

Providing interoperability between IEEE 802.11 and Bluetooth protocols for Home Area Networks

Pietro Manzoni *, Juan Carlos Cano

Department of Computer Engineering, Polytechnic University of Valencia, Camino de Vera s/n, 46071 Valencia, Spain

Abstract

We present a cluster-based low-complexity routing algorithm for self-organizing networks of mobile nodes called FRANCA. The proposed algorithm allows mobile nodes to autonomously create clusters to minimize the power consumption. We maintained our proposal as simple as possible also providing it with robustness and auto-recoverability properties to allow nodes reconfiguration to be simple and fast. FRANCA is implemented as two separate protocols: the intra-cluster data-dissemination protocol and the inter-cluster routing protocol. These protocols allow integrating IEEE 802.11 and Bluetooth networks providing interoperability among mobile devices. We present a sensitivity analysis of the overall protocol architecture by varying the critical factors related to protocol behavior and we show that, in general, FRANCA implementation allows up to 25% of energy saving while keeping the overhead extremely low.

© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Clustering; Ad hoc networks; Routing; Power management

1. Introduction

Installation cost and lack of mobility of traditional wired networks are obstacles to short-term acceptance of home networking solutions. Wireless solutions become therefore interesting because they ease the installation activity and because they enable devices mobility. Anyway, the mobile devices' dependency on batteries or other exhaustible means for their energy, despite recent advances in extending battery life, is still an important issue and the energy usage optimization an

important design criteria [1]. Moreover, the lack of interoperability among the various wireless networking technologies prevents integration of short-range wireless networks. Short-range wireless networks, include the *Personal Area Wireless Networks* (WPAN) and the *Local Area Wireless Networks* (WLAN). *Bluetooth* [2], a WPAN technology, favors low cost and low power consumption to range and peak speed, while *IEEE 802.11* [3], a WLANs technology, favors higher speed and longer range to cost and power consumption.

This paper concentrates on these two technologies and develops a data distribution proposal which provides seamless integration in a power efficient way.

Various research projects aim to improve interoperability among wireless networks. The *Bay*

* Corresponding author.

E-mail addresses: pmanzoni@disca.upv.es (P. Manzoni), jucano@disca.upv.es (J.C. Cano).

Area Research Wireless Access Network project [4] is one of the first projects to develop a digital communications architecture that integrates and operates from regional-area to in-room wireless networks. The *Cellular IP* [5] is a lightweight and robust protocol which is optimized to support local mobility but efficiently cooperates with the *Mobile IP* [6]. The *Iceberg* project [7] has, as a primary goal, to develop an Internet-based integration of telephony and data services spanning diverse access networks. Finally, the Terminode [8] project proposes a new kind of mobile information system: a decentralized, self-organizing network based on (mobile) terminals that could simultaneously work as terminals for users, as well as network nodes connecting inter-user traffic.

This paper presents a distribution algorithm to provide interoperability among mobile and fixed devices independently from the underlying physical media technology. The algorithm we propose, called FRANCA, combines distribution aspects with energy-saving aspects. It is implemented as two separate protocols: the intra-cluster data-dissemination protocol and the inter-cluster distribution protocol. The intra-cluster data-dissemination protocol groups nodes around a special one, called the *cluster leader* (CL). The CL centralizes the *power management* mechanism and acts as a proxy for data transfer between the cluster and the rest of the nodes. It assumes the burden of buffering data frames for power-saving stations and delivering them on stations' request, allowing the mobile stations to remain in their power-saving state for much longer periods. We periodically distribute the CL role among all the nodes inside the cluster, so as not to overload a single node. The inter-cluster distribution protocol takes charge of distributing data packets through the overall architecture by adopting an optimized broadcasting algorithm.

We simulated the protocols derived from FRANCA with the *ns* simulator. We evaluated the overhead and stability of the FRANCA clustering protocol and found it to be minimal. When adopting the power-saving algorithm, it showed an energy saving of up to the 30%. We then carried out a sensitivity analysis of the overall protocol architecture by varying the critical factors related

to protocol behavior and we showed that, in general, the FRANCA implementation saves up to 25% of energy while producing an extremely low overload.

The rest of this paper is organized as follows. In Section 2 we present a review of the contributions in the area of wireless networks routing protocols. Section 3 presents an overall view of our architecture. Section 4 details the intra-cluster data-dissemination protocol while Section 5 details the inter-cluster routing protocol. Section 6 discusses the performances of the overall proposal and Section 7 presents the conclusions.

2. Related work

The *mobile ad hoc networks* (MANET) are an example of MN that do not require any fixed infrastructure, which means that its topology can vary randomly and at unpredictable times. The *Internet Engineering Task Force* MANET working group [9] proposed various routing protocols for ad hoc networks during the past few years. Those protocols can be classified into four broad categories: proactive, reactive, mixed, and cluster-based.

Proactive protocols (e.g., the *destination sequenced distance vector* [10]) perform periodic route updates using control packets to pre-compute paths toward all possible destinations. In reactive protocols the route discovery procedures are invoked on demand when a source has a new connection pending toward a new destination. Examples of this category of protocols are the *ad hoc on demand distance vector* [11], the *dynamic source routing* (DSR) [12] and the *temporally ordered routing algorithm* [13]. Mixed protocols are intermediate proposals which try to dynamically adjust the degree of reactive/proactive behavior. The *zone routing protocol* (ZRP) [14] is the most representative of these proposals. The basic approach of ZRP is to use proactive routing within a zone and to use a two-way handshake reactive protocol for inter-zone routing. ZRP can also be considered a cluster-based protocol in which clusters arbitrarily overlap, since any node maintains its own constrained routing zone.

Cluster-based routing protocols organize the network into groups of nodes, called *clusters*. Krishna et al. [15] proposed a scheme which dynamically divides the whole network into a number of overlapping clusters. The algorithm creates clusters where nodes are mutually reachable using a *1-hop path*. The algorithm tries to find the largest clusters in the dynamic physical topology. The cluster structure is used to give support to an efficient and completely proactive routing mechanism. Lin and Gerla [16], proved the benefits of using cluster organization to support multimedia traffic. The authors used the cluster organization to efficiently manage and allocate resources such as bandwidth and medium access control (MAC) channel access. Basagni [17] introduces a new weight-based clustering criterion. He adopts the approach of using a coordinating node inside the cluster and presents an algorithm suitable for clustering “quasi-static” networks. The concept of a cluster-head is also adopted in [18]. The authors generalize the clustering heuristic so that a node is either a cluster-head or at most d hops away from the cluster-head. In [19] Pei et al. present a routing protocol which combines the features of the Fisheye state routing (FSR) [20] and Landmark routing. The key novelty is the use of landmarks for each set of nodes which move as a group in order to reduce the routing update overhead. As in FSR, nodes exchange link state only with their neighbors. Routes within Fisheye scope are accurate, while routes to remote groups of nodes are “summarized” by their corresponding landmarks. A packet directed to a remote destination initially aims at the Landmark; as it gets closer to destination it eventually switches to the accurate route provided by Fisheye. LANMAR provides efficient and scalable routing in large, mobile ad hoc environments in which group mobility applies.

Finally, Ramanathan and Steenstrup [21] propose the *Multimedia Support for Wireless Networks* (MMWN) system. The MMWN organizes the network by using a multiple level hybrid cluster infrastructure. The multilevel cluster architecture defines three types of elements: switches, clusters, and super-cluster. Basically switches are grouped into clusters and clusters into super-clusters, according to a set of control parameters. Al-

though the MMWN scheme uses a comprehensive approach to organize a scalable ad hoc network, the maintenance of the multilevel hierarchy involves an important overhead.

3. The basic elements

The FRANCA algorithm implementation can be dynamically loaded as a module inside the nodes that we require to be part of the whole system. We considered the diversity among home networking devices and therefore kept our proposal as simple as possible in order to not overload simple mobile computing devices. We provided it with robustness and auto-recoverability properties to allow nodes’ autonomous reconfiguration to be simple and fast. The FRANCA module operates on top of a generic *Network Interface Adaptation Layer* (NIL). This layer deals with the specific physical aspects of each different network technology. The NIL extension varies depending on the technology we are taking into consideration. In the IEEE 802.11 case the NIL is placed on top of the MAC sublayer (see Fig. 1 on the left-side); in the Bluetooth case it is placed on top of the Link manager layer (see Fig. 1 on the right-side).

Both technologies offer some alternatives to power control, by allowing network interface cards (NICs) to vary their transmission power or to be set to a sleep mode state. We defined the NIL to have a simple interface with five basic primitives, namely: NIL_Tx_Frame(), NIL_Rx_Frame(), NIL_ToSleep(), NIL_Wakeup(), NIL_Tx_power(). We are not dealing with interference problems at the physical or at the MAC layer; the IEEE 802.15 [22] Working Group is devoted to define a consensus standard in this area.

While IEEE 802.11 is based on a peer-to-peer configuration, assuming that we are using the ad hoc mode, Bluetooth is based on a master/slave configuration. Bluetooth devices form so-called *piconets* that can be extended to *scatternets*. A piconet is a collection of slave devices operating together with one common master, while a scatternet is a group of piconets with shared members. The only common limitations are on the maximum number of slaves in a piconet, and that a master in

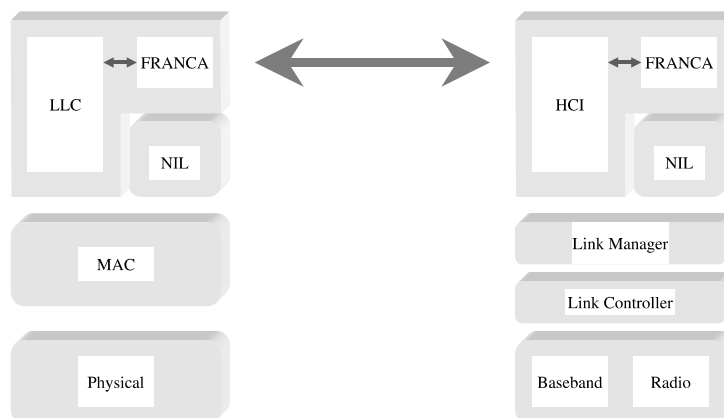


Fig. 1. FRANCA protocols location in the two different technologies.

one piconet cannot also be a master in another piconet. For FRANCA to operate correctly, we require each master to have the FRANCA implementation module installed, since typical Bluetooth slave devices, like a wireless microphone, cannot even reach the minimum computational requirement to run it. We are therefore extending the concept of a cluster node to a piconet, represented by a master. From the FRANCA point of view, this will be transparent, thanks to the master relaying activity. Two situations can occur: either all devices in a domain are using the same technology or a mixed situation can be present. In the latter case we require some of the devices to be used as *bridging nodes*. This simply means that some device while have more than one NIL installed. The FRANCA implementation module will always be unique on any participating device and will take care of all the relative frame relaying activity. Bridging nodes must be masters of the scatternets that are eventually formed.

The FRANCA algorithm is implemented as two separate protocols: the intra-cluster data-dissemination protocol and the inter-cluster routing protocol. The intra-cluster data-dissemination protocol groups nodes around a special one, called the CL. The CL will be in charge of a cluster's maintenance and communication. We periodically distribute the CL role among all the nodes inside the cluster, so as not to overload a single node. The CL also centralizes the *power management* mech-

anism and acts as a proxy for data transfer between the cluster and the rest of the nodes in the domain. It assumes the burden of buffering data frames for power-saving stations and delivering them upon stations' request, allowing the mobile stations to remain in their power-saving state for much longer periods. When a node enables the power management it goes into "sleep" mode, to minimize power consumption. Section 4 describes the details of the clustering algorithms. Routing a packet through this architecture depends on the relative position of the communicating parties. When the communicating nodes belong to the same cluster, direct connection will be used. When the communicating nodes belong to different clusters an improved broadcasting technique is used to avoid the typical problems with broadcast protocols based on flooding. Section 5 describes the details of the routing aspects.

3.1. The FRANCA data units

The FRANCA data unit's header is represented in Fig. 2. The first two bits represent the implementation version. The HLEN is a 2-bit field representing the length of the parameters field as multiples of 16 bits. The RS and RD flags are used by the inter-clusters part of the protocol to indicate whether the source and the destination are inside or outside the cluster. The R/R flag indicate whether the message "type" is a REQUEST

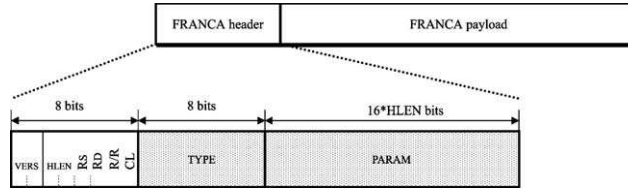


Fig. 2. FRANCA data units header structure.

($R/R = 0$) or a **REPLY** ($R/R = 1$). The **CL** flag indicate whether the node is in the Leader state or not. The octet **TYPE** holds the type of FRANCA messages, that is **HELLO**, **BYE**, **ADD**, **REJECT**, **LOST**, **READY**, **SLEEP**, **OVER**, **LEADER** and **DATA**. Finally, the **PARAM** field contains additional data required for the message.

4. The intra-cluster data-dissemination protocol

Cluster-based routing protocols organize the network into groups of nodes, called *clusters*. The main objective is to make a dynamic network appear “less” dynamic [16]. The use of cluster-based protocols presents the following potential advantages:

- The cluster structure contributes to stabilize network topology thus reducing the number of control packets required to maintain routing information.
- It becomes feasible to schedule packet transmission while adapting the energy transmission power to reduce energy consumption and interference at the same time.
- Clustering provides the infrastructure where special nodes, the cluster’s coordinators, can act as data-traffic proxys. By buffering traffic, a data-traffic proxy, allows other nodes to reduce power consumption by periodically entering into a low power consumption state.

We are obviously aware that clustering also entails a certain number of drawbacks:

- Difficulties in cluster maintenance. Clusters can become unstable as mobility grows.

- Resources consumption due to cluster maintenance activities. The resources consumption can be limited by reducing to the minimum its communication overhead and its computational complexity.
- Scalability, as the network grows and a lot of small sized clusters are formed.
- Interoperability of different technologies and device classes make protocols design more complex.

We model the overall network as a undirected graph G with a finite nonempty set of vertices $V(G)$ (the mobile nodes) and a nonempty set of edges $E(G)$ (the links). We assume the set $E(G)$ will vary with time, that is $E(G)(t)$. The FRANCA intra-cluster data-dissemination protocol is based on a graph-partitioning algorithm. We assume each domain to be a subgraph D_i induced on G by the nonempty sets of nodes contained in each of the $1 \leq i \leq h$ domains of the overall network. We assume that the D_i are connected, undirected graph with edge weights $w : E(D_i) \rightarrow Z_0^+$. The edge weights represent the power required to send a frame between the nodes connected by the edge (we are assuming symmetrical links).

We consider the spanning connected subgraph D'_i induced on D_i by the edge set J , $\forall e \in J$, $w(e) \leq w_{\max}$. The final goal is to partition D'_i into k , $(D'_{i1}, D'_{i2}, \dots, D'_{ik})$, edge-induced subgraphs, the *clusters*, which are fully connected, that is $\forall v, u \in V(D'_{ij}), v \neq u, \exists e \in E(D'_{ij})$. We require $1 \leq |V(D'_{ij})| \leq B$ where $1 \leq B \leq |V(D'_i)|$, $\forall j \in \{1, 2, \dots, k\}$.

We use a graph join algorithm [23] to find the various partitions. The join algorithm substitutes two vertices v_1 and v_2 by a unique vertex v_{12} ; we do not merge edges from v_1 and v_2 that have the same

other endpoint but we create multiple instance of those edges.

We define the *join* procedure as follows:

```

procedure join( $H, K$ )
  choose randomly  $v \in V(K)$ 
  choose randomly  $w \in V(H - K) | vw \in E(H)$ 
  if ( $\forall x \in E(K) \exists xw \in V(H)) \wedge (|V(K)| < B)$ 
  then
     $K \leftarrow K + \{w\}$ 
    join( $H, K$ )
  else
    join( $H, \{w\}$ )
  end if

```

The algorithm is activated in each node by the call: *join*($D_i, \{\}$). The algorithm keeps executing autonomously on all nodes thus handling the possible changes in the $E(D_i)$ due to nodes' mobility.

4.1. The protocol implementation

This section describes the implementation of the clustering algorithm described above. The spanning subgraph is obtained by forcing each node to set its transmission power to $w_{\max}$, as described in Section 6.1. The maximum cluster size B is considered to be equal to the number of nodes in a domain, therefore: $B = |V(D_i)|$. In any event, it could be used from within applications to limit cluster size.

The maintenance of a cluster requires recalling several status variables. We assume that these variables are stored in a *cluster control block* (CCB) in each node. The basic information stored in the CCB is the node status (CCB.STATUS), the cluster size (CCB.SIZE), the CL identifier (CCB.CLID), and the cluster members list (CCB.MLIST). The CCB.MLIST is a pointer to the first element of a circular list, ordered in ascending order according to the member identifier values.

When a mobile node is activated it enters the Leader state and executes the *Init* routine which creates a mono-cluster where the only member is the current node. Nodes in the Leader state send a HELLO message every *hTimeOut* milliseconds. When a node in the Leader state receives a HELLO message it has to decide whether or not to join. We adopted a random-based decision technique con-

strained by the maximum cluster size, but other techniques based on application requirements could be used.

The join process begins with the joining node sending an ADD REQUEST message and then waiting in the Passive Join state for an ADD REPLY message sent from the CL of the destination cluster. A CL, upon receiving the ADD REQUEST message, re-transmits it, and then moves to the Active Join state. The new node is accepted only if each member of the cluster has a 1-hop link with the new node, that is, only if all cluster members have received both ADD REQUEST messages. If none of the members sends a REJECT REPLY message, the join process terminates and the CL sends an ADD REPLY. The use of REJECT REPLY messages instead of positive reply messages, simplifies the protocol and avoids the situation where all members try to send the reply message at the same time, thus generating contention on the shared medium. The newly joined node sends a BYE message and enters the Node state. The BYE message is used to inform the members of the node that they have to start the *CL election process*. If it is not accepted, it simply returns to the Leader state.

While in the Node state, every node checks for link failures with the CL. If a node detects that it has lost a HELLO packet, it sends a LOST message and changes to the Lost state. A node can reach the Lost state either because it moved outside the cluster area or because the CL left. If while in the Lost state, the node does not hear any other LOST message it concludes that it has moved outside the cluster. It therefore sends a BYE message and starts out in the Leader state creating a new mono-cluster. If while in the Lost state, it receives a number of LOST messages equal to the size of the cluster minus two, it concludes that the CL has left. The CL election process then starts. If the number of LOST messages is larger than 1 but less than the size of the cluster minus two, the node enters into a generic instability situation which is handled by the generic recovery technique described in Section 4.4.

Algorithm 1 describes the behavior of FRANCA using Promela [24]. The messages structure is indicated as the sequence of message type, request/reply indication and CL status.

Algorithm 1. Finite-state machine diagram for the clustering algorithm

```

proctype FRANCA_icddp()
1: Leader: if
2:   :: in?ADD, req, - → out!ADD, req, cl →
      goto A_Join
3:   :: in?HELLO, req, cl → out!ADD, req,
      cl → goto P_Join
4:   :: hTimeout → out!HELLO, req, cl →
      goto Leader
5: fi;
6: A_Join: if
7:   :: in?REJECT, rep, - → out!REJECT,
      rep, - → goto Leader
8:   :: hAddTime → out!ADD, rep, - → goto
      Leader
9: fi;
10: P_Join: if
11:   :: in?REJECT, rep, cl → goto Leader
12:   :: in?ADD, rep, - → out!BYE, req, - →
      goto Node
13: fi;
14: Node: if
15:   :: in?HELLO, req, cl → goto Node
16:   :: in?ADD, req, - → goto Add
17:   :: lostHello → out!LOST, req, - → goto
      Lost
18: fi;
19: Add: if
20:   :: in?ADD, req, - → goto Node
21:   :: hTimeout → out!REJECT, rep, - →
      goto Node
22: fi;
23: Lost: if
24:   :: goneCL → electionProcess → goto
      Node
25:   :: nodeMoved → out!BYE, req, - → goto
      Leader
26: fi;

```

4.2. Power management mechanism

The network interface layer consumes a high percentage of the overall energy required for the mobile device. FRANCA includes a power management mechanism based on an experimental

technique to dynamically switch network interfaces off when the corresponding node is not involved in any data reception process [25].

The wireless NICs have varying power consumption behaviors depending on whether they are transmitting, receiving, idle or ‘sleeping’. As shown in [26], NICs are usually in idle mode, which suggests that reducing the energy consumption due to the idle mode is fundamental for saving global NIC energy. The basic approach is to set the NIC in the sleep mode whenever possible. Sleep mode in wireless NIC technology involves not only the radio power shutdown, but also processor sleep modes. It is feasible to enter a sleep mode whenever sleep duration is longer than the time taken to awaken (recovery time). The deeper the sleep mode, the less power required and the longer the recovery time. Section 6 will detail this aspect.

The CL centralizes the power management mechanism. It assumes the burden of buffering data frames for stations in power-saving mode. Buffered data frames will be delivered when requested from the stations. The power-saving mode occurs when setting the node network interface to ‘sleep’ mode. Each station uses a **SLEEP REQUEST** message to inform the CL about how many **HELLO** periods it will remain in the power-saving mode. The CL checks whether it has enough buffer space to reserve, and will accordingly send a positive or negative **SLEEP REPLY** message. This allows the CL to reserve enough buffer space for each station, according to a pre-established policy. The CL has buffer space reserved for multicast and broadcast frames, too.

Once a station wakes up, it checks whether there are data frames waiting by sending a **READY REQUEST** to the CL. If there is buffered data awaiting, the CL will start sending the frames, eventually sending a **READY REPLY**. At this point, the station sends another **SLEEP REQUEST** and returns to power-saving mode. Since the CL has already reserved space for that station, no reply message is required.

This scheme allows the mobile stations to remain in the power-saving state for much longer periods, so saving more battery power. However, this mechanism may affect packet loss index

performance as perceived at the wireless station. Section 6 presents an evaluation of the current global power saving compared to packet loss rate. These results indicate that the power management mechanism performs well in scenarios such as data streams generated by continuous streaming media, i.e., a video source. Short-lived data transmissions do not take advantage of this mechanism, which can, of course, be disabled [27].

4.3. The cluster leader election process

The CL election process is based on a ring-based algorithm. The purpose of the ring algorithm is to distribute the turn of being the CL among all cluster members. Each cluster station, the CL included, stores a circular ordered list (CCB.MLIST) with the addresses of all its fellows.

At the reception of each *ringTime* HELLO message, the CL sends a LEADER REQUEST message indicating the first address value in its list. The indicated node sends then a LEADER REPLY message. The CL, on receiving the LEADER REPLY message, sends an OVER message and transits to the Node state. At the same time, the other nodes validate their entries for the new CL. If, for whatever reason, no LEADER REPLY is received, the CL will eliminate the current element from the list, select the next one, and send another LEADER REQUEST message. The other nodes, on hearing another LEADER REQUEST message, will update their CCB.

4.4. Robustness and correctness

Whenever a node enters an unstable state, for example, because it receives inconsistent information, it must send a BYE message and execute the Init routine. In general, for FRANCA to work correctly, that is to be stable, four conditions must hold:

1. a mobile node must either be in the Leader state, or in the Node state;
2. at any given time, only one member must be in the Leader state: the CL;
3. each member must know its CL's identifier.

4. each node in a domain must belong to a unique cluster;

In this section we sketch a proof of correctness for FRANCA. We select a set of events that can cause changes in the cluster structure and we show that FRANCA always reaches stability within a finite time. A change in the cluster topology can be caused by one of the followings events: a new node is switched on, a node is switched off, the CL is switched off, a node leaves its cluster, and the CL leaves the cluster.

- *A new node is switched on.* As described in Section 4.1, when a node is switched on, it executes the Init routine. This routine builds a new cluster where the four conditions hold.
- *A node is switched off.* A node can be switched off either in a controlled way, or because of a failure. If a node switches off in a controlled way, it will first send a BYE message. Since we deal with 1-hop clusters, all cluster members will receive the message and simply update their membership table by removing the node. If the node switches off due to an error, its absence will be detected by the ring algorithm, as described in Section 4.3. In the worst case, the cluster members realize a member failure before the CL role completes a complete round of the logical ring. Once the nodes know about the faulty node, they remove it from their membership table.
- *The CL is switched off.* The CL can be switched off either in a controlled way or because of a failure. If the CL switches off in a controlled way, it will first send a BYE message. As in the previous case, all cluster members will receive the message, remove the node from their membership table, and then select a new CL as described in Section 4.3. If the CL switches off because of an error, the cluster members will stop receiving the periodic HELLO messages. Consequently, they will send a LOST message and enter the Lost state. In this situation, and as described above, either a new CL is selected or various mono-clusters are created.
- *A node leaves its cluster.* This case corresponds to the sequence of two of the previous

events: *a node is switched off* in a controlled way followed by the event *a new node is switched on*.

- *The CL leaves its cluster.* This case corresponds to the sequence of two of the previous events: *the CL is switched off* in a controlled way followed by the event *a new node is switched on*.

5. The inter-cluster routing protocol

FRANCA uses an efficient and reactive diffusion protocol to avoid the typical problems with broadcast protocols based on flooding. Efficiency is achieved by taking advantage of the energy-saving algorithm and by a fast frame-discarding approach. The protocol PDU header means it is possible to distinguish whether a frame is coming from inside ($RS = 0$) or outside ($RS = 1$) the cluster, and whether the destination is inside ($RD = 0$) or outside ($RD = 1$) the cluster. Every node knows the other members of its cluster through the `CCB.MLIST`. The `DATA` type PDUs are normally handled by CLs only, since the energy-saving algorithm forces the other nodes to sleep most of the time. If a node that is not in the Leader state and receives a `DATA` type PDU with $RD = 1$, it must discard it immediately.

The general behavior of the inter-cluster protocol is quite straightforward. Supposing node *a* inside cluster A wants to send data to node *c* in cluster C. It sends a `DATA` frame to its CL by setting: $HLEN = 1$, $PARAM = 0$, $RS = 0$ and $RD = 1$. The CL broadcasts the frame by switching the RS to 1. When a CL receives a frame with both RS and RD set to 1, it first checks if $PARAM < max_{hops}$. If $PARAM$ is greater than the maximum permitted hops, the packet is silently discarded. Otherwise, the CL has to check whether the destination node is inside its cluster or not. If the destination belongs to its cluster, both the RS and the RD are set to 0 and the frame is buffered, or directly sent to the destination node depending on whether the energy-saving algorithm is active or not. If the destination does not belong to its cluster, the $PARAM$ field is incremented by 1 and re-broadcast. We consider that a value for $max_{hops} = 3$ is a reasonable value

to allow early loop detection in relatively small domains, e.g., $250\text{ m} \times 250\text{ m}$.

6. Simulation

In this section we analyze the performance of the FRANCA implementation, studying its stability and overhead. The simulation results were obtained using the *ns-2* simulator [28]. We evaluate the performance of the intra-cluster protocol with several mobility patterns. We analyze the total energy consumed and the per-state percentage of energy consumed supposing that all mobile devices are equipped with IEEE 802.11b NICs. We also evaluate the packet error rate and the average end-to-end data packet delay. The reference simulation scenario consists of 10 mobile nodes over a $250\text{ m} \times 250\text{ m}$ area. We randomly place nodes at the start within the predefined simulation area. Each simulation run simulates 10 min. We used constant bit rate (CBR) data sources. Based on the basic scenario, where nodes are initially grouped in two main clusters, we analyzed four different dynamic scenarios in an attempt to replicate activities that are likely to occur in bounded areas:

- In scenario #1, each node alternatively selects the random destination in another cluster.
- In scenario #2, nodes select the random destination inside their cluster.
- In scenario #3, a new activity zone appears. Thus, nodes can randomly choose to behave as in scenario #2, or migrate to the new activity zone, and establish a new cluster.
- Finally, scenario #4 combines the behavior of scenarios #1 and scenario #2.

We used the Random Waypoint model [12] as the mobility model. In this model, each node randomly selects a destination and a speed; the maximum speed was 10 m/s. The node then moves to its selected destination at the selected speed. Once it reaches the destination, it stops for a random pause time. The pause time is uniformly distributed between 0 and `PAUSE_TIME`. The node eventually selects a new destination and speed combination, and repeats the movement.

6.1. The energy consumption model

We calculated the energy required to transmit and receive a packet p by using $E_{tx}(p) = 280 \text{ mA} \times v \times t_p$ and $E_{rx}(p) = 240 \text{ mA} \times v \times t_p$, respectively. The values used for the voltage and the packet transmission time were: $v = 5 \text{ V}$ and $t_p = ((p_h / (2 \times 10^6)) + (p_d / (11 \times 10^6))) \text{ s}$, where p_h and p_d are the packet header and payload size in bits, respectively. Since receiving a packet and just being idle are energetically equivalent [29], we assumed $E_{idle}(t) = 220 \text{ mA} \times v \times t$, where t is the NIC idle time. For the sleep mode we considered a current consumption value of 30 mA and a recovery time of 5 ms; therefore the energy consumed by a node that goes to sleep n times and remains in the sleep state during $t \text{ s}$ is $E_{sleep}(t) = (30 \text{ mA} \times v \times t) + (240 \text{ mA} \times v \times (5 \times 10^{-3}) \times n)$.

The total transmission power consumption P_{tx} , can be decomposed as $P_{tx} = P_e + P_{RF}$. The P_e factor represents the power consumption of the control logic and modulator; we considered $P_e = 240 \text{ mA} \times v = 1200 \text{ mW}$. The P_{RF} is the radio frequency (RF) energy and can vary depending on the transmission range. We approximate the maximum as $P_{RF} = (280 \text{ mA} - 240 \text{ mA}) \times v = 200 \text{ mW}$ and considered P_{RF} equal to the radiated power P_r . The NIC we used as a reference specified 250 m as the maximum range, transmitting at 11 Mbps in an open environment with a bit-error rate greater than 10^{-5} . We used a simplified formula for the free space propagation model to calculate the value for P_{RF} assuming a typical radius size for a cluster of about 80–120 m [30]: $P_{RF} = (P_r \times (4\pi)^2 \times d^2) / \lambda^2 = 46 \text{ mW}$.

Therefore, the energy required to transmit a packet p when using a cluster size of about 80–120 m is $E_{txcluster}(p) = (P_e + P_{RF}) \times t_p \simeq 1.25 \times 10^{-3} \times t_p \text{ J} = w_{max} \times t_p \text{ J}$.

6.2. The results

We studied two different configurations: the FRANCA_{S+} which uses the power-saving algorithms and the FRANCA_{S-} which does not use the power-saving algorithms. We evaluate: energy consumption, packet error rate, and packet end-to-end delay. We take into account the effects of

some relevant attributes such as the application sending rate and node mobility. The comparison is made with the equivalent network configuration but substituting FRANCA with the DSR [12] protocol, (see Section 2. DSR was chosen because, while being simple, it was shown to be one of the most efficient routing protocols, especially in bounded regions [26]).

6.2.1. Varying sending rate

We begin by analyzing the effect of varying the application sending rate. We ran simulations while decreasing the packet sending rate from 1.5, 1, 0.5 and 0.25 Mbps (packet size is 512 bytes; node speed is 1 m/s). Fig. 3 compares the average energy consumption of the DSR protocol, the FRANCA_{S-} Protocol, and the FRANCA_{S+} protocol, respectively.

The figure shows that the FRANCA_{S+} protocol saves up to 20%, 22%, 15% and 18% in scenarios #1, #2, #3 and #4, respectively. The explanation is in the use of the FRANCA_{S+} protocol: as cluster interaction increases, as in scenarios #3 and #4, the energy savings decrease. On the contrary, as cluster interaction decreases, as in scenario #2, the energy savings increase. This result is basically due to the fact that as we increase mobility among different clusters, we reduce the possibility for nodes to use the power-saving algorithm and therefore save energy. More stable scenarios such as #2, allow all cluster members to periodically sleep, thus saving more energy. Fig. 3 also shows that the FRANCA protocol behaves identically to the DSR protocol for all simulations. This means that the FRANCA protocol does not have any extra overhead compared to the classical DSR. With respect to the data packet delivery ratio, the DSR protocol outperforms all others, even if the difference is quite small. The FRANCA_{S-} and the FRANCA_{S+} protocols only differ by 1% and 3% respectively. The FRANCA_{S+} behaves extremely well in scenario #2, saving up to 22% of the total energy while delivering the same fraction of data packets as the DSR protocol.

6.2.2. Varying node speed

We evaluate the effect of increasing mobility from walking speed (1 m/s) to slow vehicle speeds

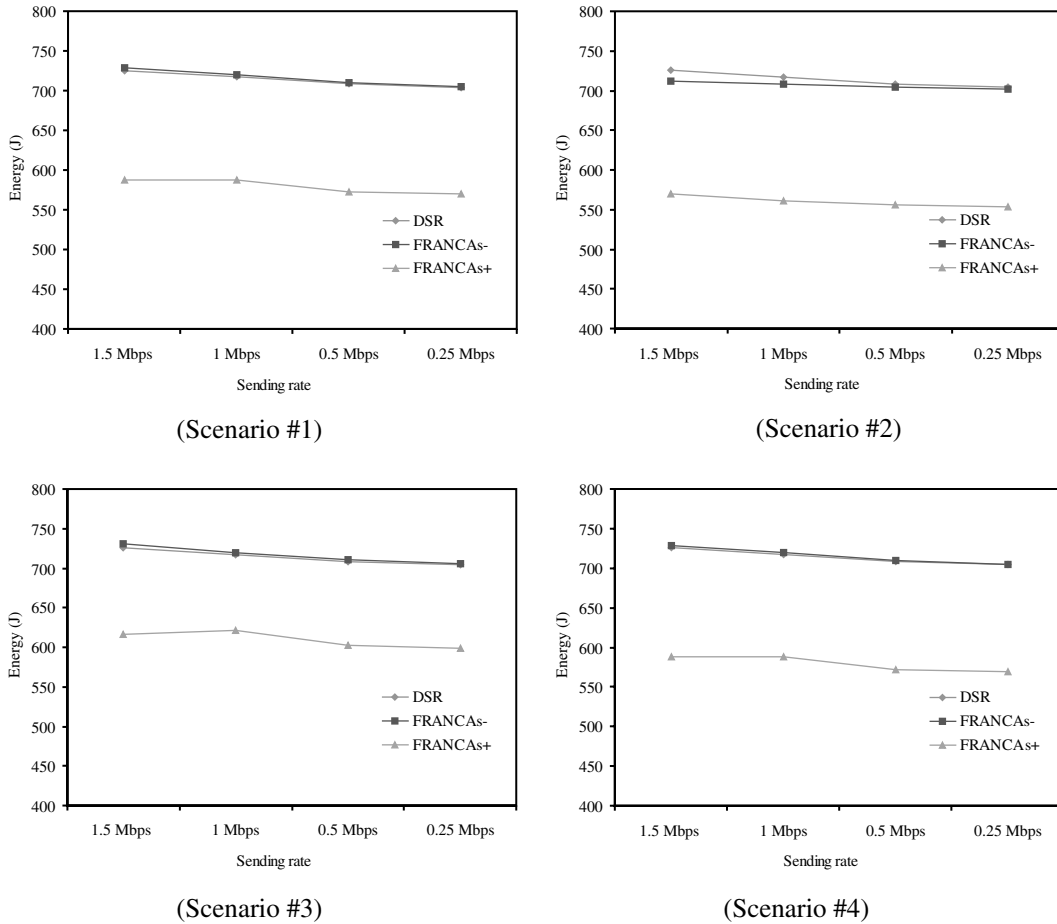


Fig. 3. Energy consumption average. Node speed is 1 m/s.

(10 m/s). We fixed the traffic sending rate to 0.25 Mbps, and repeated simulations by varying the maximum node speed from 1, 2, 5 and 10 m/s. The average energy consumption behaves similarly as in Section 6.2.1. On average, the FRANCA_{S+} protocol can save up to 20%, 21%, 16% and 19% of energy in scenarios #1, #2, #3, and #4, respectively. Node speed does not significantly affect the average energy consumption. Fig. 4 shows that the average packet loss decreases as node speed increases. This is because as node speed increases, the cluster infrastructure stabilizes more quickly, and therefore the proportion of data packets successfully delivered increases too. As an example, the FRANCA_{S+} protocol loses up to 2% of the

total data packets in scenario #1 with a node speed of 1 m/s, yet after increasing the node speed to 10 m/s it only loses 0.3%. Globally, the DSR protocol again outperforms the FRANCA_{S-} and the FRANCA_{S+} by small percentages: 0.5% and 1%, respectively.

6.2.3. The effect of radio transmission range

We evaluate the impact of the radio transmission range on channel utilization comparing the DSR and the FRANCA_{S-} protocols. While the DSR protocol uses a tx power of $E_{tx}(p) = 280 \text{ mA} \times v \times t_p \text{ J}$, the FRANCA_{S-} reduces this value to $E_{txcluster}(p) = 1.25 \times 10^{-3} \times t_p \text{ J}$. This power reduction implies a smaller radio transmission

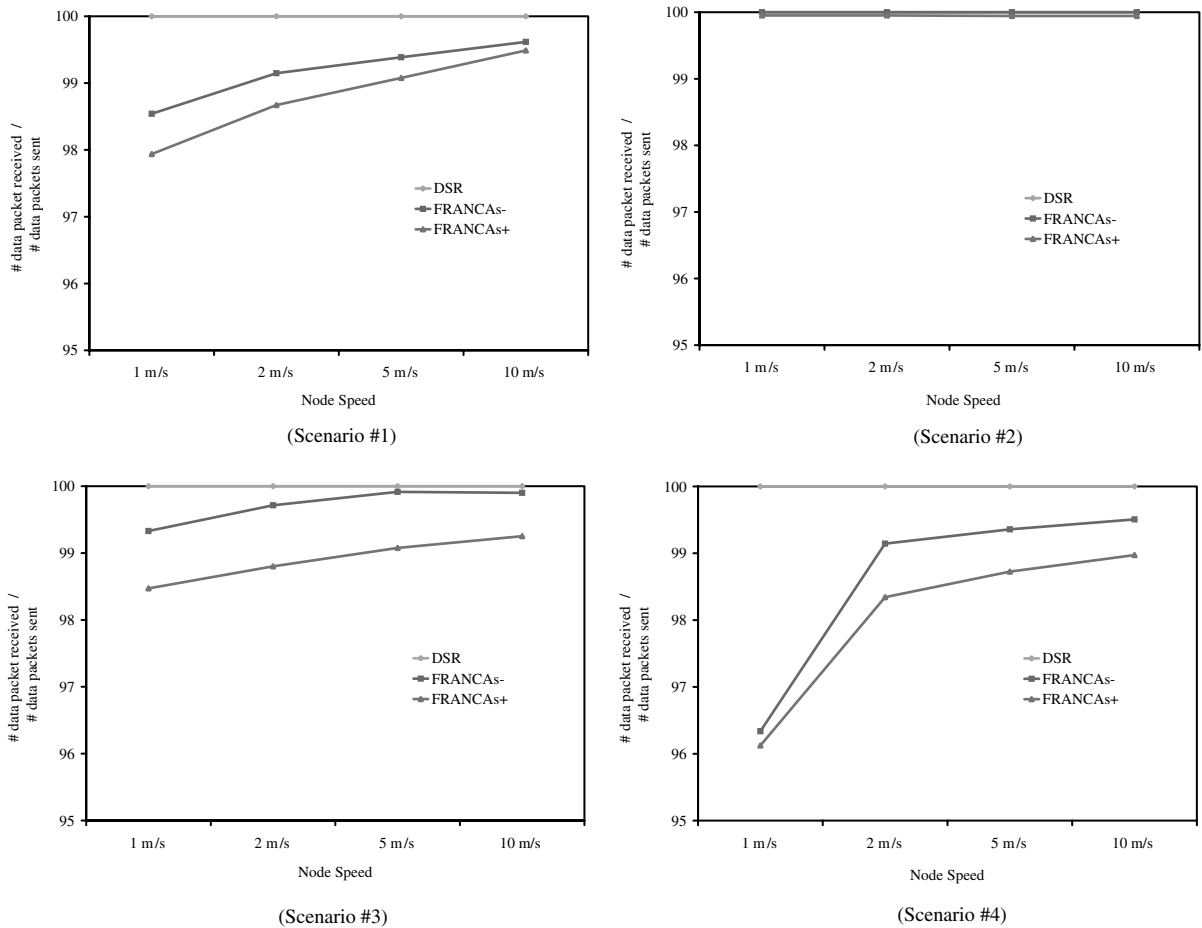
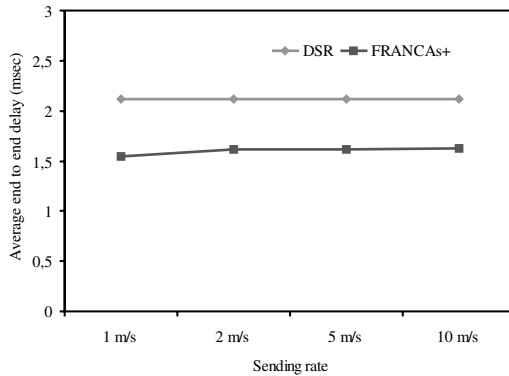


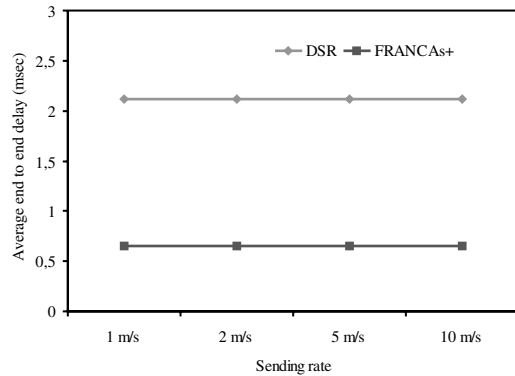
Fig. 4. Fraction of application data packets successfully delivered. Packet sending rate is 0.25 Mbps.

range thus reducing channel interference and increasing channel utilization. We evaluate the average end-to-end data packet delay as a measure of channel utilization. We repeated the simulations of Sections 6.2.1 and 6.2.2 by adding a new peer-to-peer CBR data stream. The results confirm that for all simulations the FRANCA_S- protocol significantly reduces the end-to-end data packet delay with respect to the DSR protocol. The explanation can be found in the radio transmission range. While the FRANCA_S- protocol transmits packets from both sources in parallel; the DSR protocol cannot do the same. Therefore, under the DSR protocol, when packets from the first source are being transmitted, packets from the second source

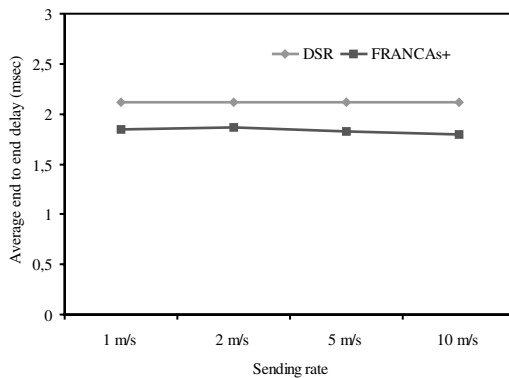
must wait in intermediate buffers until the previous packets arrive at target. This is due to channel interference. Fig. 5 shows the results obtained by varying the node speed. The FRANCA_S- protocol reduces the average end-to-end delay by 24%, 69%, 15% and 14% in scenarios #1, #2, #3 and #4 with respect to the DSR protocol. As the cluster interference increases, routes in the FRANCA_S- protocol become longer, therefore increasing the channel interference, and the end-to-end delay. The end-to-end delay increases also because when nodes periodically go into the sleep state, packets must wait in the CL queues until nodes awaken. Similar results are obtained when varying the sending rate.



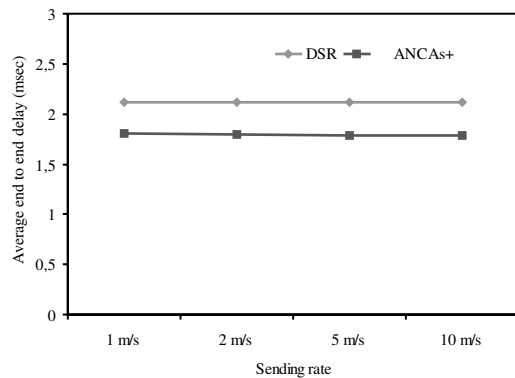
(Scenario #1)



(Scenario #2)



(Scenario #3)



(Scenario #4)

Fig. 5. End to end data packet delay. Packet sending rate is 0.25 Mbps.

7. Conclusions

We consider that our work offers two important contributions. First of all the power-saving algorithm based on the clustering technique we propose, and also the integration possibility offered by the NIL and FRANCA modules.

The idea of the NIL is to have the possibility to adopt FRANCA independently from the underlying technologies. This paper concentrated on the IEEE 802.11 and Bluetooth technologies and presented a data distribution proposal which provides seamless integration in a power efficient way. We concentrate on Bluetooth and IEEE 802.11 technologies because they are the two technologies we are working with in the actual

implementation. We currently have the first prototype of the NIL that works in Linux and Windows 2000 platforms [31] with IEEE 802.11 below, and we are porting them to an USB type HCI for Bluetooth. We do not see any other technology that seems to have a clear advantage over these two. We described a distribution algorithm to provide interoperability among mobile devices, like personal digital assistants, handheld personal computers, displays, printers, or cellular phones which operate in a home area context. The algorithm we propose, called FRANCA, combines distribution aspects with energy-saving aspects, and it is implemented as two separate protocols: the intra-cluster data-dissemination protocol and the inter-cluster distribution protocol. The

proposed algorithm allows mobile nodes to autonomously create clusters to minimize the power consumption.

We simulated the protocols derived from FRANCA with the *ns* simulator. We evaluated the overhead and stability of the FRANCA clustering protocol which we found was minimum. When adopting the power-saving algorithm, our algorithm, showed see an energy saving of up to the 30%. We then presented a sensitivity analysis of the overall protocol architecture by varying the critical factors related to protocol behavior and we showed that, in general, the FRANCA implementation allows up to 25% of energy saving while keeping the overhead extremely low. We also proposed a possible solution to the general problem of broadcasting data in a wireless environment and showed how it affects the performance of the whole network.

We are currently implementing a prototype version on FRANCA-derived protocols to experiment with different realistic mobility patterns and to investigate thoroughly the coexistence issues of the different technologies.

References

- [1] S. Corson, J. Macker, Mobile Ad Hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, RFC 2501, January 1999.
- [2] Promoter Members of Bluetooth SIG, Specification of the Bluetooth System—Core, Version 1.1, Bluetooth SIG, Inc., February 2001.
- [3] IEEE/IEC Std 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, The Institute of Electrical and Electronics Engineers, August 1999.
- [4] R.H. Katz, E.A. Brewer, E. Amir, H. Balakrishnan, A. Fox, S. Gribble, T. Hodes, D. Jiang Giao Thanh Nguyen, V. Padmanabhan, M. Stemm, The Bay Area research wireless access network (BARWAN), Compton'96. Technologies for the Information Superhighway Digest of Papers, 1996.
- [5] A.G. Valk, Cellular IP: a new approach to Internet host mobility, ACM SIGCOMM Computer Communication Review 29 (1) (1999).
- [6] C. Perkins (Ed.), IP Mobility Support, RFC 2002, October 1996.
- [7] H.J. Wang, B. Raman, C. Chuah, et al., ICEBERG: an internet-core network architecture for integrated communications, IEEE Personal Communication 7 (4) (2000) 10–19.
- [8] L. Blazevic, L. Buttyan, S. Capkun, S. Giordano, J.P. Hubaux, J.Y. Le Boudec, Self-organization in mobile ad-hoc networks: the approach of terminodes, IEEE Communications Magazine (June 2001).
- [9] Internet Engineering Task Force, Manet working group charter. Available from <<http://www.ietf.org/html.charters/manet-charter.html>>.
- [10] C.E. Perkins, P. Bhagwat, Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers, ACM Computer Communication Review 24 (2) (1994) 234–244.
- [11] C. Perkins, E. Royer, S.R. Das, Ad Hoc On-demand Distance Vector (AODV) routing, Internet Draft, MANET Working Group, draft-ietf-manet-aodv-06.txt, July 2000, work in progress.
- [12] D.B. Johnson, D.A. Maltz, Y.-C. Hu, J.G. Jetcheva, The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, Internet Draft, MANET Working Group, draft-ietf-manet-dsr-04.txt, November 2000, work in progress.
- [13] V. Park, S. Corson, Temporally-ordered Routing Algorithm (TORA) Version 1—Functional Specification, Internet Draft, MANET Working Group, draft-ietf-manet-tora-spec-03.txt, November 2000, work in progress.
- [14] Z. Haas, M. Pearlman, The Zone Routing Protocol (ZRP) for Ad Hoc Networks, Internet Draft, MANET Working Group, draft-ietf-manet-zone-zrp-02.txt, June 1999, work in progress.
- [15] P. Krishna, N.H. Vaidya, M. Chatterjee, D.K. Pradhan, A cluster-based approach for routing in dynamic networks, ACM SIGCOMM Computer Communication Review (CCR) 27 (1) (1997).
- [16] C.R. Lin, M. Gerla, Adaptive clustering for mobile wireless networks, IEEE Journal on Selected Areas in Communications 15 (7) (1997).
- [17] S. Basagni, Distributed clustering for ad hoc networks, in: Fourth International Symposium on Parallel Architectures, Algorithms, and Networks, Fremantle, Australia, June 1999.
- [18] A.D. Amis, R. Prakash, T.H.P. Vuong, D.T. Huynh, Max-Min D-cluster formation in wireless ad hoc networks, in: IEEE Infocom 2000, Tel-Aviv, Israel, March 2000.
- [19] G. Pei, M. Gerla, X. Hong, LANMAR: Landmark routing for large scale wireless ad hoc networks with group mobility, in: Proceedings of the 2000 First Annual Workshop on Mobile and Ad Hoc Networking and Computing, Boston, MA, August 2000.
- [20] G. Pei, M. Gerla, T.-W. Chen, Fisheye state routing in mobile ad hoc networks, in: Proceedings of the ICDCS Workshop, Taipei, Taiwan, April 2000.
- [21] R. Ramanathan, M. Steenstrup, Hierarchically-organized, multihop mobile wireless networks for quality-of-service support, ACM/Baltzer Mobile Networks and Applications 3 (1) (1998).
- [22] IEEE 802.15 Working Group for WPANs, IEEE 802.15 web site. Available from <<http://ieee802.org/15/index.html>>, May 2001.

- [23] D.R. Karger, C. Stein, A new approach to the minimum cut problem, *Journal of the ACM* 43 (4) (1996) 601–640.
- [24] G.J. Holzmann, *Design and Validation of Computer Protocols*, Prentice-Hall International Editions, Englewood Cliffs, NJ, 1991.
- [25] J.C. Cano, P. Manzoni, Evaluating the energy-consumption reduction in a MANET by dynamically switching-off network interfaces, in: *Proceedings of the 6th IEEE Symposium on Computers and Communications*, Hammamet, Tunisia, July 2001.
- [26] J.C. Cano, P. Manzoni, A performance comparison of energy consumption for mobile ad hoc networks routing protocols, in: *IEEE/ACM MASCOTS 2000: Eighth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, August 2000.
- [27] S. Basagni, D. Bruschi, I. Chlamtac, A mobility-transparent deterministic broadcast mechanism for ad hoc networks, *IEEE/ACM Transactions on Networking* 7 (6) (1999) 799–807.
- [28] K. Fall, K. Varadhan, ns notes and documents. The VINT Project. UC Berkeley, LBL, USC/ISI, Xerox PARC, February 2000. Available from <<http://www.isi.edu/nsnam/ns/ns-documentation.html>>.
- [29] L. Feeney, Investigating the energy consumption of an IEEE 802.11 network interface, Swedish Institute of Computer Science, Technical Report no. T99/11, December 1999.
- [30] T.S. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, Englewood Cliffs, NJ, 1996.
- [31] C.M.T. Calafate, P. Manzoni, PICA: a protocol implementation specific API. Available from <<http://reptar.grc.upv.es/PICA>>.



Pietro Manzoni is an associate professor of Computer Science at the Polytechnic University of Valencia, Spain. He received the M.S. degree in Computer Science from the “Università degli Studi” of Milan, Italy in 1989 and the Ph.D. in Computer Science from the Polytechnic University of Milan, Italy in 1995. His research activity is related to wireless networks protocol design and implementation.



Juan-Carlos Cano is an assistant professor in the Department of Computer Engineering at the Polytechnic University of Valencia (UPV) in Spain. He earned an M.Sc. in Computer Science from the UPV in 1994, where currently he is a Ph.D. candidate. Between 1995–1997 he worked as a programming analyst at IBM’s manufacturing division in Valencia. His current research interests include power aware routing protocols for mobile ad hoc networks. You can contact him at jucano@disca.upv.es