS) (Fenet & Hassas, 1998) was the first ACO algorithm to be proposed in the literature. The pheromone values are updated by *all* the ants that have completed the tour. Solution components, denoted with $c_{i,j}$, are the edges of the graph, and the pheromone update for $\tau_{i,j}$, that is, for the pheromone associated to the edge joining cities i and j, is performed as follows:

$$\tau_{i,j} \leftarrow (1-\rho) \cdot \tau_{i,j} + \sum_{k=1}^{m} \Delta\tau_{i,j}^k \qquad (3)$$

Where $\rho \in (0,1]$ is the evaporation rate, m is the number of ants, and $\Delta\tau_{i,j}^k$ is the quantity of pheromone laid on edge (i,j) by the k-th ant:

$$\Delta\tau_{i,j}^k = \begin{cases} \dfrac{1}{L_k} & \text{if } k-th \text{ ant travels on edge } (i,j) \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

where L_k is the tour length of the k-th ant.
In order to construct the solutions, the ants traverse the construction graph and make a probabilistic decision at each vertex. The transition probability of the k-th ant moving from city i to city j is given by:

$$P(c_{i,j} \mid s_k^p) = \begin{cases} \dfrac{\tau_{i,j}^a \cdot \eta_{i,j}^\beta}{\sum_{c_{i,j} \in N(s_k^p)} \tau_{i,j}^a \cdot \eta_{i,j}^\beta} & \text{if } j \in N(s_k^p) \\ 0 & \text{otherwise} \end{cases} \qquad (5)$$

where $N(s_k^p)$ is the set of components that do not belong yet to the partial solution s_k^p of ant k, and parameters a and β control the relative importance of the pheromone versus the heuristic information $\eta_{i,j}=1/d_{i,j}$, where $d_{i,j}$ is the length of component $c_{i,j}$.

3.2 Ant Colony System
The Ant Colony System algorithm (Dorigo & Gambardella, 1997) was proposed as an improvement over the original AS algorithm. The first relevant difference between ACS and AS is the decision rule used by the ants during the construction process. Ants in ACS use the so-called *pseudorandom proportional* rule: the probability for an ant to move from city i to city j depends on a random variable q uniformly distributed over [0,1], and a parameter q_0; if $q \leq$

q_0, then, among the feasible components, the component $\tau_{i,j}\eta_{i,j}^{\beta}$ that maximizes the product is chosen; otherwise, the same equation as in AS is used. This rather greedy rule, which favours exploitation of the pheromone information, is counterbalanced by the introduction of a diversifying component: the *local pheromone update* (Ducatelle et al., 2005). The local pheromone update is performed by all ants after each construction step. Each ant applies it only to the last edge traversed:

$$\tau_{i,j} = (1-\varphi)\cdot\tau_{i,j} + \varphi\cdot\tau_0 \tag{6}$$

where $\varphi \in (0,1]$ is the pheromone decay coefficient, and τ_0 is the initial value of the pheromone. The interesting goal of the local update is to diversify the search performed by subsequent ants during one iteration. In fact, decreasing the pheromone concentration on the edges as they are traversed during one iteration encourages subsequent ants to choose other edges and hence to produce different solutions. This also prevents that several ants produce identical solutions during one iteration. Additionally, because of the local pheromone update in ACS, the minimum values of the pheromone are limited.

As in AS, also in ACS at the end of the construction process a pheromone an *offline* pheromone update is performed. This update is performed only by the best ant and only edges visited by the best ant are updated, according to the equation:

$$\tau_{i,j} \leftarrow (1-\rho)\cdot\tau_{i,j} + \varphi\cdot\Delta\tau_{i,j}^{best} \tag{7}$$

where $\Delta\tau_{i,j}^{best} = 1/L_{best}$ if the best ant used edge *(i,j)* in its tour, $\Delta\tau_{i,j}^{best} = 0$ otherwise. L_{best} can be set to either the length of the best tour found in the current iteration (L_{ib}) or the best solution found since the start of the algorithm (L_{bs}).

3.3 MMAS: MAX-MIN Ant System

MAX-MIN ant system (MMAS) algorithm (Stützle & Hoos, 1998) is another improvement of the original AS algorithm. Unlike AS, only the best ant adds pheromone trails, and the minimum and maximum values of the pheromone are explicitly limited (in AS and ACS these values are limited implicitly as a result of the algorithm working rather than a value set explicitly by the algorithm designer).

The pheromone update equation (applied, as in AS, to all the edges) is the following:

$$\tau_{i,j} \leftarrow (1-\rho)\cdot\tau_{i,j} + \Delta\tau_{i,j}^{best} \tag{8}$$

where $\Delta\tau_{i,j}^{best} = 1/L_{best}$ if the best ant used edge *(i,j)* in its tour, $\Delta\tau_{i,j}^{best} = 0$ otherwise. As in ACS, L_{best} can be set (subject to the algorithm designer decision) to either the length of the best tour found in the current iteration (L_{ib}) or the best solution found since the start of the algorithm (L_{bs}), or to a combination of both.

The pheromone values are constrained between a max value τ_{max} and a minimum value τ_{min} by verifying, after they have been updated by the ants, that all pheromone values are within the imposed limits: $\tau_{i,j}$ is set to τ_{max} if $\tau_{i,j} > \tau_{max}$ and to τ_{min} if $\tau_{i,j} < \tau_{max}$. The minimum value $\tau_{i,j} < \tau_{min}$ is most often experimentally chosen (however, a theory about how to define its value analytically has been developed). The maximum value τ_{max} may be calculated analytically using the optimum ant tour length value. For the TSP, $\tau_{max} = 1/(\varphi\cdot L^*)$, where L^* is the

length of the optimal tour. If L^* is not known, it can be approximated by L_{bs}. It is important to underline that the value of the trails is set to τ_{max}, and that the algorithm is restarted when no improvement can be observed for a given number of iterations (Stützle, 1999).

3. ACO routing algorithms for MANETs

A mobile ad-hoc network (MANET) is a set of mobile nodes which communicate over radio. These networks have an important advantage, they do not require any existing infrastructure or central administration. Therefore, mobile ad-hoc networks are suitable for temporary communication links.

Due to the limited transmission range of wireless interfaces, usually communication has to be relayed via intermediate nodes. Thus, in mobile multi-hop ad-hoc networks each node also has to be a router. To find a route between different endpoints is a major problem in mobile multi-hop ad-hoc networks. Many different approaches to handle this problem were proposed in literature (Buruhanudeen et al., 2007), but so far no routing algorithm has been suitable for all situations.

Analyzing some important features of mobile ad-hoc networks, the following considerations explain why ant algorithms could perform well in these networks:

- Dynamic topology: this property is responsible for the unfulfilling performances of many classical routing algorithms in mobile ad-hoc networks. The ant algorithms are based on autonomous agent systems imitating individual ants. This allows a high adaptation to the current topology of the network.
- Local information: in contrast to other routing approaches, the ant algorithms make use of local information; no routing tables or other similar information have to be transmitted to other nodes of the network.
- Link quality: it is possible to integrate the connection/link quality into the computation of the pheromone concentration, especially into the evaporation process. This will improve the decision process with respect to the link quality.
- Support for multi-path: each node has a routing table with entries for all its neighbours. Adding the information about the pheromone concentration, the decision rule for selection of the next node could be based on the pheromone concentration at the current node.

In this section, an overview of the main ant based routing algorithms proposed explicitly for MANETs will be presented.

Ad hoc Networking with Swarm Intelligence (ANSI). ANSI is a reactive routing protocol (Rajagopalan & Shen, 2005) which defines two kinds of mobile agents called *forward reactive ants* and *backward reactive ants*. The routing tables in ANSI contain an entry for each reachable node and next best hop while the ant decision tables store the pheromone values. In ANSI, the forward reactive ants are generated only when a node has to transmit data to another node. The forward reactive ants are broadcast while the backward reactive ants retrace the path of forward reactive ants and update the pheromone values at the nodes. The data packets choose the next hop deterministically i.e., the hop which contains the largest pheromone value is chosen as the next hop.

Ant-colony-based Routing Algorithm (ARA). ARA is another reactive routing protocol (Günes & Spaniel, 2003) for MANETs. The routing table entries in ARA contain pheromone values for the choice of a neighbour as the next hop for each destination. The pheromone

values in the routing tables decay with time and the nodes enter in a sleep mode if the pheromone in the routing table has reached a lower threshold. As in ANSI, route discovery in ARA is performed by two kind of mobile agents: forward ants and backward ants. During route discovery, the forward and backward ant packets characterized by unique sequence numbers to prevent duplicate packets, are flooded through the network by the source and destination nodes, respectively. The forward and backward ants update the pheromone tables at the nodes along the path for the source and destination nodes respectively. At the end of the route discovery process for a particular destination, the source node does not generate new mobile agents for the destination instead the route maintenance is performed by the data packets.

Probabilistic Emergent Routing Algorithm (PERA). Also in PERA (Baras & Mehta, 2003) route discovery is performed by forward and backward ants. These ant agents create and adjust probability distribution at each node for the node's neighbours. The probability related to a neighbour reflects the relative likelihood of that neighbour forwarding and eventually delivering the packet. Each forward node contains the IP address of its source node, the IP address of the destination node, a sequence number, a hop count field and a dynamically growing stack The stack contains the information about the nodes traversed by the forward ant and the times at which the nodes have been traversed. When a node does not have a record of a route to a destination, it creates a forward ant and the node pushes its own IP address on to the stack of the forward ant as well as the time at which the ant is created. Henceforth, the node keeps sending forward ants periodically to the destination for as long as a route is required. When a forward node reaches the destination, the destination node creates a new backward ant. It uses the information contained in the forward ant on the reverse path to modify the probability distribution at each node and update routing tables to reflect the current status of the network. Since the forward ant is broadcast at the source and intermediate nodes, each forward ant will cause the broadcast of multiple forward ants, several of which may find different paths to the destination, generating multiple backward ants.

POSition based ANT colony routing algorithm (POSANT). POSANT is a reactive routing algorithm (Kamali & Opatrny, 2008) based on ant colony optimization and location of nodes. This protocol is able to find optimum or nearly optimum routes when a given network contains nodes with different transmission ranges. Each node is assumed to be aware of its position, the position of its neighbours and the position of the destination node. A route in POSANT is searched only when there is a collection of data packets that are to sent from a source node to a destination node. Sending the data packets will start after a route from source to destination is established. Before that, only forward and backward ants are being exchanged. In order to minimize the time that POSANT spends to find a route while keeping the number of generated ants as small as possible, information about the position of nodes is used as a heuristic value. Neighbours in POSANT are partitioned into three zones in dependence of the position. The use of location information as a heuristic parameter results in a significant decrease of the time required to establish routes from a source to a destination. Moreover, having a short route establishment time, POSANT reduces greatly the number of control messages. POSANT has also a higher delivery rate with a shorter average packet delay than other position based routing algorithms.

Ant Routing Algorithm for Mobile Ad hoc networks (ARAMA). ARAMA (Hossein & Saadawi, 2003) is a proactive routing algorithm. As in other ACO algorithms for MANETs, the forward ant has to collect path information. However, in ARAMA, the forward ant takes

into account not only the hop count factor but also the links local heuristic along the route such as the node's battery power and queue delay. ARAMA defines a value called *grade*, calculated by each backward ant, which is a function of the path information stored in the forward ant. At each node, the backward ant updates the pheromone amount of the node's routing table, using the *grade* value. The protocol uses the same *grade* to update pheromone value of all links. In ARAMA the route discovery and maintenance overheads are reduced by controlling the forward ant's generation rate.

HOPNET. This is a hybrid ant colony optimization routing protocol (Wanga et al., 2008) based on ants hopping from one zone to the next. HOPNET is highly scalable for large networks compared to other hybrid protocols. The HOPNET algorithm consists of the local proactive route discovery within a node's neighbourhood and reactive communication between the neighbourhoods. The network is divided into zones which are the node's local neighbourhood. A routing zone consists of the nodes and all other nodes within the specified radius length measured in hops. A node may be within multiple overlapping zones and zones could vary in size. The nodes can be categorized as interior and boundary (or peripheral) nodes with respect to the central node. Each node has two routing tables: *Intrazone Routing Table* (IntraRT) and *Interzone Routing Table* (InterRT). The IntraRT is proactively maintained so that a node can obtain a path to any node within its zone quickly. This is done by periodically sending out forward ants to sample path within its zone and determine any topology changes. Once a forward ant reaches a destination, a corresponding backward ant is sent back along the path discovered. The InterRT stores the path to a node beyond its zone. This source routing table is setup on demand as routes outside a zone is required. The peripheral nodes of the zone are used to find routes between zones. For small number of nodes, due to the constant movement of border nodes, new routes have to be determined continuously resulting in more delay than other hybrid routing protocols.

Distributed Ant Routing (DAR). In DAR (Rosati et al. 2008) routes are created on-demand, in order to have a low routing signalling load. Forward ants collect information only about the identities of the crossed nodes and move towards the destination choosing the next hop only on a pheromone basis. The amount of pheromone deposited by backward ants on each crossed link is constant. In DAR, in each node the routing tables are stochastic: next hop is selected according to weighted probabilities, calculated on the basis of the pheromone trails left by ants. When a node receives a datagram with destination d, if the routing entry for d is available, then the datagram is forwarded. Otherwise, the datagram is buffered and forward ants are sent out at constant rate r_{ae} (ant emission rate) in order to search a path to d. The forward ant goes to each node according to the probabilities for the next hop in the routing table at the current node. Thus, the forwarding of the forward ant is *probabilistic* and allows exploration of paths available in the network. Datagrams are routed *deterministically* based on the maximum probability at each intermediate node from the source node to the destination node. This process creates a complete global route by using local information. The simplicity of the protocol could be helpful in achieving seamless routing in networks constituted by heterogeneous elements.

Ant-based Distributed Route Algorithm (ADRA). In ADRA (Zheng et al., 2008) ants move across the network between randomly chosen pairs of nodes. Along the path, ants deposit simulated pheromones as a function of their hop distance from their source node, the quality of the link, the congestion encountered on their journey, the current pheromones the nodes possess and the velocity at which the nodes move. The node also ages the link by pheromones evaporating. An ant selects its path at each intermediate node according to the

distribution of simulated pheromones at each node. In order to accelerate the convergence rate of the congestion problem and the shortcut problem, the parameters are given with different weight values to update the probability routing table. The ADRA system exhibits many attractive features of distributed control.

Ant-based Energy Aware Disjoint Multipath Routing Algorithm (AEADMRA). Earlier research has proposed several unipath routing protocols for MANETs. However, due to the dynamic topology of these networks, the single path is easily broken leading to a new route discovery process and an increase in both delay and control overhead. AEADMRA (Wu et al., 2007) was proposed to alleviate these problems. This algorithm is based on swarm intelligence and especially on the ant colony based meta-heuristic. AEADMRA has been designed to enable path accumulation in route request/reply packets and discover multiple energy aware routing paths with a low routing overhead.

ImProved Ant Colony Optimization algorithm for mobile ad hoc NETworks (PACONET). PACONET is a reactive routing protocol (Osagie et al., 2008) which also uses two kinds of agents: forward ant (FANT) and backward ant (BANT). The FANT explores the paths of the network in a restricted broadcast manner in search of routes from a source to a destination. The BANT establishes the path information acquired by the FANT. These agents create a bias at each node for its neighbours by leaving a pheromone amount from its source. Data packets are stochastically transmitted towards nodes with higher pheromone concentration along the path to the destination. FANTs also travel towards nodes of higher concentration but only if there is no unvisited neighbour node in the routing table. The rows of the routing table represent the neighbours of a node and the columns represent all the nodes in the network. Each pair (row, column) in the routing table has two values: a binary value indicating if the node has been visited and the pheromone concentration. All possible paths are explored to find the best path towards the destination. The node with the highest pheromone is chosen as the next hop after the FANT has determined that it has not visited the node before.

AntHocNet. This is a hybrid routing protocol (Caro et al. 2004) consisting of both reactive and proactive components. Nodes do not maintain routes to all possible destinations at all the times and generate mobile agents only at the beginning of a data session. The mobile agents search for multiple paths to the destination and these paths are set up in the form of pheromone tables indicating their respective quality. During the course of the data session, the paths are continuously monitored and improved in a proactive manner.

4. ACO techniques in load balancing

Routing problem in MANET is very challenging and difficult due to the mobility of nodes. Ant colony optimization is an efficient optimization technique used to find the optimum shortest route in the ad-hoc network. However, other problems has to be addressed in order to obtain full efficiency. Network congestion is one of these problems and is present when load is not perfectly balanced. In this case the simple implementation of ant behaviour is not sufficient and some adjustments have to be applied. Load-balancing becomes one of the important issues since the network performance such as network throughput and end-to-end delay can be improved if the loads are well balanced. In the following subsections some ACO algorithms for load balancing, improving efficiency and stability of classical ACO algorithms, will be described.

4.1 ABC: Ant Based Control

Ant based control system (ABC) (Schoonderwoerd et al., 1996) was designed to solve the load-balancing problem. Each row in the pheromone table represents the routing preference for each destination, and each column represents the probability of choosing a neighbour as the next hop. Along the paths, incoming ants update the entries in the pheromone table of a node. In order to mitigate stagnation, three approaches are adopted:

* aging;
* delaying;
* noise.

Aging is designed to discourage ants from following the trails of an ant that has travelled a longer path to some destination. In contrast to evaporation, aging may induce an ant to select a nonoptimal link, if the path from a node to its destination is very long. Used in conjunction with aging, delaying is designed to reduce the flow rates of ants from a congested node to its neighbours. By slowing down the ants originating from a congested node, the amount of pheromone they deposit reduced with time because of the aging process. Noise approach enables ants to choose a path randomly not taking into account the influence of the pheromone table. Thus, ants can explore new and better routes, particularly in dynamic networks.

In one of the ramifications of the ABC system (Guérin, 1997), *smart ants* are adopted to enhance performance. While in classic ABC an ant updates only the entry corresponding to the source node in the pheromone table of each node it passes, smart ants update all the entries in the pheromone table at each node. By performing more pheromone updates at every intermediate node, smart ants are more complex but fewer smart ants are needed in order to achieve the same routing purpose.

In another ramification of the ABC system (Subramanian et al., 1997), two kinds of ants are proposed: *regular ant* and *uniform ant*. Regular ant uses the accumulated cost of a path to determine the amount of pheromone to deposit. A regular ant that travels a higher cost path to a destination node deposits lesser pheromone. Unlike regular ants, uniform ants choose their next nodes in a random way. Moreover, while regular ants use the accumulated cost in the direction from source to destination, uniform ants use the accumulated cost in the reverse direction to establish the amount of pheromone to deposit.

4.2 Ant-Net

Ant-Net algorithm (Caro & Dorigo, 1997) was originally designed for routing in packet-switched networks. Unlike traditional routing algorithms which focused on shortest path routing, AntNet aims to optimize the performance of the entire network. In AntNet, forward ants are launched at regular intervals from a source node N_s to a destination node N_d to discover a feasible low-cost path. Backward ants travel from N_d to N_s to update pheromone tables at each intermediate node. From N_s to N_d, a forward ant selects the next hop node N_i using a random scheme that take into consideration of both the probability of choosing N_i, called P_{id} and a heuristic correction factor I_{ni}. While I_{ni} depends on the queue length at N_i, P_{id} is a selection probability which can be viewed as a pheromone concentration that can be reinforced by other ants.

As a forward ant travels from source node to destination node, it collects statistics such as the local data traffic condition on each intermediate node and the trip time to reach N_i. When a forward ant arrives at destination, a backward ant will be activated. This ant

updates the probabilistic pheromone table at each intermediate node N_i and the estimated trip time for the path N_s - N_i. Backward ants update the selection probability by determining the *goodness* of the trip times of forward ants, and the amount of reinforcement using a *squash* function.

The *goodness* of the trip time is a relative measure determined comparing the current trip time to the current statistical estimates and the confidence interval of the best trip time. The squash function is a nonlinear function that is more sensitive in rewarding solutions with higher goodness values.

This algorithm (called Ant-Net-CL) alleviates the problem of stagnation. However, using both forward and backward ants generally doubles the routing overhead.

In another version of Ant-Net, called Ant-Net-CL (Caro & Dorigo, 1998) forward ants travel from a source to a destination in high priority queues, and backward ants estimate the trip time (by size of queuing data, links' bandwidth and delay), update local traffic statistics, and determine and deposit the amount of probability to reinforce. Since backward ants determine the amount of reinforcement using real time statistics, the routing information is comparatively more accurate and up-to-date.

Another ramification of AntNet (Baran & Sosa, 2000) is characterized by the five following distinguishing features from AntNet:

1. intelligent initialization of AntNet;
2. intelligent pheromone updates after link or node failures;
3. use of noise to mitigate stagnation;
4. deterministic rather than probabilistic selection of a node;
5. restricting the number of ants inside a network.

The first feature was included to regulate the exploration ants in the initial stage. The original entries in a routing table consist of a uniform distribution of probabilities which may not reflect the states of the network. Taking into consideration the a-priori knowledge of the network, ants in this work are configured to select neighbouring nodes with a higher initial probability. While AntNet did not consider situations of link failures, this version suggests that in case of link failures, the corresponding probability of a link that fails will be set to zero and will be distributed evenly among the remaining neighbouring nodes. The third feature deals with noise, where some ants select paths uniformly without considering the effect of pheromone concentration. The fourth feature uses a deterministic approach for the selection of the next hop. However, this approach may lead to a possible infinite looping. The fifth feature suggests to fix an upper bound in number of ants inside a network. Although restricting the number of ants may reduce routing overhead and possible congestion, it also places a restriction on the frequency of launching ants which may lead to possible reduction in the adaptiveness of the routing algorithm.

4.3 ASGA (Ant System with Genetic Algorithm) and SynthECA (Synthetic Ecology of chemical Agents)

Ant system with genetic algorithm (ASGA) was designed to solve problems of point-to-point, point to multipoint and cycle (multipath) routing in circuit-switched networks (White et al. 1998). In ASGA explorer ants are used to update pheromone tables. Although similar to AntNet, explorers travel in a round trip, but unlike backward ants in AntNet, explorers deposit the same amount of pheromones in their return trips. In addition, evaporation agents and pheromone heuristic control were used to mitigate stagnation. The genetic

algorithm was added to increase the adaptivity of ants. For instance, if the best path is congested, it increases the likelihood of ants to find an alternative path. However, unlike the ABC system, ASGA was not designed to solve the load-balancing problem in circuit-switched networks.

Subsequently, in order to solve this problem, ASGA was generalized to a framework called *Synthetic ecology of chemical agents* (*SynthECA*) (White, 2000). SynthECA was also designed to solve other problems such as fault location detection in circuit-switched networks. Although SynthECA was not designed with any specific type of ants, all ants in SynthECA are characterized with a combination of the following:

- emitters;
- receptors;
- chemistry;
- migration decision function;
- memory.

Emitters are used to generate different types of *chemical* pheromone. Pheromones are represented by strings such as "1100" or "10#1." While each type of pheromone corresponds to a *genotype*, each string corresponds to a *chromosome* in GA. Pheromone is generated by an emitter decision function (EDF). As in GA, the operations of crossover and mutation are applied in the EDF to evolve the pheromone types. With various pheromone types and pheromone reactions, ants can be designed to send and sense more types of signals in their stigmergic communication.

In order to sense local pheromone changes generated by emitters, a *receptor* is used. Using receptor detection function (RDF), a receptor senses different types of pheromone. By configuring ants with different EDFs and RDFs, more sophisticated pheromone manipulation techniques such as privileged pheromone laying and pheromone heuristic control can be realized.

Chemistry is a set of rules (inspired by GA) that specifies pheromone reactions. In SynthECA, ants use pheromone reactions to send out control information to other ants. In the set of rules, five types of pheromone reactions are specified as follows:

1. $X \rightarrow$ "nothing:" this is similar to evaporation;
2. $X+Y \rightarrow Y$: this is applied when two ants are competing for a path and only one ant will prevail;
3. $X+Y \rightarrow Z$: this rule is used to report the status of network resources (e.g., poor connection quality);
4. $X+Y \rightarrow X+Z$: this rule, in computational terms, represents a conditional construct. A pheromone type Y is transformed into another type of pheromone Z in the presence of a specific type of pheromone X;
5. $X+Y \rightarrow W+Z$: this rule allows two ants X and Y to jointly communicate both inhibitory (e.g., W) and excitatory (e.g., Z) messages to other ants.

While a *migration decision function* is a set of rules determining the next hop of an ant, pheromones (i.e., labels and concentrations) and the state of an ant are stored in the ant's *memory*.

Using a combination of the above five components, several types of ants such as *route finding agent* (RFA), *connection monitoring agent* (CMA) and *fault detection agent* (FDA) can be configured to solve different networking problems. RFAs include *explorers*, *allocators* and *deallocators*. An explorer is used to find a path from a source to a destination and is configured with an emitter for a single type of pheromone and three receptors for sensing

pheromone, measuring link costs and detecting quality of links. Using a probability function, an explorer chooses a path taking into account the pheromone and the cost of the path. Travelling from source to destination, explorer records all the nodes it passed. When it reaches destination, it returns to via the same path and deposits pheromone along the way, which may influence the pheromone concentration of other types. Explorers are also programmed to also take into consideration the quality/reliability of the link. While an allocator is used to obtain link resources, a deallocator release resources previously acquired by an allocator.

CMA's are activated if the quality of service changes. A CMA evaluates the quality of a link using local traffic statistics and it deposits a special type of pheromone (called *q-chemical*) to indicate the quality of the associated link. CMAs use q-chemical to indirectly communicate the quality of links to FDAs while they circulate the network for diagnostics purposes.

4.4 MACO: Multiple Ant Colony Optimization

In MACO (Sim & Sun, 2003), more than one colony of ants are used to search for optimal paths, and each colony of ants deposits a different type of pheromone represented by a different colour. Although ants in each colony respond to pheromone from its own colony, MACO is augmented with a *repulsion* mechanism preventing ants from different colonies to choose the same optimal path. In order to establish connections between two gateways, two groups of mobile agents (e.g., MAG1 and MAG2), acting as routing packets, construct, manipulate and consult their own routing tables. In MACO, each group of mobile agents corresponds to a colony of ants, and the routing table of each group corresponds to a pheromone table of each colony. Even though MAG1 and MAG2 may have their own routing preferences, they also take into consideration the routing preferences of the other group. While the routing preferences of ants are recorded in their pheromone tables, the routing preferences of mobile agents are stored in their routing tables. In constructing its routing table, MAG1 (respectively, MAG2) consults the routing table of MAG2 (respectively, MAG1) in order to avoid routing packets to those paths that are highly preferred by the other group. This increases the chance of distributing data traffic. By adopting the MACO approach, it may be possible to reduce the likelihood that all mobile agents establish connections using *only* the optimal path. The advantage of using MACO is that it is more likely to establish connections through multiple paths to help balance the load but does not increase the routing overhead.

5. Applications and new directions

The works surveyed in the previous sections addressed the application of swarm intelligence and in particular ACO algorithms to solve the routing problem and/or load balancing in MANETs. However, ACO algorithms have been applied to solve different kinds of problems in MANETs. Reduction of power consumption is one of these important issues in ad hoc wireless networks. Mobile nodes are powered by battery and an efficient utilization of battery energy is very important. When a node exhausts its available energy, it ceases to work and the lack of mobile nodes can result in network partitioning. In recent years, some improvement in ACO routing algorithms were proposed in order to reduce the communication load related to energy spent with communications (De Rango & Tropea, 2009; Zyiadi et al., 2009; Li & Shi, 2009). In (De Rango & Tropea, 2009) has been proposed a novel routing algorithm able to satisfy multiple metrics for a multi-objective optimization

like end-to-end delay, load balancing and energy savings. Innovation factor of this proposal has been the pheromone updating policy and the joint metric used.

The demand for quality of services (QoS) in MANETs suggested also the development of QoS routing strategies computing paths that are suitable for different type of traffic generated by various applications while maximizing the utilizations of network resources. The main problem to be solved by QoS routing algorithm is the Multi-Constraint Path problem. Instead of using a shortest path algorithm based on statically configured metrics, as in traditional routing protocols, the algorithm must select several alternative paths that are able to satisfy a set of constraints regarding, for instance, end-to-end delay bounds and bandwidth requirements. Several approaches (Shokrani & Jabbehdari, 2009; Liu et al., 2007; Liu et al. 2008, Zhang & Li, 2008) have been proposed to address the complexity of multi-constrained path computation problem using ACO approach.

Another interesting issue in MANETs, in which has been employed the behavioural principle present in ant colonies, is address management. In ad-hoc networks, address management is a particularly tough challenge, because of their dynamically changing topology, and the sort of events that occur in their environment. In (Pachon & Madrid, 2009) has been proposed a solution to this problem, involving the self-organization and emergency principles governing the behaviour of ant colonies.

All the works presented in this chapter show how swarm intelligence and in particular the behaviour of ant colonies have inspired a number of successful methods and techniques to solve different problems in MANETs. However, the potential of this distributed intelligent technique is more clear when applied to other dynamic network scenarios. In (De Rango et al., 2008), for example, has been showed how minimum hop count and load balancing metrics based on ant behaviour over a HAPs mesh can lead to a better management of the system resources and an increase in the number of calls admitted by the system. This is only one example of the wide range of applications covered by swarm intelligence techniques which underlines the importance of this approach in communication networks.

6. References

Baran, B. and Sosa, R. (2000). A new approach for AntNet routing, *Proc. 9th Int. Conf. Computer Communications Networks*, 303-308, 0-7803-6494-5, Las Vegas, USA

Baras, J. S. & Mehta, H. (2003). A Probabilistic Emergent Routing Algorithm for Mobile Ad Hoc Networks, *Proceedings of the Conference on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt '03)*, Sophia-Antipolis, France, (March 2003)

Bonabeau, E.; Dorigo, M. & Theraulaz, G. (1999). *Swarm Intelligence. From Natural to Artificial Systems*, Oxford University Press, 0-19-513159-2, Oxford

Bonabeau, E.; Dorigo, M. & Theraulaz, G. (2000). Inspiration for optimization from social insect behaviour, *Nature*, Vol. 406, No. 6, (July 2000) 39-42, 0028-0836

Buruhanudeen, S.; Othman, M., Othman, M. & Mohd Ali, B. (2007). Existing MANET routing protocols and metrics used towards the efficiency and reliability - an overview, *Proceedings of the 14th Int. Conference on Telecommunications and 8th Malaysia. Int. Conference on Communications (ICT - MICC 2007)*, 231-236, Penang, Malaysia, (May 2007)

Caro, G. D. & Dorigo, M. (1998). AntNet: Distributed stigmergetic control for communications networks. *J. Artif. Intell. Res.*, Vol. 9, No. 3, 317-365

Caro, G. D. & Dorigo, M. (1997). AntNet: A Mobile Agents Approach to Adaptive Routing, In : *IRIDIA – Technical Report Series*, IRIDIA, 97-12, Universitè Libre de Bruxelles.

Černý, V. (1985). A thermo dynamical approach to the traveling salesman problem, *Journal of Optimization Theory and Applications*, Vol. 45, No. 1, (January 1985) 41-51

Deneubourg, J. L.; Aron, S. Goss, S. & Pasteels, J. M. (1990). The self-organizing exploratory pattern of the argentine ant, *Journal of Insect Behavior,* Vol. 3, No. 2, 159-168

De Rango, F.; Tropea, M.; Provato, A.; Santamaria, A.F. & Marano, S. (2008). Minimum Hop Count and Load Balancing Metrics Based on Ant Behavior over HAP Mesh, *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE* , 1-6, 1930-529X, New Orleans, (Nov.-Dec. 2008)

De Rango, F. & Tropea, M. (2009). Energy saving and load balancing in wireless ad hoc networks through ant-based routing, *International Symposium on Performance Evaluation of Computer & Telecommunication Systems*, SPECTS 2009., Vol. 41, (July 2009), 978-1-4244-4165-5

Di Caro, G.; Ducatelle, F. & Gambardella, L. M. (2004). AntHocNet: an Ant-Based Hybrid Routing Algorithm for Mobile Ad Hoc Networks, In: *IDSIA-25-04-2004 Technical Report*, IDSIA/USI-SUPSI, 1-12, Dalle Molle Institute for Artificial Intelligence, Switzerland

Dorigo, M.; Maniezzo, V. & Colomi, A. (1996). The Ant System: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man and Cybernetics - Part B*, Vol. 26, No. 1, 29-41

Dorigo, M. & Gambardella, L. M. (1997). Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, (April 1997) 53-66, 1089-778X(97)03303-1

Dorigo, M.; Caro, G. D. & Gambardella, L. M. (1999). Ant algorithms for discrete optimization, *Artificial Life*, Vol. 5, No. 2, (April 1999) 137-172, 1064-5462

Dorigo, M. & Stützle, T. (2004). *Ant Colony Optimization*, MIT Press, 0-262-04219-3, Cambridge, MA, UK

Dorigo, M.; Birattari, M. & Stützle, T. (2006). Ant Colony Optimization. Artificial Ants as a Computational Intelligence Technique, In: *IRIDIA - Technical Report Series* , IRIDIA, 1-12, 1781-3794, Université Libre de Bruxelles

Ducatelle, F.; Di Caro, G. & Gambardella, L. M. (2005). Using ant agents to combine reactive and proactive strategies for routing in mobile ad hoc networks, *International Journal of Computational Intelligence and Applications*, Vol. 5, No. 2, (June 2005) 169-184

Fenet, S. & Hassas, S (1998). An ant system for multiple criteria balancing, *Proceedings of 1st International Workshop on Ants Systems*, Brussels, (September 1998)

Glover, F. & Laguna, M. (1997). *Tabu Search*, Kluwer Academic Publishers, 079239965X, Norwell, MA, USA

Guérin, S. (1997). Optimization Multi-Agents en Environment Dynamique: Application au Routage Dans les Réseaux de Telecommunications, In: *DEA*, Univ. Rennes I, Ecole Nat. Supér. Télécommun. Bretagne, France

Günes, M. & Spaniel, O. (2003). Ant-Routing-Algorithm for mobile multi-hop ad-hoc networks, *Proceedings of Int. Conference on Network Control and Engineering for QoS (Net-Con 2003)*, 120-138, Muscat, Oman, (October 2003)

Hossein, O. & Saadawi, T. (2003). Ant routing algorithm for mobile ad hoc networks (ARAMA), *Proceedings of the 22nd IEEE International Performance, Computing, and Communications Conference*, 281-290, Phoenix, USA, (April 2003)

Kamali, S. & Opatrny, J. (2008). A Position Based Ant Colony Routing Algorithm for Mobile Ad-hoc Networks, *Journal of Networks*, Vol. 3, No. 4, (April 2008) 31-41

Li, Z. & Shi, H. (2008). A Data-Aggregation Algorithm Based on Adaptive Ant Colony System in Wireless Sensor Networks, *Congress on Image and Signal Processing2008, CISP '08*. Vol. 4, 449-453, 978-0-7695-3119-9, , Sanya, China, (May 2008)

Liu, M.; Sun, Y.; Liu, R. & Huang, X. (2007). An Improved Ant Colony QoS Routing Algorithm Applied to Mobile Ad Hoc Networks, *International Conference on Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007*, 1641-1644, 978-1-4244-1311-9, Shanghai, (September 2007)

Liu, Y.; Zhang, H.; Ni, Q.; Zhou, Z. & Zhu G. (2008). An Effective Ant-Colony Based Routing Algorithm for Mobile Ad-Hoc Network. *Circuits and Systems for Communications, 2008. ICCSC 2008. 4th IEEE International Conference on* , 100-103, 978-1-4244-1707-0, Shanghai, China, (May 2008)

Lourenço, H. R.; Martin, O. & Stützle, T. (2002). Iterated local search, In: *Handbook of Metaheuristics of International Series in Operations Research Management Science*, Glover, F., & Kochenberger, G., (Eds.), 321-353, Kluwer Academic Publishers, Norwell, MA, USA

Osagie, E.; Thulasiraman, P. & Thulasiram, R. K. (2008). PACONET: imProved Ant Colony Optimization routing algorithm for mobile ad hoc NETworks, *Proceedings of 22nd International Conference on Advanced Information Networking and Applications*, 204-211, Okinawa, Japan, (March 2008)

Pachon, A. & Madrid, J.M. (2009). Application of an ant colony metaphor for network address management in MANETs. *Communications, 2009. LATINCOM '09. Conference on IEEE Latin-American*, 1-6, 978-1-4244-4387-1, Medellin, (September 2009)

Rajagopalan, S. & Shen, C. (2005). ANSI: a unicast routing protocol for mobile ad hoc networks using swarm intelligence, *Proceedings of the International Conference on Artificial Intelligence*, 24-27, Las Vegas, USA, (June 2005)

Rosati, L.; Berioli, M. & Reali, G. (2008). On ant routing algorithms in ad hoc networks with critical connectivity, *Ad Hoc Networks*, Vol. 6, No. 6, (August 2008) 827-859

Schoonderwoerd, R.; Holland, O., Bruten, J. & Rothkrantz, L. (1996). Ants for Load Balancing in Telecommunication Networks, In: *HPL-96-35 Technical Report*, Hewlett Packard Laboratory, Bristol, UK

Sim, K. M. & Sun, W. H. (2003). Ant Colony Optimization for Routing and Load-Balancing: Survey and New Directions, *IEEE Transactions on Systems, Man and Cybernetics - Part A: System and Human*, Vol. 33, No. 5, 560-572

Shokrani, H. & Jabbehdari, S. (2009). A novel ant-based QoS routing for mobile ad hoc networks, *Proceedings of the First international Conference on Ubiquitous and Future Networks*, 79-82, 978-1-4244-4215-7, Hong Kong, (June 2009)

Stützle, T. & Hoos, H. (1998). The MAX-MIN Ant System and Local Search for Combinatorial Optimization Problems: Towards Adaptive Tools for Combinatorial Global Optimization, In: *Meta-Heuristics: Advances and Trends in Local Search*

Paradigms for Optimization, Vos, S.; Martello, S., Osman, I. H. & Roucairol, C., (Eds.), 313-329, Kluwer Academic Publishers, Norwell, MA, USA

Stützle, T. (1999). Local Search Algorithms for Combinatorial Problems: Analysis, Improvements, and New Applications, In: *DISKI*, Infix Publishers, Vol. 220, Sankt Augustin, Germany

Subramanian, D.; Druschel, P. & Chen, J. (1997). Ants and reinforcement learning: A case study in routing in dynamic networks, *Proc. Int. Joint Conf. Artificial Intelligence*, 832–838, 1-555860-480-4, Palo Alto, USA

Tarasewich, P. & McMullen, P. R. (2002). Swarm Intelligence: Powers in numbers, *Communications of the ACM*, Vol. 45, No. 8, (August 2002) 62-67, 0001-0782

Wang, J.; Osagie, E., Thulasiraman, P. & Thulasiram, R. K. (2009). HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network, *Ad Hoc Networks*, Vol. 7, No. 4, (June 2009) 690-705

White, T.; Pagurek, B. & Oppacher, F. (1998). ASGA: Improving the ant system by integration with genetic algorithms, *Proceedings of 3rd Genetic Programming Conf.*, 610–617, (July 1998)

White, T. (2000). SynthECA: A Society of Synthetic Chemical Agents. *Ph.D.dissertation*, Carleton University, Northfield, MN

Wu, Z.; Song; H., Jiang, S. & Xu, X. (2007). Ant-based Energy Aware Disjoint Multipath Routing Algorithm in MANETs, *Proceedings of International Conference on Multimedia and Ubiquitous Engineering*, 674-679, Vol. 1, No. 1, Seoul, Korea, (April 2007)

Zhang, Y. & Li, Z. (2009). HCRS: A Routing Scheme for Ad Hoc Networks as a QoS Guarantee Primitive, *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on* , 1-4, 978-1-4244-3692-7, Beijing, (September 2009)

Zheng, X.; Guo, W. & Liu, R. (2004). An ant-based distributed routing algorithm for ad-hoc networks (ADRA), *Proceedings of International Conference on Communications, Circuits and Systems (ICCCAS 2004)*, 412-417, Vol. 1, No. 1, Chengdu, China, (June 2004)

Ziyadi, M.; Yasami, K. & Abolhassani, B. (2009). Adaptive Clustering for Energy Efficient Wireless Sensor Networks Based on Ant Colony Optimization, *Communication Networks and Services Research Conference*, CNSR '09, Seventh Annual, 330-334, 978-1-4244-4155-6, May 2009, Moncton, NB
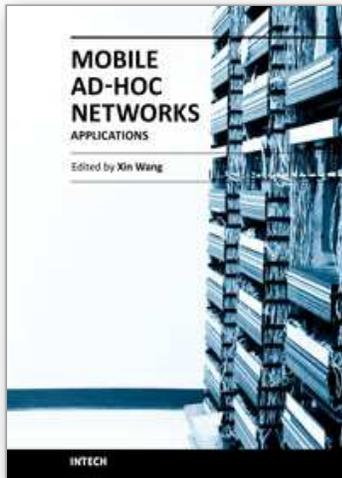
Mobile Ad-Hoc Networks: Applications

Edited by Prof. Xin Wang

Being infrastructure-less and without central administration control, wireless ad-hoc networking is playing a more and more important role in extending the coverage of traditional wireless infrastructure (cellular networks, wireless LAN, etc). This book includes state-of the-art techniques and solutions for wireless ad-hoc networks. It focuses on the following topics in ad-hoc networks: vehicular ad-hoc networks, security and caching, TCP in ad-hoc networks and emerging applications. It is targeted to provide network engineers and researchers with design guidelines for large scale wireless ad hoc networks.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Floriano De Rango and Annalisa Socievole (2011). Meta-Heuristics Techniques and Swarm Intelligence in Mobile Ad Hoc Networks, Mobile Ad-Hoc Networks: Applications, Prof. Xin Wang (Ed.), ISBN: 978-953-307-416-0, InTech, Available from: http://www.intechopen.com/books/mobile-ad-hoc-networks-applications/meta-heuristics-techniques-and-swarm-intelligence-in-mobile-ad-hoc-networks

INTECH

open science | open minds