Swarm Intelligence for Routing in Communication Networks

I. Kassabalidis*, M.A. El-Sharkawi*, R.J.Marks II*, P. Arabshahi†, A.A. Gray†,

* Dept. of Electrical Eng., Box 352500
University of Washington
Seattle, WA 98195 USA
[ikas,melshark,marks]@ee.washington.edu

†Jet Propulsion Laboratory
4800 Oak Grove Drive, MS 238-343
Pasadena, CA 91109 USA
[payman,gray]@jpl.nasa.gov

Abstract – **Swarm intelligence, as demonstrated by natural biological swarms, has numerous powerful properties desirable in many engineering systems, such as network routing. In addition, new paradigms for designing autonomous and scalable systems may result from analytically understanding and extending the design principles and operations exhibited by intelligent biological swarms. A key element of future design paradigms will be emergent intelligence – simple local interactions of autonomous swarm members, with simple primitives, giving rise to complex and intelligent global behavior. Communication network management is becoming increasingly difficult due to the increasing size, rapidly changing topology, and complexity of communication networks. A new class of algorithms, inspired by swarm intelligence, is currently being developed that can potentially solve numerous problems of modern communications networks. These algorithms rely on the interaction of a multitude of simultaneously interacting agents. A survey of such algorithms and their performance is presented here.**

I. Introduction

Modern communication networks are becoming increasingly diverse and heterogeneous. This is the consequence of the addition of an increasing array of devices and services, both wired and wireless. The need for seamless interaction of numerous heterogeneous network components represents a formidable challenge, especially for networks that have traditionally used centralized methods of network control. This is true for both packet-switched and virtual-circuit networks, and the Internet, which is becoming an ever more complex collection of a diversity of subnets. The need to incorporate wireless and possibly ad-hoc networks into the existing wire-link infrastructure renders the requirement for efficient network routing even more demanding.

Routing algorithms in modern networks must address numerous problems. Two of the usual performance metrics of a network are average throughput and delay. The interaction between routing and flow control affects how well these metrics are jointly optimized. Bertsekas and Gallager [1] note that the balance of delay and throughput is determined by the flow-control scheme – good routing results in a more favorable delay-throughput curve. *Quality of service* (QoS) guarantee is another important performance measure [2,3]. Here, a user might require a guaranteed allocation of bandwidth, a maximum delay, or a minimum hop-count. Such guarantees only make sense for virtual-circuit networks [2]. This is because in applications that require logical connections there is demand for a minimum flow rate of data. This is unlike packet-switched types of service where best-effort routing is implemented. Although logical connections use static routing, the establishment of the connection is prone to the same problems that affect routing in the rest of the network [3].

Current routing algorithms are not adequate to tackle the increasing complexity of such networks. Centralized algorithms have scalability problems; static algorithms have trouble keeping up-to-date with network changes; and other distributed and dynamic algorithms have oscillations and stability problems [1].

Swarm intelligence routing provides a promising alternative to these approaches. Swarm intelligence utilizes mobile software agents for network management. These agents are autonomous entities, both proactive and reactive, and have the capability to adapt, cooperate and move intelligently from one location to the other in the communication network [4]. Swarm intelligence, in particular, uses *stigmergy* (i.e. communication through the environment) for agent interaction [5,6,7,9]. Swarm intelligence exhibits *emergent behavior* wherein simple interactions of autonomous agents, with simple primitives, give rise to a complex behavior that has not been specified explicitly [8].

In Section II, we give an overview of existing routing algorithms, including their individual merits and weaknesses. An discussion of swarm intelligence and its attractive features appears in Section III and, in Section IV, we present some specific swarm-based algorithms and discuss their applicability and performance. Section V concludes the paper.

II. Routing Algorithms

Routing algorithms can be classified as static or dynamic, and centralized or distributed. *Centralized* algorithms are usually used in legacy routing systems and have problems with scalability and inordinate demand for managing decisions requiring human attention [10]. Another drawback is the inability of the network to recover in case of failure at the central controlling station. *Static* routing assumes that network conditions are time-invariant. The method does not assess the network load when trying to find the shortest-path route. Ahuja, Magnanti, and Orlin [11] show that maximizing throughput for a time varying load in a limited-capacity transmission line is an NP-complete problem. *Adaptive* routing schemes also have problems, including inconsistencies arising from node failures and potential oscillations that lead to circular paths and instability [1]. Another problem with adaptive algorithms applied to ad-hoc networks arises when changes in the network occur too frequently to allow routing updates to propagate throughout all network nodes. A network is called *combinatorially stable* if it changes sufficiently slowly for the routing updates to be propagated to all the nodes [3].

Routing algorithms can also be classified as *minimal* or *non-minimal*. Minimal routing allows packets to follow only minimal cost paths, while non-minimal routing allows more flexibility in choosing the path by utilizing other heuristics [2]. Minimal routing can further be subdivided into *optimal routing* and *shortest-path routing*. In the former, the objective is to optimize the mean flow of the entire network; while in shortest-path routing the goal is to find the minimum-cost path between two nodes [1,7].

Another class of routing algorithms is one where the routing scheme guarantees specified QoS requirements pertaining to delay and bandwidth. These algorithms are usually message based, *i.e.* they find a feasible path satisfying the QoS constraints based on an exchange of messages between the nodes [11]. These algorithms have the tendency to temporarily overuse network resources until they find the appropriate path. The Dijkstra and Bellman-Ford algorithms [1] are examples.

Yet another form of network control, which relies heavily on routing, is that of *load balancing* [7,9,19,20]. Here the goal is to balance the load throughout all network resources without idleness and overloading.

III. Swarm Intelligence Overview

Swarm Intelligence appears in biological swarms of certain insect species. It gives rise to complex and often intelligent behavior through complex interaction of thousands of autonomous swarm members. Interaction is based on primitive instincts with no supervision. The end result is accomplishment of very complex forms of social behavior and fulfillment of a number of optimization and other tasks [6].

The main principle behind these interactions is called *stigmergy*, or communication through the environment. An example is pheromone laying on trails followed by ants. Pheromone is a potent form of hormone that can be sensed by ants as they travel along trails. It attracts ants and therefore ants tend to follow trails that have high pheromone concentrations. This causes an *autocatalytic* reaction, i.e., one that is accelerated by itself. Ants attracted by the pheromone will lay more of the same on the same trail, causing even more ants to be attracted.

Another form of stigmergy alters the environment in such a manner as to promote further similar action by the agents. This process is dubbed *task-related stigmergy*. An example is sand grain laying by termites when constructing nests [6]. In the initial stages of construction, termites lay sand grains at random locations. This stimulates further laying by other members of the swarm, until a single heap of sand grains randomly reaches a critical mass that is larger than its neighboring heaps. At that point, most termites are attracted to that specific heap, thereby selecting that specific site for construction of their nest.

Swarm intelligence boasts a number of advantages due to the use of mobile agents and stigmergy [2,3,4,6,7,8,9]. These are:

1. *Scalability:* Population of the agents can be adapted according to the network size. Scalability is also promoted by local and distributed agent interactions.
2. *Fault tolerance:* Swarm intelligent processes do not rely on a centralized control mechanism. Therefore the loss of a few nodes or links does not result in catastrophic failure, but rather leads to graceful, scalable degradation.
3. *Adaptation:* Agents can change, die or reproduce, according to network changes.
4. *Speed:* Changes in the network can be propagated very *fast*, in contrast with the Bellman-Ford algorithm [1].
5. *Modularity:* Agents act independently of other network layers [9].
6. *Autonomy*: Little or no human supervision is required.
7. *Parallelism*: Agent's operations are inherently parallel.

These properties make swarm intelligence very attractive for ad-hoc wireless networks. They also render swarm intelligence suitable for a variety of other applications, apart from routing, including robotics [12,13,14] and optimization [15,16,17].

IV. Swarm Intelligence Routing: Examples

There are a number of proposed swarm-based routing algorithms. The most celebrated one is *AntNet* [6,7], an adaptive agent-based routing algorithm that has outperformed the best-known routing algorithms on several packet-switched communications networks. For telephone networks, there also exists a successful application of swarm intelligence dubbed *Ant-Based Control* (ABC) [6,19,20]. Heusse et al. [8] give another interesting example using a variation of swarm routing based on Bellman's principle of dynamic programming [21]. These algorithms are discussed in further detail below.

Other examples also exist and present some interesting variations of swarm-based routing. Oida & Masatoshi [2] present an algorithm dubbed agent-based routing system (ARS) whose main goal is to achieve high utilization of network resources. The authors propose an extension of the *AntNet* algorithm with QoS guarantees, imposing certain restrictions on bandwidth and hop-count. Lipperts & Kreller [9], take a different agent based approach for load balancing. They propose the use of two classes of agents, dubbed *"strategy"* agents and *"load"* agents. The role of the load agents is to find shortest paths based on Dijkstra's algorithm [22]. The strategy agents control the population of the load agents based on network conditions. Varella & Sinclair [23] apply swarm intelligence for virtual-wavelength-path routing. They propose the separation of ants into colonies, with ants being attracted to the pheromone of their own colony and repelled by pheromone of other colonies. Thus, ants of each colony attempt to discover the shortest path independent from the path discovered by other colonies. This leads to a more even load distribution throughout the network.

More examples of swarm based routing applications exist in the literature. Bonabeau et al. [24] discuss an improvement of ant-based algorithms by dynamic programming. Di Caro and Dorigo [25-30] present a number of interesting variations based on ant-like agents. White et al. also discuss various enhancements of routing algorithms in [31-35].

A. AntNet

In the *AntNet* algorithm, routing is determined by means of very complex interactions of *forward* and *backward* network exploration agents ("ants"). The idea behind this sub-division of agents is to allow the backward ants to utilize the useful information gathered by the forward ants on their trip from source to destination. Based on this principle, no node routing updates are performed by the forward ants. Their only purpose in life is to report network delay conditions to the backward ants, in the form of trip times between each network node. The backward ants inherit this raw data and use it to update the routing table of the nodes.

An example of an AntNet routing table is in Table I. The entries of the routing table are probabilities, and as such, must sum to 1 for each row of the network. These probabilities serve a dual purpose: (1) the exploration agents of the network use them to decide the next hop to a destination, randomly selecting among all candidates based on the routing table probabilities for a specific destination (2) the data packets deterministically select the path with the highest probability for the next hop.

Table I. AntNet Routing Table

		Next Hop	
		B	C
Destination	E	0.15	0.85
	F	0.75	0.25

The sequence of actions in AntNet (see Fig. 1) is simple and intuitive:

1. Each network node launches forward ants to all destinations in regular time intervals.
2. The ant finds a path to the destination randomly based on the current routing tables.

3. The forward ant creates a stack, pushing in trip times for every node as that node is reached
4. When the destination is reached, the backward ant inherits the stack.
5. The backward ant pops the stack entries and follows the path in reverse.
6. The node tables of each visited node are updated based on the trip times.

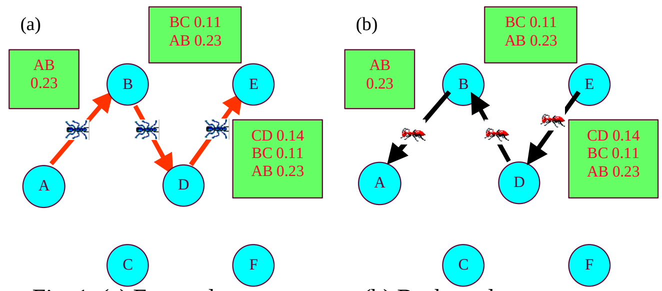

Fig. 1. (a) Forward ant movement (b) Backward ant movement

The update of the routing table is reminiscent of other actor-critic systems, where the raw information contained in the trip time is processed by the critic and then used to train the actor to manage the system more efficiently (see Fig. 2).

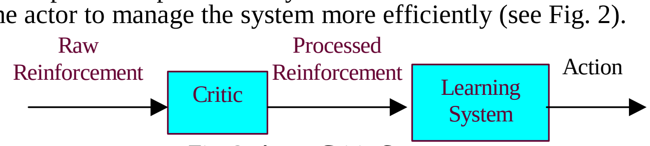

Fig. 2. Actor-Critic System

As an intermediate quantity in the processing of the raw trip time information, we need a measure that takes on small values when the trip time is short relative to the mean and vice-versa. This quantity, r', is derived according to:

$$r' = \begin{cases} \frac{T}{c\mu}, c \geq 1, if\ \frac{T}{c\mu} < 1 \\ 1, otherwise \end{cases}$$

where T is the trip time, μ is average of T, and C is a scaling factor, usually set to 2.

Except for the routing table, each node also possesses a table with records of the mean and variance of the trip time to every destination. A typical trip-time table is Table II.

TABLE II. ANTNET ROUTING TABLE

		Trip Times	
		B	C
Destination	E	0.24	0.02
	F	0.18	0.01

The ratio of the variance to the mean, (σ/μ), is used as a measure of the consistency of the trip times, and to accordingly alter the effect of the trip time on the routing table. Based on the value of r', we determine the relative goodness of the trip time of an ant. Corresponding strategies of either decreasing or increasing the value of r' by a certain amount are then followed, based on setting the threshold for the good/bad trip time to 0.5, and selecting a threshold δ for the (σ/μ) ratio (see Table III).

TABLE III. PROCESSING CASES

	$r' < 0.5$	$r' > 0.5$
$\frac{\sigma}{\mu} > \delta$	$+(1-e^{-a'\frac{\sigma}{\mu}})$	$-(1-e^{-a'\frac{\sigma}{\mu}})$
$\frac{\sigma}{\mu} < \delta$	$-e^{-a\frac{\sigma}{\mu}}$	$+e^{-a\frac{\sigma}{\mu}}$

The principle behind these updates is that small values of r' correspond to small values of T and vice versa. By way of example, and examining the case where the consistency is high and the time is good, we want the processed r' to be even smaller. Doing so underscores the goodness of this trip time and its consistency. Therefore, an exponential quantity is subtracted. This quantity is the exponentially decaying function of the consistency ratio and achieves its highest value when the variance is very small. The decay rate can be controlled through parameters a' and a.

Further positive or negative reinforcement of good or bad routes takes place next, via *negative feedback*. Any positive reinforcement of probability should be negatively proportional to current probabilities, and any negative reinforcement should be proportional to current probabilities. The effect of this is to prevent saturation to 0 or 1 of the routing table probabilities. The node that receives the positive reinforcement is the one from which the backward ant comes. This is the same node chosen by the forward ant as next-hop on the way to its destination. All the other neighbors of the current node need to be negatively reinforced to preserve the unit sum of all the next-hop probabilities. The reinforcement equations are:

$$r_{+} = (1-r')(1-P_{df})$$

$$r_{-} = -(1-r')P_{dn}, n \neq f, n \in N_k$$

where P_{df}, P_{dn} are the previous routing table probabilities, f is the node from which the backward ant comes, N_k is the neighborhood of node k (current node), and d is the destination node (see Fig. 3).

The last step is to update the routing table probabilities using the following rules.

$$P_{df} = P_{df} + r_{+}$$

$$P_{dn} = P_{dn} + r_{-}$$

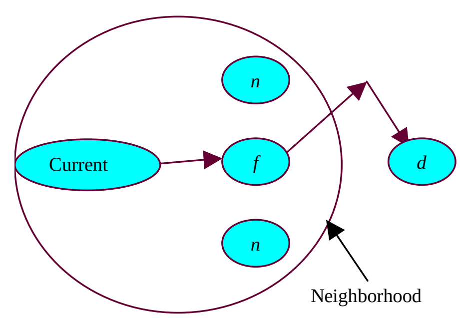

Fig. 3. Current Node Neighborhood

The packets of the network then use these probabilities in a deterministic way, choosing as next hop the one with the highest probability.

B. Ant-based Control

Ant-based Control (ABC) is another successful swarm intelligence based algorithm designed for telephone networks.

This algorithm shares many key features with *AntNet*, but has important differences. The basic principle shared is the use of a multitude of agents interacting using stigmergy. The algorithm is adaptive and exhibits robustness under various network conditions. It also incorporates randomness in the motion of ants. This increases the chance of discovery of new routes. In ABC , the ants only traverse the network nodes probabilistically, while the telephone traffic follows the path of highest probability.

The routing table of every node is the same as *AntNet*. The update philosophy of the routing table is slightly different though. There is only one class of ants, which is launched from the sources to various destinations at regular time intervals. The ants are eliminated once they reach their destination. Therefore, the probabilities of the routing tables are updated as the ant visits the nodes, based on the life of the ant at the time of the visit. The life of the ant is the sum of the delays of the nodes $T = \sum_i D_i$. The delays D_i are given by $D_i = c \cdot e^{-d \cdot S}$, where *c, d* are design parameters and *S* is the spare capacity of each node in the telephone network. Then a step size is defined for that node, according to: $\delta r = \frac{a}{T} + b$, where *a* and *b* are both design parameters. This step size rule is chosen heuristically. It assigns a greater step size to those ants who are successful at reaching the node faster. The routing table is then updated according to:

$$r^i_{i-1,s}(t+1) = \frac{r^i_{i-1,s}(t) + \delta r}{1 + \delta r}$$

$$r^i_{n,s}(t+1) = \frac{r^i_{n,s}(t)}{1 + \delta r}, n \neq i-1$$

where *s* is the source node, *i* is the current node and *i*-1 the previous node.

TABLE IV. ABC ROUTING TABLE

		Next Hop	
		B	*C*
Destination	*E*	0.65	0.35
	F	0.55	0.45

Note that the ant both uses and updates the routing table at the same time. For example, in Table IV, if the source is node *F* and the destination is node *E*, then the ant will update the row for *F* and use the node for *E* to find the next hop. It functions as an ant that is both a forward and a backward ant. The update rules are such that the condition $\sum_n r^i_{n,s} = 1$, where *n* are all the neighbors to *i*, is satisfied.

C. Multiple Round Trip Routing

Another interesting example of swarm intelligence applied to packet-switched networks is Multiple Round Trip routing [7]. As in *AntNet*, nodes launch forward ants in regular intervals. The basic version utilizes the cost measured by the forward ants to update the routing table entries. The forward ant keeps track of the visited notes in a stack J^k and of their associated cost $d^k_{n,d}$. This cost can be the wait time and the transmission delay for each visited node *n*. The cost $d^k_{n,d}$ is defined as the sum of all the costs from node *n* to destination node *d*. Once the destination *d* is reached, then a backward ant is launched, which updates the distance estimation $D^n_{j,d}$ for node *n* to *d* via *j* as follows:

$$D^n_{j,d}(t) = (1-\eta) D^n_{j,d}(t-1) + \eta d^k_{n,d}$$

where η is the learning rate. The routing table probabilities, whose use is similar to those in AntNet, are updated as follows:.

$$p^n_{j,d}(t) = \frac{\left(\frac{1}{D^n_{j,d}(t)}\right)^\beta}{\sum_l \left(\frac{1}{D^n_{l,d}(t)}\right)^\beta}, j \neq l ,$$

where β is a non-linearity factor.

The interesting improvement to this algorithm is based on Bellman's principle of *dynamic programming*. Every node in the path J^k of a source-destination pair *s-d*, is considered a destination. The back-propagating agent will update the routing table of a visited node *n* not just for the destination, but also for the intermediate nodes. Hence the updates occur all at once. For example, on node *n* in Fig. 4, the backward agent will also update the entry for node s_1 as follows:

$$D^n_{s_1,s_1}(t) = (1-\eta) D^n_{s_1,s_1}(t-1) + \eta d^k_{n,s_1}$$

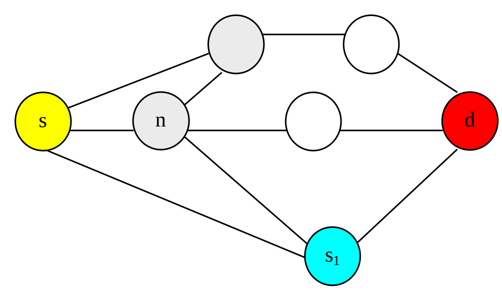

Fig. 4. Multiple Trip Routing Example

AntNet and Multiple Trip Routing are two examples of the class of swarm intelligence algorithms that incorporate round-trip agents. In this type of algorithms, the forward ants act as investigators and the backward ants are the ones who update the routing tables. ABC is an algorithm that incorporates only forward agents, who perform the update as they travel through the network. In this type of algorithms update is faster and more reliable, since there is no delay between the information gathering and the actual update [8].

V. CONCLUSIONS

In this paper, we have presented an overview of *swarm intelligence* applied to network routing. Inherent properties of swarm intelligence as observed in nature include: massive system scalability, emergent behavior and intelligence from low complexity local interactions, autonomy, and *stigmergy*, or communication through the environment. These properties are desirable for many types of networks. Swarm intelligent-based approaches hold great promise for solving numerous problems of ad-hoc power aware networks. *Swarm intelligence* however is a new field and much work remains to be done. Comparison of the performance of swarm-based algorithms has been done by emulation. Analytic proof and models of the swarm-based algorithm performance remain topics of ongoing research.

References

1. D. Bertsekas and R. Gallager. Data Networks. Prentice-Hall, Inc, Upper Saddle River, New Jersey, 1992

2. K. Oida and M. Sekido, "An agent-based routing system for QoS guarantees", *Proc. IEEE International Conference on Systems, Man, and Cybernetics*, Oct. 12-15, pp. 833-838, 1999.

3. Satyabrata Chakrabarti and Amitabh Mishra, "QoS Issues in Ad Hoc Wireless Networks", IEEE Communication Magazine, February 2001.

4. A. Bieszczad, B. Pagurek, and T. White, "Mobile agents for network management", *IEEE Communication Surveys,* Fourth Quarter 1998, vol. 1, no. 1, 1998.

5. Grassé, P.P. 1959. La reconstruction du nid et les coordinations inter-individuelles chez Bellicositermes natalensis et Cubitermes sp. La theorie de la stigmergie: Essai d'interpretation des termites constructeurs. *Ins. Soc*., 6, 41-83.

6. E. Bonabeau, M. Dorigo, and G. Théraulaz, *Swarm intelligence: from natural to artificial systems*, Oxford University Press, 1999.

7. G. Di Caro and M. Dorigo, "AntNet: a mobile agents approach to adaptive routing", Tech. Rep. IRIDIA/97-12, Université Libre de Bruxelles, Belgium

8. M. Heusse, D. Snyers, S. Guérin, and P. Kuntz, "Adaptive agent-driven routing and load balancing in communication network", *Proc. ANTS'98, First International Workshop on Ant Colony Optimization*, Brussels, Belgium, October 15-16, 1998.

9. S. Lipperts and B. Kreller, "Mobile agents in telecommunications networks - a simulative approach to load balancing", *Proc. 5th Intl. Conf. Information Systems, Analysis and Synthesis, ISAS'99*, 1999.

10. Raman, L.: OSI system and Network Managements, IEEE Communication: Managements of Heterogeneous Networks, vol. 36, no.3, March 1998

11. R. K. Ahuja, Magnanti T.L., and J.B. Orlin. Network Flows: Theory, Algorithms and Applications. Prentice Hall, Inc., Upper Saddle River, New Jersey, 1993

12. *Sugawara, K.; Sano, M.; Yoshihara, I.; Abe, K.; Watanabe,* Foraging behaviour of multi-robot system and emergence of swarm intelligence**,** Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on, Volume: 3, 1999 Page(s): 257 -262 vol.3

13. *Fukuda, T.; Funato, D.; Sekiyama, K.; Arai, F.*,Evaluation on flexibility of swarm intelligentsystem Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on , Volume:4,1998 Page(s): 3210 -3215 vol.4

14. *Kawabata, K.; Suzuki, T.; Hayama, T.; Kobayashi, H.,* Distributed intelligent control structure for multi-legged walking robots Advanced Motion Control, 1996. AMC '96-MIE. Proceedings., 1996 4th International Workshop on , Volume:1,1996, Page(s): 231 -236 vol.1

15. A. Colorni, M. Dorigo, and V. Maniezzo, "An investigation of some properties of an ant algorithm", *Proc. Parallel Problem Solving from Nature Conference (PPSN 92)*, Brussels, Belgium, pp. 509-520, 1992.

16. M. Dorigo, V. Maniezzo, and A. Colorni, "The ant system: optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man, and Cybernetics*, Part B, vol. 26, no. 1, pp. 29-41, 1996.

17. A. Colorni, M. Dorigo, and V. Maniezzo, "An investigation of some properties of an ant algorithm", *Proc. Parallel Problem Solving from Nature Conference (PPSN 92)*, Brussels, Belgium, pp. 509-520, 1992.

18. S. Chen and K. Nahrstedh, "An Overview of Quality-of-Serive routing for the next Generation High-Speed networks" IEEE Network, pp64-79, Nov/Dec 1998

19. R. Schoonderwoerd, O.E. Holland, J. Bruten, L. Rothkrantz, "Ant-based load balancing in telecommunications networks", *HP Labs Technical Report*, HPL-96-76, May 21, 1996.

20. R. Schoonderwoerd, O.E. Holland, J.L. Bruten, "Ant-like agents for load balancing in telecommunications networks", *Proc. First ACM International Conference on Autonomous Agents.*

21. Bellman, Richard Ernest, "Dynamic programming", PrincetonUniversity Press, 1957

22. Dijkstra, E.W.: A Note on Two Problems in Connection with Graphs. In: Numeric Mathematics 1, p. 269 – 271,1959

23. G. Navarro Varela and M.C. Sinclair, "Ant colony optimisation for virtual-wavelength-path routing and wavelength allocation", *Proc. 1999 Congress on Evolutionary Computation*, Washington DC, USA, pp. 1809-1816, July 1999.

24. E. Bonabeau, F. Henaux, S. Guerin, D. Snyers, P. Kuntz, and G. Theraulaz, "Routing in telecommunications networks with "smart" ant-like agents", *Proc. Intelligent Agents for Telecommunications Applications '98*.

25. G. Di Caro and M. Dorigo, "An adaptive multi-agent routing algorithm inspired by ants behavior", *Proc. PART98 - Fifth Annual Australasian Conference on Parallel and Real-Time Systems*, Adelaide, Australia, September 28-29, 1998.

26. G. Di Caro and M. Dorigo, "Ant colonies for adaptive routing in packet-switched communications networks", *Proc. PPSN V - Fifth International Conference on Parallel Problem Solving from Nature*, Amsterdam, Holland, September 27-30, 1998.

27. G. Di Caro and M. Dorigo, "Extending AntNet for best effort quality-of-service routing", *Proc. ANTS'98 - First International Workshop on Ant Colony Optimization*, Brussels, Belgium, October 15-16, 1998.

28. G. Di Caro and M. Dorigo, "Two ant colony algorithms for best-effort routing in datagram networks", *Proc. 10th International Conference on Parallel and Distributed Computing and Systems*, Las Vegas, Nevada, October 28-31, 1998.

29. G. Di Caro and M. Dorigo, "AntNet: distributed stigmergetic control for communications networks", *Journal of Artificial Intelligence Research*, vol. 9, pp. 317-365, 1998.

30. G. Di Caro and M. Dorigo, "Mobile agents for adaptive routing", *Proc. 31st Hawaii International Conference on System Sciences*, IEEE Computer Society Press, Los Alamitos, CA, pp. 74-83, 1998.

31. T. White, "Swarm intelligence and problem solving in telecommunications", *Canadian Artificial Intelligence Magazine*, Spring, 1997.

32. T. White, "Routing with swarm intelligence", Technical Report SCE-97-15, Systems and Computer Engineering Department, Carleton University, September, 1997.

33. T. White and B. Pagurek, "Towards multi-swarm problem solving in networks", *Proc. Third International Conference on Multi-Agent Systems (ICMAS '98)*, July, 1998, pp 333-340.

34. T. White, B. Pagurek, and F. Oppacher, "ASGA: improving the ant system by integration with genetic algorithms", *Proc. Third Genetic Programming Conference*, July, 1998, pp. 610-617.

35. T. White, B. Pagurek, F. Oppacher, "Ant search with genetic algorithms: application to path finding in networks", *Combinatorial Optimization '98*, Brussels, Belgium, April 15th-17th, 1998.