

CONTEXT-AWARE INTER-NETWORKING FOR WIRELESS NETWORKS

Franck Legendre, Marcelo Dias de Amorim, and Serge Fdida*

LIP6/CNRS – University of Paris VI

8, rue du Capitaine Scott – 75015 – Paris, France

{legendre,amorim,sf}@rp.lip6.fr

Abstract Key issues of wireless ad hoc networks are auto-configuration and flexibility. Due to mobility, wireless networks are subject to frequent splits and merges. In many situations, overlapping networks must combine to form a new single network requiring a partial or total reconfiguration. In this context, existing proposals assume that merging networks use similar routing schemes. Nevertheless, there is no guarantee that different networks implement the same routing protocols. In such a scenario, routing protocols must adapt to the changing nature of the network characteristics. The wide spectrum under which ad hoc wireless networks are used lead to the design of multiple routing protocols (*e.g.*, proactive, reactive and adaptive). Merging of wireless networks using distinct routing protocol is thus unavoidable and leverages particular difficult problems. We propose in this paper a flexible solution that enables heterogeneous networks to efficiently cope with merging. We designed a routing translator daemon based on the neighborhood context enabling neighbor nodes using distinct protocols to interoperate. The neighborhood context is obtained from the Neighborhood Routing Protocol Discovery Protocol. Several scenarios under different assumptions show that AODV, DSR and OLSR can efficiently interoperate. We give insights on how routing must adapt and evolve in order to cope with merging.

1. Introduction

Wireless technologies show a convergence between standards of the telecommunication industry and standards of the computer industry in the name of IP. Nevertheless, no unique standard has been defined for the routing algorithm in mobile ad hoc networks. Different classes of routing

*This work has been supported by CNRS and Euronetlab.

protocols are under standardization. Examples are the Ad hoc On-Demand Distance Vector (AODV [Perkins et al., 2003]), the Dynamic Source Routing (DSR [D.B. Johnson and Hu, 2003]), the Optimized Link State Routing (OLSR [Clausen and Jacquet, 2003]), and the Topology Dissemination Based on Reverse-Path Forwarding (TBRPF [R. Ogier and Lewis, 2004]).

Routing protocols can be classified in three categories: reactive, proactive, and adaptive. In reactive routing protocols (AODV, DSR), routes are discovered on demand, which reduces overhead by sending routing information only when needed. Proactive protocols (OLSR, TBRPF) exchange routing information in a background process. While on-demand routing protocols suit sparse communication patterns and small networks, proactive protocols have immediate route availability that is more adapted to dense communication patterns, larger networks, and applications requiring QoS guarantees. Hybrid or adaptive routing protocols like the Zone Routing Protocol (ZRP) [Haas, 1997] have been proposed combining advantages of both on-demand and proactive approaches. Even though hybrid protocols are promising solutions, they are not candidates for standardization. We can expect for a single adaptive protocol that will automatically and optimally fit all network situations.

Meanwhile, the restricted number of standardized routing protocols will enable equipments with sufficient capabilities to implement several protocols and use the appropriate one when joining a network. It will be possible for a group to switch from one routing protocol to another in order to efficiently adapt to new network conditions. But it is likely that equipment vendors will either choose a standardized protocol or develop proprietary protocols based on recent developments on adaptive routing. In both cases, these observations legitimate the urgent need for a general routing framework which allows interaction and interoperation between heterogeneous ad hoc routing protocols. This issue has already been tackled in [Lu and Bhargava, 2001], where a hierarchical architecture is setup between merging networks with group agents dedicated to inter-network communication.

We propose a decentralized approach where a new protocol, the Neighborhood Routing Protocol Discovery Protocol (NRPDP) enables nodes to discover the routing capabilities (*i.e.*, supported routing protocols) of their neighbors. Based on these acquired information about the neighborhood, heterogeneous nodes are able to translate routing control packets of their respective routing protocols so that it can be understood by all their neighbors. This translation is realized on the fly by the Routing Translator Daemon (RTD) using a set of defined rules dependent on

the context. Note that NRPDP and RTD are not intended to replace existing approaches, but to allow heterogeneous nodes to interoperate.

We will show the features of our approach through two scenarios, one where AODV interoperates with DSR and another where AODV interoperates with OLSR.

The paper is organized as follows. Section 2 introduces our network model. In Section 3, we present the advantages and requirements for a *smooth* merging. Section 4 formalizes our framework and introduces the design and mechanisms required. In Section 5, we illustrate our solution with two networks using distinct routing schemes, (*i.e.*, AODV, DSR, and OLSR). Finally, Section 6 discusses future research investigation and concludes the paper.

2. Network model: the cell approach

In this paper, we define the concept of network cells. A cell, C , is the spatial region spanned by a set of nodes, N_C , willing to participate in some collaborate activity (*e.g.*, emergency disaster relief, battlefield troops, conference attendees) or with similar interest in spatial movement (*e.g.*, public transport users). The evolution of a cell is progressive, *i.e.* new arriving nodes acquire the address allocation scheme and routing scheme from an existing cell member. The address assignment may be of any kind (conflict-free, conflict-detection, or best-effort). For example, nodes can acquire an address from their neighbors or randomly generate their addresses and verify their uniqueness using a duplicate address detection (DAD) [Nesargi and Prakash, 2002] mechanism. For routing, we assume that due to the diversity of situations, nodes implement a set of existing protocols. However, since equipments have disparate capabilities (*i.e.*, processing, memory), they do not always implement the same set of routing protocols but it is likely that they will implement the most adopted ones. We believe that a restricted list of protocols will be supported by vendor equipments, some respecting standards and others proprietary implementations. When cells C_1 and C_2 are about to merge, the overlapping region $\mathcal{S}_{C_1, C_2} = C_1 \cap C_2$ is called a *scar-zone* and is delimited by a *scar-zone membrane* (cf. Fig. 1). Nodes located in the scar-zone are called *scar-zone* nodes, S_C , while nodes outside the scar-zone, I_C , are named *interior* nodes. Depending on the respective mobility of the two cells, the scar-zone can evolve between a minimal overlapping where the two cells are interconnected via one radio link to a complete overlapping where the spatial extent of one cell is included in the other.

When C_1 and C_2 merge, two cases may occur:

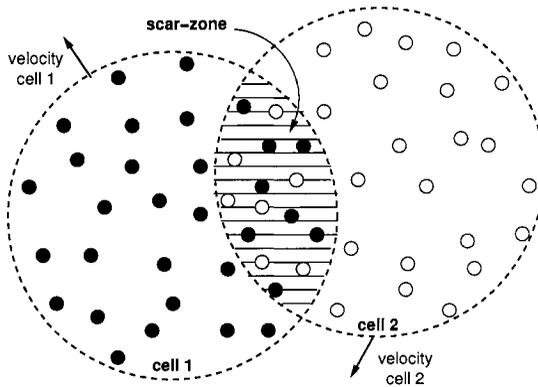


Figure 1. Network cell model.

- **Merging cells are homogeneous:** in this case, both cells use the same routing protocol. Existing approaches address only this case. We believe this is an ideal case that does not hold all the time.

- **Merging cells are heterogeneous:** heterogeneity has been widely studied in ad hoc environments, but restricted to PHY and MAC layers. For example, several frameworks have been proposed to enable the convergence of different radio access technologies known as Software Definable Radio (SDR) [JSAC, 1999]. With IP as the universal standard, heterogeneity at upper layer levels has been superficially considered. With the various routing protocols available and under standardization, it is clear that protocol heterogeneity must be tackled in a similar manner to what is done in wired networks.

3. Heterogeneous merging: a smooth approach

Heterogeneous merging can be addressed in two different ways. The first approach – inspired from proposals related to addressing schemes that cope with merging – would be to completely reconfigure both or only one of the cells. The second approach consists in enabling interaction between heterogeneous protocols, which does not require a global reconfiguration. In the following, we discuss the advantages and drawbacks of the first approach and from conclusions we propose our scalable framework.

3.1 The case of heterogeneous cell interoperability

In the first approach, nodes of the two cells agree on a common routing protocol. This approach is similar to what is done in addressing schemes where one of the two cells completely reconfigures its addresses in order to have a coherent and homogeneous addressing space. Even if this approach is the simplest one, we demonstrate that under some network conditions it is not achievable and raises scalability problems.

First, the reconfiguration of one of the networks requires all nodes to implement and swap to the required protocol. Unless an active approach is available, this case is hardly achievable.

Second, considering that all nodes can converge to the same routing protocol, all on-going routing control traffic, communications, and states held by each node (*i.e.*, routing tables, forwarding tables, etc.) would be lost. For example, changing from OLSR to a reactive protocol would invalidate proactive routing tables.

Third, depending on the network conditions and more specifically on the cell mobility, swapping to another routing protocol may be sub-optimal. For example, if several successive merges occur in a short period of time it would require multiple reconfigurations. The system would become instable and no more routing would be possible. Another example is if merging is only transient. In this case again, reconfiguration can be sub-optimal in some cases.

As described in the introduction, the different categories of routing protocols are adapted to particular network conditions and must be chosen accordingly. If reconfiguration is required, a lap of time is needed to discover the new network situation when merging occurs. During this short period of time it might not be optimal to swap to an inappropriate routing protocol.

For all these reasons, we plead for a smooth merging protocol. This protocol can be complementary to a possible posterior reconfiguration of two merging cells.

3.2 Addressing heterogeneous cell interoperability

We propose a completely different approach where nodes do not reconfigure themselves when merging. Instead, we benefit from the fact that routing protocols can be of the same family, which may differ in implementation but not in their general philosophy. We also take advantage of existing routing states already stored by nodes, as in the case of link-state (LS) routing protocols. In order to do this we must, depend-

ing on the context and the routing protocols involved, either translate routing control packets from one protocol to another or allow the interaction of nodes relevant to routing. While the former proposal is designed for protocols of the same family, the latter is suited for protocols that are completely different in their conception and do not enable a direct translation.

With this approach, we want to conceive a new protocol with the following design goals:

- Routing control packets (link-state updates, route requests, route replies) must not be modified to suit our needs.
- Inter-cell communication should avoid dedicated and centralized translator cluster-head or group leader.
- Routing loops and unnecessary recursive translations must be avoided.

4. Design and mechanisms

In order to achieve inter-cell operability, scar-zone nodes – nodes located in the overlapping region – must define their neighborhood environment context (*i.e.*, routing capabilities of neighbors). Based on this context, specific translation must be performed or, if translation is not possible, nodes must execute appropriate neighborhood interaction.

Proactive and reactive protocols can be implemented as routing daemons (executed in user space), like wired routing protocols such as RIP and OSPF. We define then a general ad hoc routing layer that enables the flexible use of several routing protocols. This daemon intercepts the I/O of routing control packets and process these packets (requests, replies, updates) based on the status of the node:

- The node operates normally if it is in the interior zone.
- The node executes a context-dependent algorithm (neighborhood's context) if it is in the scar-zone.

We propose to associate this new daemon with the Neighborhood Routing Protocol Discovery Protocol (NRPDP). This protocol enables neighbor nodes to discover their respective routing capabilities (*i.e.*, the implemented routing protocols).

4.1 The NRPDP Protocol

Nodes of the scar-zone must be aware that they are possible candidates for network routing protocol translation or interaction. In order

to achieve this, nodes must detect that a merging is about to occur and then determine if they are scar-zone nodes or not. Previous approaches proposed the use of Network IDentifiers (NID) to detect merging between distinct cells. A NID is a random number generally broadcasted over the entire cell or piggybacked during the address allocation process. NRPDP sends periodic Hello messages containing a cell's NID. This procedure enables the detection of merging and defines a scar-zone if merging cells use the same or a different routing scheme.

Nodes located in the scar-zone exchange their routing capabilities (*i.e.*, supported routing protocols) and the current protocol in use – the *mother* routing protocol. For example, in Figure 3 node *X* sends {*AODV*, DSR} to *Y*, pointing out the routing protocols it supports and the current routing protocol in use in its cell, indicated by {*Protocol*}. Similarly, node *Y* sends its NRPDP Hello packet to *X*, {*DSR*, AODV}.

The NRPDP daemon interacts with the routing translator daemon by communicating the neighborhood context (cf. Fig. 2).

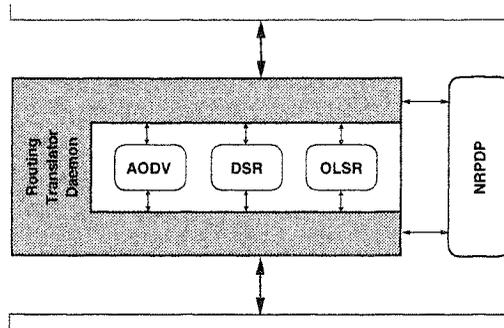


Figure 2. NRPDP and routing translator.

4.2 The Routing Translator Daemon

Once a node is aware of its neighbors' capabilities, thanks to NRPDP, the routing translator daemon can take appropriate actions relative to routing. Depending on the context, a translator is either off or on. If the node is surrounded by a homogeneous neighborhood, the translator is off, otherwise the translator is on. In the latter case, depending on the context, the translator may take one of the following three actions: (a) it forwards the packet as it is to homogeneous neighbors, (b) it translates the packet for heterogeneous neighbors using a different routing protocol of the same category, otherwise (c) it interacts with an heterogeneous

neighbor node to retrieve required information for the routing process. In the following section, we will explicitly detail these different contexts through examples.

Table 1. The Routing Translator Daemon table.

RREQ ID	Dest. IP Address	Dest. Seq. Num.	Orig. IP Addr.	Orig. Seq. Num.	Hop count	DSR request id.
m	B	0	A	seq_A	1	$hash(seq_A)$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots

5. Application: AODV \leftrightarrow (DSR,OLSR)

In this section, we illustrate how merging cells using heterogeneous routing protocols can interoperate in a flexible and efficient manner. The vocabulary and acronyms used in the following refer specifically to each protocol. We do not discuss in this paper the route maintenance inherent to reactive protocols and assume that the mobility relative to the establishment of a path does not incur any link breakage. We only consider how paths are established and do not consider how packets are forwarded on these paths.

OLSR and TBRPF are two experimental RFCs proposed for standardization by the IETF MANET Working Group. We have chosen OLSR for our case study. OLSR is a LS routing protocol where nodes periodically exchange topology information with other nodes in the network. After a time needed for convergence, each node has an optimized view of the current topology. OLSR differs from pure LS in that it optimizes the flooding of updates and reduces the size of LS packets.

AODV (experimental RFC) and DSR (Internet-Draft) are the two most referenced reactive protocols. In reactive protocols, paths from a source to a destination are discovered on-demand with query packets for a destination flooded across the entire network. In AODV, intermediate nodes learn the path to the source and enter the route in their forwarding table. When the destination is reached, it uses the path traced by query packets as reverse route toward the source. DSR distinguishes from AODV by the use of source-routing, *i.e.* intermediate nodes concatenate their IDs in the route query packet header. The destination then retrieves the path and uses source-routing to respond to

the source. Compared to AODV, intermediate nodes in DSR do not keep any state related to a given path.

5.1 AODV ↔ DSR

Consider the scenario shown in Fig. 3, where cell 1, C_1 , runs AODV and cell 2, C_2 , runs DSR. Consider also two nodes, A and B , with $A \in I_{C_1}$, $B \in I_{C_2}$ (i.e., $\{A, B\} \not\subset \text{scar-zone}$). Here, we study how paths can be established in these cells in both ways, from A to B and vice-versa. Note that, in this case, protocols belong to the same family which clearly simplifies the interoperability – only a simple translation is required.

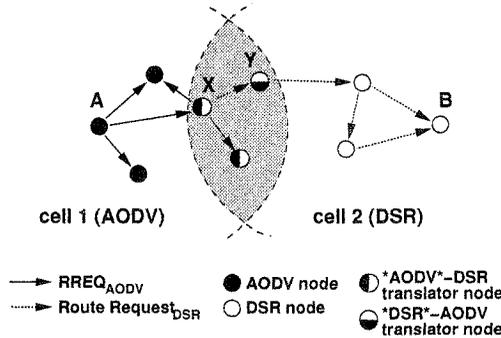


Figure 3. Translation AODV ↔ DSR.

For the establishment of a path from A to B , $A \rightarrow B$, A floods a RREQ (Route Request). When a node in the scar-zone receives the request, here X , it translates this AODV RREQ into a DSR Route Request. X associates the AODV's RREQ with an entry in the table maintained by its RTD (Tab. 1). AODV's RREQ sequence number, seq_A , is associated with the newly emitted DSR Route Request sequence number result of $\text{hash}(seq_A)$, where $\text{hash}(\cdot)$ is a hash function shared by all nodes. This entry indicates that a translation will be required at the reception of the DSR Route Reply.

When the DSR Route Reply is received, an AODV RREP is sent using the reverse route entry established by the initial RREQ. In order to avoid loops, the translation requires the use of the same sequence number and the precise translation of the number of hops. We cannot use the same sequence number since the sequence number fields of these protocols have different sizes. This is why we use a hash function common to all nodes. For the hop count, in the AODV header the number of hops

is represented by a hop count field, while in DSR counting the number of concatenated IDs gives the hop number. This translation is done by referring to the corresponding entry in the RTD table. As well, the correct association between routing control packets prevents recursive translations. For example, X receives an AODV RREQ from Y in reply to its original DSR Route Request sent to Y . By comparing the received sequence number with the one contained in its RTD table, X detects this route request is generated in response to its original request.

Similarly, for the path $B \rightarrow A$, B floods a DSR Route Request that is translated into an AODV RREQ by scar-zone nodes. These scar-zone nodes will receive RREPs that will be translated back to DSR Route Replies.

5.2 AODV \leftrightarrow OLSR

Here, we consider that C_1 runs AODV while C_2 runs OLSR. This case is more complex to treat since these protocols belong to different families. More elaborated node interactions are then required.

For the path from A to B , $A \rightarrow B$, node A floods a RREQ in its cell. Nodes of the scar-zone cannot perform any direct mapping of an AODV RREQ into an OLSR control packet. In Figure 4, node X forwards the RREQ to its neighbors. Node Y belonging to the OLSR cell will receive this message. Y 's routing translator daemon catches this packet, from the context transmitted by its NRPDP, determines that it is in the scar-zone, and looks up its OLSR routing table for the destination. With the retrieved information, the translator daemon replies with a RREP including the number of hops obtained from the OLSR routing table. Node A will receive several RREPs and choose the shortest path to start the communication. From now on, A has a path to join B but B does not have a reverse path to A . There are several solutions to this problem. As soon as the RREQ is received, X sends a TC – message (Topology Declaration, *i.e.* advertisement of LS) announcing that A can be reached through X .

Now, if we consider the case where B wants to launch a communication with node A , the process is more complex since A 's address will not be in the routing table of the OLSR nodes. If we consider that the AODV cell uses the same address prefix, nodes of the scar-node can send a Host/Network Association (HNA) message indicating they are gateways for the announced prefix. Since this assumption may not hold in ad hoc networks where addresses can be randomly chosen, another solution is required. We propose that the translator routing daemon of OLSR nodes in S_{C_1, C_2} uses LS update packets to inform I_{C_2} nodes that

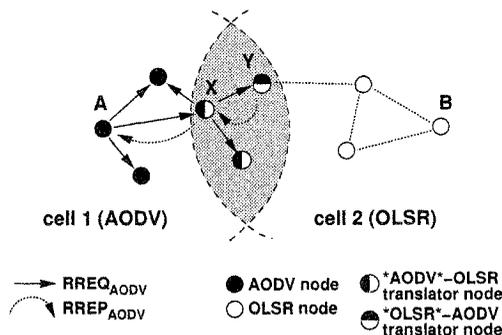


Figure 4. Translation AODV \leftrightarrow OLSR.

another network is reachable through them. Upon the reception of this information, the routing daemon of OLSR nodes adds a default route in their OLSR routing table. This entry will be used when forwarding a packet to a node in the other cell. When an OLSR scar-zone node receives a packet for node *A*, it passes the packet to an AODV node that initiates a $RREQ_{AODV}$ and waits to receive the $RREP_{AODV}$ before forwarding the packet.

Note that there is no need in these examples for OLSR nodes to implement the AODV protocol and inversely. They only require to *understand* routing control packets of other protocols and take appropriate action defined by a set of rules depending on the context.

6. Conclusion

The case of merging networks is an area with many open questions. Little attention has been given by the ad hoc routing community on issues related to heterogeneity involving layers above PHY and MAC. Our proposal is one of the first attempts to consider routing protocol heterogeneity. This paper proposes a novel framework for merging heterogeneous cells. Contrary to previous proposals, we consider the case where cells use distinct routing schemes. One of our main goals is to enable a smooth and flexible cell interoperability whatever routing protocols are involved. We proposed a new routing approach, *i.e.* a context-aware routing based on neighborhood capabilities. We designed NRPDP, a new protocol for discovering neighbor routing capabilities, and an associated routing daemon (the Routing Translator Daemon).

The advantage of our solution is to enable flexible interoperation between merging cells. We showed that three of the main candidates for

standardization can easily interoperate without modifying their specifications. With our scheme, we avoid the time needed for a complete reconfiguration. Even if reconfiguration is undertaken, we avoid the stoppage before reconfiguration is achieved. As well, envisioning the benefits of our approach to adaptive routing protocols, our solution can be viewed as complementary. Since adaptive protocols borrow concepts from proactive and reactive routing protocols, the framework presented in this paper would only undergo minor adaptations. In fact, only new translation rules and protocol-specific interactions must be added to our translator daemon. Generalizing our approach to an interoperability of all routing protocols, we are considering the development of an inter-network routing protocol that will be used as a common routing protocol between wired and wireless heterogeneous networks. This work is also motivated by the fact that, similarly to wired networks, bandwidth (*i.e.*, radio) is a scarce resource as well as energy that cannot be managed without introducing some policy and negotiation between merging networks.

As a conclusion, if the current trend followed by the IETF's MANET Working Group is verified, we believe several ad hoc routing protocols will be standardized. These protocols would benefit from our approach since they would become *compliant* instead of concurrent.

References

- [Clausen and Jacquet, 2003] Clausen, T. and Jacquet, P. (2003). Optimized Link State Routing Protocol OLSR. RFC. RFC-3626.
- [D.B. Johnson and Hu, 2003] D.B. Johnson, D.A. Maltz and Hu, Y-C. (2003). The Dynamic Source Routing protocol for Mobile ad hoc networks (DSR). Internet-Draft. draft-ietf-manet-dsr-09.txt - Work-in-progress.
- [Haas, 1997] Haas, Z. (1997). A new routing protocol for the reconfigurable wireless networks. In *Proc. of the IEEE International Conference on Universal Personal Communications*.
- [JSAC, 1999] JSAC, IEEE (1999). Special issue on software radios. In *Mobile Computing*, volume 17-4. IEEE.
- [Lu and Bhargava, 2001] Lu, Y. and Bhargava, B. (2001). Achieving flexibility and scalability: A new architecture for wireless network. In *Proc. of International Conference on Internet Computing (IC'2001)*, pages 1105-1111, Las Vegas.
- [Nesargi and Prakash, 2002] Nesargi, S. and Prakash, R. (2002). Manetconf: Configuration of hosts in a mobile ad hoc network. In *Proc. of Infocom'02*, New York.
- [Perkins et al., 2003] Perkins, C. E., Belding-Royer, E. M., and Das, S. (2003). *Ad hoc On-Demand Distance Vector AODV Routing*. IETF RFC 3661.
- [R. Ogier and Lewis, 2004] R. Ogier, F. Templin and Lewis, M. (2004). Topology Dissemination Based on Reverse-Path Forwarding (TBRPF). RFC. RFC-3684.