

Research Article

Multiple Interface Parallel Approach of Bioinspired Routing Protocol for Mobile Ad Hoc Networks

L. J. García Villalba,¹ D. Rupérez Cañas,¹ A. L. Sandoval Orozco,¹ and T.-H. Kim²

¹ *Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA), Facultad de Informática, Universidad Complutense de Madrid (UCM), Despacho 431, Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain*

² *School of Information Science, GVSA and UTAS, 20 Virginia Court, Sandy Bay, Hobart, TAS 7001, Australia*

Correspondence should be addressed to L. J. García Villalba, javiergv@fdi.ucm.es

Received 30 September 2012; Accepted 6 October 2012

Academic Editor: Sabah Mohammed

Copyright © 2012 L. J. García Villalba et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The design of routing protocols for mobile ad hoc networks (MANETs) is a complex task given the dynamic nature of such networks. Particular types of routing protocols are known as bioinspired. This work presents a parallelization of AntOR-DNR, a bioinspired routing protocol for mobile ad hoc networks based on the Ant Colony Optimization (ACO) algorithm. This new protocol, called PAntOR-MI, uses, as well as PAntOR, the thread programming based on shared memory. This new parallelization is applied in route discovery phases, route local repair process, and link failure notification. The simulation results indicate that PAntOR and PAntOR-MI improve performances of AntOR, whilst it is also noticed that PAntOR-MI is the most suitable for highly dynamic environments.

1. Introduction

A mobile ad hoc network (MANET) [1] is a collection of mobile devices, which form a network of communication without predefined infrastructure. This fact determines the design of routing protocols for this type of network to suppose an arduous task. Particular types of routing protocols are called bioinspired [2], taking into account the behavior of some animals (insects, etc.) to obtain their food. Related to these, the algorithms based on Ant Colony Optimization (ACO) [3], are particularly relevant. A representative protocol of so-called bioinspired is AntOR [4], adaptive and multihop routing protocol based on AntHocNet [5, 6]. The specification of this protocol includes two versions: disjoint-link routes (AntOR-DLR) and disjoint-node routes (AntOR-DNR). A parallel approximation of AntOR-DNR is PAntOR [7, 8] improving performances of AntOR-DNR through thread programming based on shared memory in the phases of routing information setup, route local repair, and link failure notification. This paper presents a new parallel approximation of AntOR-DNR, called PAntOR-Multiple

Interface (PAntOR-MI) which, as its name suggests, differs essentially from PAntOR in the use of multiple interfaces. This paper consists of 6 sections, with this introduction being the first of them. The rest of the paper is structured as follows: Section 2 briefly discusses the most representative work on parallel techniques for bioinspired protocols based on the behavior of ants. Section 3 briefly comments on AntOR-DNR and PAntOR, also showing a comparison between both. Section 4 presents PAntOR-MI, with emphasis on differences from its predecessor, PAntOR. Section 5 shows a comparative study between AntOR-DNR, PAntOR, and PAntOR-MI. Finally, the conclusions are exposed in Section 6.

2. Related Works

In this section we present the most representative parallelization techniques that make ACO more efficient. First of which, introduced by [9], explains a method that can solve difficult combinatorial optimization problems. Stützle [10] applies an approximation master/slave to parallelize several different

search methods of ACO solutions. Reference [11] shows a hybrid system of parallelization which consists of evaluating the performance of communication multithreading Message Passing Interface (MPI). This approach applies MPI between nodes and multithreading within nodes. Finally, [7, 8] presents PAntOR, an approximation parallel based on programming by threads, which constitutes the starting point of the present paper. The main idea of this protocol is to replace each broadcast message by a message managed by a thread that is addressed to each one-hop neighbour, that is, launches a thread by each node in the neighbour table. This is done in protocol phases: routing information setup, route local repair, and link failure notification.

3. AntOR versus PAntOR

AntOR [4] has the following characteristics which are different from AntHocNet [5, 6]:

- (i) Disjoint-link and Disjoint-node protocol,
- (ii) separation between the pheromone values in the diffusion process, and
- (iii) use distance metric in route proactive exploration.

PAntOR [7, 8] paralyze Disjoint-node version (AntOR-DNR) that consists of nodes from routes which are not shared. We have chosen this version because routes are more difficult to get and maintain. To understand how P-AntOR works, it is necessary to use three concepts well.

- (a) *Process*. It is a program running which is managed by the Operating System.
- (b) *Thread*. It consists of the basic unit of execution, so any program that executes has at least a thread.
- (c) *POSIX Thread*. It is a Standard based in thread API for C/C++.

We use POSIX Thread because it allows a new concurrent process flow to expand, which is the most efficient multicore systems, generating a flow of processes that can be scheduled to run on another processor, thus speed through distributed processing is achieved. Programming with threads carries less overhead than expanding a new process, because the system does not initialize a new environment and virtual memory space for that process.

This version tries to replace each broadcast message by a message managed by a thread that is addressed to each one-hop neighbour.

This parallel technique, which launches a thread by each node in the neighbour table, is used in the following phases of AntOR.

3.1. Routing Information Setup. Figure 1 shows a flow chart representing the parallelism in the route discovery process. When a source node is ready to send data to the destination node, it activates the route discovery process. This process is parallelized using threads, so that it launches a Reactive Forward Ant (RFA) reactive to the one-hop neighbours

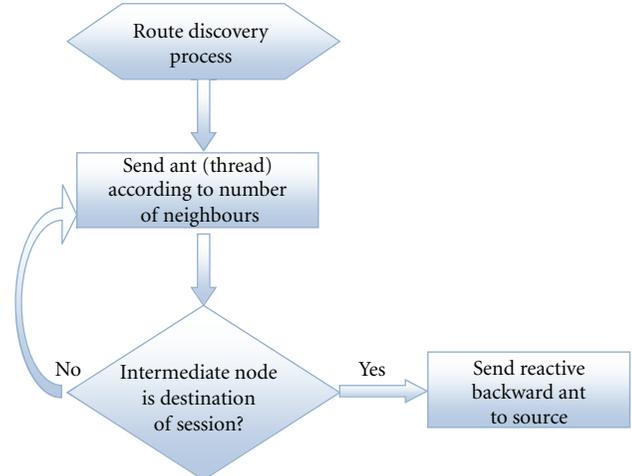


FIGURE 1: Parallelization of route discovery phase.

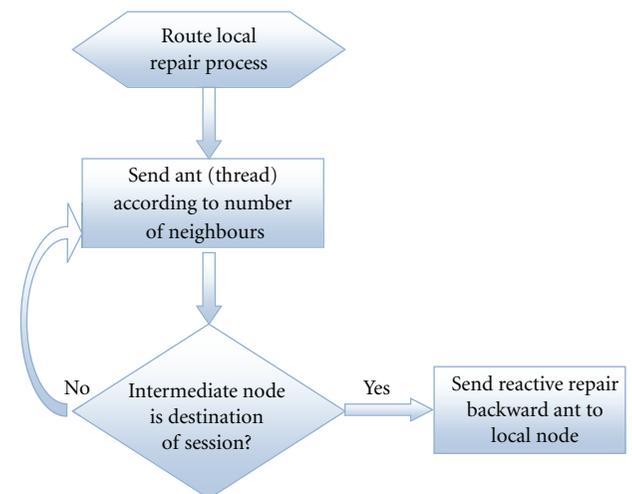


FIGURE 2: Parallelization of local route repair phase.

through an independent thread. When an intermediate node receives this ant repeats the process, but whether it is a destination node, this node sends its corresponding Reactive Backward Ant (RBA).

3.2. Route Local Repair. The operation is similar to route discovery, unless it is done locally, as shown in Figure 2.

3.3. Link Failure Notification. This process updates the routing table to link failures. It is required to be taken promptly because of its importance. The nodes send ants through independent threads until an intermediate node has some alternative route to the destination after updating the routing table, as shown Figure 3.

3.4. Results. Next we show a comparison between PAntOR and AntOR. To this end, the metrics used in the simulations were as follows.

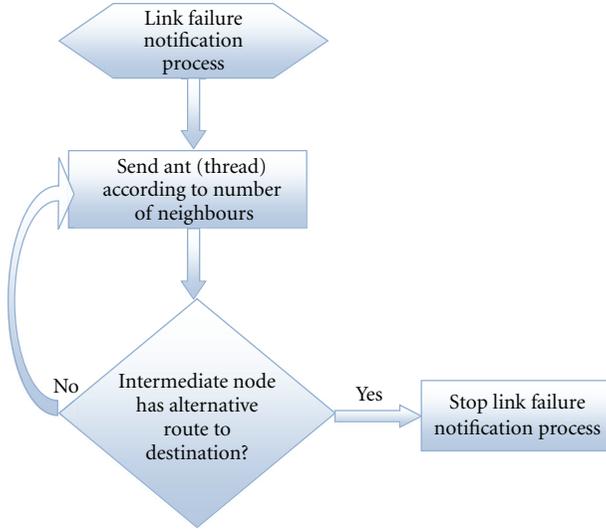


FIGURE 3: Parallelization of link failure notification phase.

- (i) *Throughput*: consists of volume of work or information flowing through a system. It is calculated by dividing the total number of bits delivered to the destination by the packet delivery time.
- (ii) *Delivered Data Packet Ratio*: relationship between number of packets sent and the number of packets delivered successfully.
- (iii) *Overhead in number of packets*: relationship between the total numbers of transmitted control packets by the nodes of network and the number of delivered data packets to their destinations.

Experiments with the Network Simulator NS-3 have been realized. Simulations parameters are as follows: we have used 100 nodes configured according to the Standard IEEE 802.11b, moving in a random scenario of dimensions $1200\text{ m} \times 1200\text{ m}$ according to the mobility pattern *Random WayPoint* (RWP). The application of data traffic is *Constant Bit Rate* (CBR) with a rate of packet sending of 2048 bps (4 packets of 64 bytes per second). We apply 5 random data sessions, where mobility is variable from 0 m/s up to a maximum of 10 m/s. Pause time is kept constant to a value of 30 s. Total simulation time is 120 s. In Figure 4 we can observe how the Throughput is better at PAntOR at AntOR.

In Figure 5 see how overhead in number of packets is better in the parallel version than in the original, because the creation of routes is faster.

Figure 6 behaves similar to the representation in Figure 4, but using another scale. It notes that the performance of the packet delivery remains reasonably good even at high speeds.

4. PAntOR-Multiinterface

This variant of PAntOR consists of having more than one interface, and to parallelize the sending of broadcast messages by interface through threads. Each interface is managed by a thread. The main idea of this parallelization

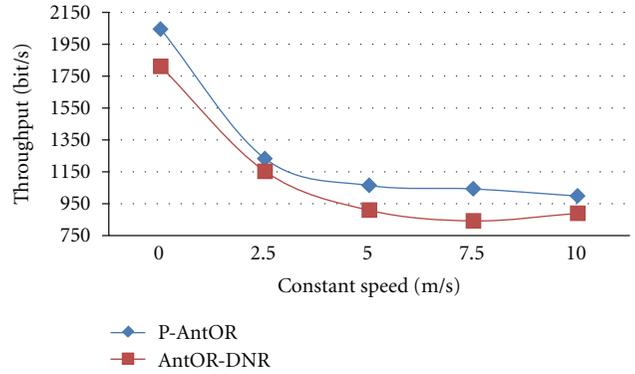


FIGURE 4: Constant speed against Throughput.

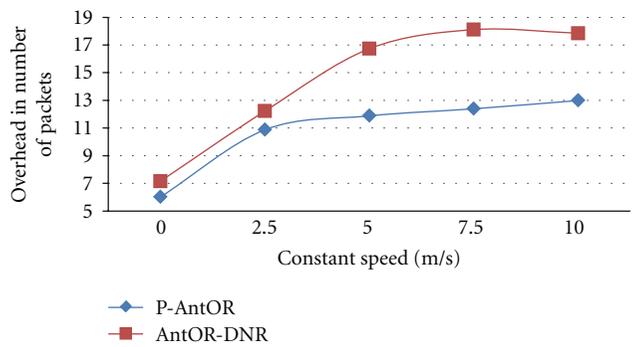


FIGURE 5: Constant speed against overhead in number of packets.

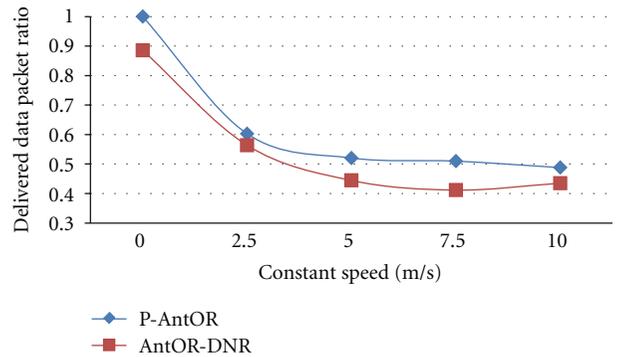


FIGURE 6: Constant speed against Delivered Data Packet Ratio.

is to be applied to systems with several multiinterfaces, which launches an ant broadcast mode in an independent thread for each interface that provides the node. The main difference with PAntOR is that PAntOR-MI uses more than one interface, parallelizing each interface by means of a thread.

To understand this variant we provide Algorithm 1.

It can be seen that while running the routing information setup, a reactive message is launched in a broadcast way for each interface that have the node, and such an interface via a thread is managed.

```

while Routing Information Setup do
  for Cont = 1 to Max.Interfaces do
    Send Broadcast Message by Thread (Cont);
  end
end
end

```

ALGORITHM 1: Parallelization core in PAntOR-MI.

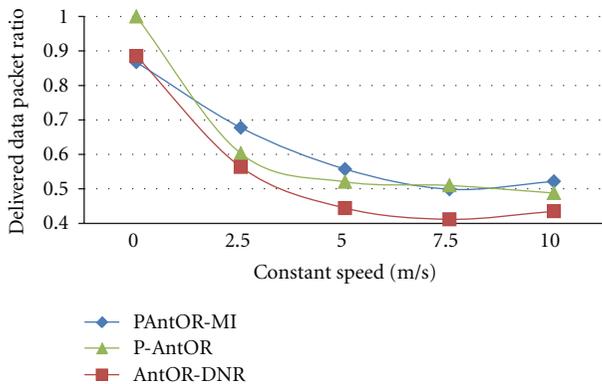


FIGURE 7: Constant Speed against Delivered Data Packet Ratio.

5. PAntOR-MI versus AntOR and PAntOR

The performance metric analyzed in this comparison is Delivered Data Packet Ratio, which consists of the relationship between the number of packets sent and the number of packets delivered correctly. To perform this comparison Network Simulator NS-3 has been used with the following parameters: a random scenario with dimensions of 1200 m \times 1200 m has been designed, where 100 nodes configured according to the Standard IEEE 802.11b; they move according to the mobility pattern *Random WayPoint* (RWP). The application of data traffic is based on *Constant Bit Rate* (CBR) with a packet sending rate of 2048 bps (4 packets of 64 bytes per second). We apply 5 random data sessions, where mobility is variable from 0 m/s up to a maximum of 10 m/s. Pause time is kept constant to a value of 30 s. Total simulation time is 120 s. In this comparison PAntOR-MI use nodes with two interfaces.

In Figure 7 we appreciate how the delivered packet ratio is better in these two parallel versions than in AntOR. Also we see how PAntOR-MI improves to AntOR and PAntOR. Also a greater tolerance to the mobility of the nodes is observed, behaving better in the more dynamic scenarios.

6. Conclusions

This work has presented a new bioinspired routing protocol for mobile ad hoc networks obtained thanks to new parallelization techniques of a base protocol called AntOR which has two versions, the so-called Disjoint-link (AntOR-DLR) and Disjoint-node (AntOR-DNR). The new parallel approach (PAntOR-MI) used the disjoint-node version of AntOR (AntOR-DNR) as the main protocol, as well as the

existing (PAntOR). The parallelization technique employed is a large-grained approach, in which a multicore machine in a shared memory system has been used. The novelty of PAntOR-MI is that we have more than one interface, and we parallelize the sending of broadcast messages by interface using threads. Each interface is managed by a thread. The obtained simulation results indicate that PAntOR-MI improves performances of PAntOR, with further observations showing that this improvement is most evident in most dynamic environments.

Acknowledgment

This work was supported by the Agencia Española de Cooperación Internacional para el Desarrollo (AECID, Spain) through Acción Integrada MAEC-AECID Mediterráneo A1/037528/11.

References

- [1] M. Abolhasan, T. Wysocki, and E. Dutkiewicz, "A review of routing protocols for mobile Ad Hoc networks," *Ad Hoc Networks*, vol. 2, no. 1, pp. 1–22, 2004.
- [2] V. Jha, K. Khetarpal, and M. Sharma, "A survey of nature inspired routing algorithms for MANETs," in *Proceedings of the 3rd International Conference on Electronics Computer Technology (ICECT'11)*, pp. 16–24, Kanyakumari, India, April 2011.
- [3] M. Dorigo and T. Stützle, *Ant Colony Optimization*, The MIT Press, Bradford Company Scituate, Mass, USA, 2004.
- [4] L. J. G. Villalba, D. R. Cañas, and A. L. S. Orozco, "Bio-inspired routing protocol for mobile Ad Hoc networks," *IET Communications*, vol. 4, no. 18, pp. 2187–2195, 2010.
- [5] F. Ducatelle, *Adaptive routing in Ad Hoc wireless multi-hop networks [Ph.D. thesis]*, Università della Svizzera Italiana, Istituto Dalle Molle di Studi Sull'Intelligenza Artificiale, 2007.
- [6] G. Di Caro, F. Ducatelle, and L. M. Gambardella, "AntHocNet: an adaptive nature-inspired algorithm for routing in mobile Ad Hoc networks," *European Transactions on Telecommunications*, vol. 16, no. 5, pp. 443–455, 2005.
- [7] L. J. García Villalba, D. Rupérez Cañas, and A. L. Sandoval Orozco, "Parallel approach of a bioinspired routing protocol for MANETs," *International Journal of Ad Hoc and Ubiquitous Computing*. In press.
- [8] D. Rupérez Cañas, A. L. Sandoval Orozco, L. J. García Villalba, and T.-H. Kim, "Comparing AntOR-disjoint node routing protocol with its parallel extension," in *Proceedings of the International Conference on Multimedia, Computer Graphics and Broadcasting (MulGraB'11)*, pp. 305–309, 2011.
- [9] B. Bullnheimer, G. Kostis, and Strauss, "Parallelization strategies for the ant systems," in *High Performance Algorithms and*

Software in NonLinear Optimization Series: Applied Optimization, vol. 24, 1998.

- [10] T. Stützle, "Parallelization strategies for ant colony optimization," in *Proceedings of the Parallel Problem Solving from Nature*, vol. 1498, 1998.
- [11] R. Thakur and W. Gropp, "Test suite for evaluating performance of multithreaded MPI communication," *Parallel Computing*, vol. 35, no. 12, pp. 608–617, 2009.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

