Discovery Mechanisms for Wireless Mesh Networks Management Architectures

José Ángel Irastorza, Luis Francisco Díez, Ramón Agüero, and Luis Muñoz

University of Cantabria, Santander, Spain
{angel,ramon,luis}@tlmat.unican.es,
luis-francisco.diez@alumnos.unican.es

Abstract. The management tasks which have been traditionally employed over traditional wired networks should also play a key role for ensuring the proper operation of the so-called *Personal Networks*, with certain particular characteristics which make their management quite a complex task. Amongst all the challenges which need to be coped with, there is one which outstands over the rest, being ensuring an autonomous operation, qualifying them as *self-** manageable/configurable/... networks. This paper analyzes, over a hierarchical/distributed management architecture, defined to be used over personal networks, the performance of a discovery mechanism by means of which agents are able to locate managers, and associate to them. A complete implementation of the whole architecture has been made in the framework of the *ns-2* simulator (based on SNMP), including the mechanisms and procedures required to handle the discovery and association between managers and agents.

Keywords: Mesh networks, Management, Self-* networks, Discovery mechanisms, Simulation.

1 Introduction

The evolution of wireless devices and terminals, together with the development of wireless network technologies which can be used to interconnect them, are two of the causes of the creation of more versatile, dynamic and user-centric communication environments than the ones which characterized legacy wired network infrastructures. This type of scenarios are frequently based on the use of multi-hop or mesh topologies, where nodes are characterized for their heterogeneity, ranging from laptops (or the more recent smart phones, tablets, etc) to much more limited devices, like sensors or actuators. These two latter types of terminals have recently gathered relevant attention, with the upcoming of the *Internet of things*.

All these new communication environments need to be managed. In this sense, management tasks must be able to ensure an efficient and effective network operation, from the perspective of both the physical resources and the involved distributed systems. Although a great variety of management architectures and models to be used over fixed networks have been extensively studied, this is not

K. Pentikousis et al. (Eds.): MONAMI 2011, LNICST 97, pp. 120–134, 2012.

the case when it comes to the networks which are the focus of this work. These are characterized by: dynamic network topologies, limited bandwidths, unreliable links, collisions, energy limitations, resource and service discovery within dynamic environments, etc.

The design of a management framework for this type of networks should take into account the aforementioned characteristics. In particular, the establishment of an appropriate organizational model is of outer relevance; this is to say, a proper definition of the agent/manager roles and their adequate distribution within the network nodes, so as to ensure a minimum impact of the management overhead over data traffic and the quality perceived by the end-users.

Most of the traditional organizational models are centralized, and they are not appropriate for this type of networks [6], since they were originally conceived to be implemented over fixed networks. With the main goal of overcoming these disadvantages, we present a organizational model which follows a three-level distributed/hierarchical structure. A top level (# 1) manager, which is selected amongst the second level managers, which take a local manager role, controlling a set of nodes, which can be seen as a cluster (with a certain connectivity between them). Finally, agents are located at the third level of the hierarchy. Although three levels are defined, there only exist two management communication planes: one comprising the agents and their corresponding manager (second level), and another one which interconnect all level-2 managers between them and with the overall manager. Hence, we can see the level-2 managers as an overlay network where communication is *peer-to-peer*. This distributed/hierarchical proposal gives a greater reliability and efficiency to the management subsystem, as well as it ensures a lighter overhead, both from the point of view of the communications and the system resources [6].

Using the proposed organizational scheme, two research lines are opened: the first one studies the assignment, by means of certain strategies, of the manager role to a set of network nodes, while the second one focuses on the operation of the discovery mechanisms which are needed so as the agents could become aware of the best manager and associate to it. This work, which belongs to the latter line of research, is structure according to the following points: Section 2 presents the related state of the art and antecedents of this work, so as to better motivate it; Section 3 introduces the strategies which were used to establish the manager deployment, as well as the parameters which were selected so as to compare the performance of each of them; Section 4 depicts the operation of the discovery protocols, which are evaluated, by means of extensive simulation campaigns, in Section 5. Finally, Section 6 presents the main conclusions of this work.

2 Related Work

The relevance given to multi-hop and mesh topologies by the scientific community is sharply increasing. Although the research on the multi-hop networks realm started almost 20 years ago, mainly through the activity of the *Mobile Ad Hoc Networks* (MANET) IETF working group, we have seen in the latest years a continuous change in the perception of this type of networks. Initially they were presented as topologies which could be spontaneously deployed in those situations in which, for any reason, there was not a subjacent infrastructure (typical application scenarios were a war situation or a natural disaster) and, in addition, the network was intrinsically highly dynamic, being the nodes characterized by a relevant mobility. This perception of scenarios and applications has lost its initial relevance and, at the time of writing, mesh topologies are believed to provide a set of other important benefits. Traditional network operators can, for instance, think about using these deployments so as to extend their coverage area on an economical way. In this sense, IEEE working groups dealing with wireless technologies already incorporate multi-hop topologies in their specifications, like IEEE 802.11s [2] or IEEE 802.16j [1]. Likewise, the use of multi-hop deployments is being considered in the the framework of future cellular networks, LTE [11] and was also part of the TETRA [9] specification.

If the importance given to management task has been traditionally rather high for any type of network, this is even more relevant for wireless mesh networks, since they need to adapt themselves to the changing conditions (self-configurable) and their resources should be used (managed) efficiently, since they are more scarce than in wired networks [4].

As was mentioned before, the main goal of this work is to analyze the behavior of the discovery mechanism for a management architecture to be used over a mesh network. In this sense, the goal is not for a node to find a route to any random destination (as it is the case for traditional routing protocols), but to locate the most appropriate manager according to a number of parameters. For that, we start from a previous analysis in which we studied different manager deployment strategies [6] and we focus, this time, on the specific operation and performance of the mechanisms and protocols designed for carrying out such discovery. It is worth highlighting that we will use the same nomenclature as the one which was traditionally used in the *ad-hoc* realm to differentiate the two searching procedures which will be analyzed: reactive and proactive. In the first one, managers do not announce their presence, and thus the agents need to initiate a search procedure so as to locate them; on the other hand, in the proactive scheme, managers periodically announce their presence by broadcast messages and the agents use the gathered information so as to establish to which one they should associate.

From the above, it can be said that this work is close to those which exist in the framework of service or *gateway* discovery. For example, the authors of [3] analyze the delay and performance of the communications between nodes and *gateways*, but they do not describe the way those were deployed in the network. In [7], the authors study security concerns when sending traffic to a set of *gateways* which are optimally deployed within the network. We will use a mechanism similar as the one proposed in [12] to select the best manager (according to a weighted sum of parameters of merit). However, as opposed to all the aforementioned papers, the goal of this work is to thoroughly study the behavior of the manager deployment strategies, analyzing their advantages and drawbacks and their influence on the performance of the discovery mechanisms (association time, discovery traffic overhead, etc). Another aspect which is related to the work which is being presented herewith is *Service Discovery Protocols* (SDP) and their application over multi-hop networks. The authors of [8, 13] survey the various proposals which have been made as well as the challenges which need to be coped with. Although the complexity which is intrinsic to the services is much greater, especially in terms of their description (ontologies) or architecture (overlay, use of directories), the two papers also make the distinction between reactive and proactive discovery. Furthermore, [13] highlights the fact that the evaluation of discovery mechanisms is not mature enough. One of the few works which carries out an analysis similar to the one which will be later presented is [5], in which the authors analyze the overhead and the minimum required time to locate the services; however they study strategies which are based on caching or ring-based search, while we analyze the different manager deployment strategies and the effect they have over the discovery mechanisms.

3 Manager Deployment Strategies

One of the key objectives of this work is to analyze the influence of the four manager deployment strategies which were introduced in [6]. Those were defined according to a basic set of three parameters which would establish the goodness of the strategy.

- *Coverage probability.* It refers to the probability that any agent can establish a communication with, at least, one manager, becoming part of the management architecture.
- *Number of hops.* One of the limitations which are normally attributed to multi-hop topologies is the additional interference which they might bring about. In order to limit them, it would be convenient that the length of the paths between agents and managers was as short as possible.
- *Agent distribution.* This parameter aims at characterizing the fairness of the distribution of agents between the managers. In an optimum scenario, each of the managers should have the same number of agents, while in the worst case, one manager would have all the agents, the others having none. Taking the relative difference between these two situations, we defined the β parameter as follows:

$$\beta = \frac{1}{2\left(A_C - \dfrac{A_C}{M}\right)} \sum_m \left| A_m - \frac{A_C}{M} \right| \qquad (1)$$

  Where A_C is the overall number of covered agents, M the number of managers and A_m the number of agents controlled by the m^{th} manager. It can be seen that the β parameter is restricted to the interval $[0, 1]$, corresponding to the best and worst cases, respectively.

3.1 Strategy 1: Random Deployment

We assume that managers are randomly deployed, without any previous planning. This reflects a worst-case scenario, since, depending on the particular network topology, we could find completely isolated managers, without any node within their coverage area.

3.2 Strategy 2: Geometric Optimal Deployment

In this case we assume that the managers are deployed at those locations which ensure a maximum (*geographical*) coverage of the area under analysis, without considering the particular network topology. Hence, this strategy might not guarantee the best behavior in terms of the coverage probability, since this would heavily depend on the particular position of the nodes.

3.3 Strategy 3: Topological Optimal Deployment

We assume a global knowledge of the network topology, and we select the nodes to take the manager role those which guarantee a minimum overall *cost*, being this cost related to the number of hops which are required to reach a manager. In order to solve this problem, the *p-median* [10] can be used, establishing a set of M managers to minimize the following function.

$$\sum j \in N \left\{ \min_{i \in M} d_{ij} \right\} \tag{2}$$

in which N is the set of all nodes and d_{ij} is the distance (in number of hops) between nodes i and j.

3.4 Strategy 4: Topological Sub-optimal Deployment

One of the disadvantages which are traditionally attributed to the *p-median* method is that its main (and mostly unique) goal is covering all nodes (this is to say, the demand should be satisfied). Considering the particular characteristics of the network topology, it migth be better leaving some nodes aside the management tasks, so as to favor a fairer distribution of the rest of agents. In order to achieve this, we propose a slight modification of the traditional *p-median* algorithm, so that it does not consider those sub-graphs with a size smaller that ν nodes, being ν a design parameter, which should consider the additional benefit of not managing a set of nodes and the corresponding loss (in terms of the probability of being managed).

4 Discovery Protocol

As was said before, the discovery protocol defines two different operations: proactive, in which managers periodically announce their presence and, furthermore, maintain the association with the corresponding agents; and reactive, in which the agents trigger the search for managers (which do not announce their presence) and take an active role in the maintenance. Manager announcements (in the proactive mode) and their search (in the reactive operation) are both based on broadcast traffic, which is disseminated throughout the multi-hop network, and it is only forwarded by the agents. Furthermore, in order to avoid flooding of the network, we have limited the maximum number of hops for any packet. The rest of traffic which is used in the two modes of operation is unicast. Besides, and in order to keep the information of the broadcast traffic updated, nodes maintain tables to store the relevant data about the ones they receive information from (having an opposite role).

In any case, it is worth highlighting that the agents are the ones which finally take a decision about the manager they would try to associate to, while the managers confirm or reject the requests, depending on whether the number of current associated agents surpasses a predefined threshold; this parameter aims at avoiding the congestion which would happen around the manager node. In this way, each table entry maintain a state for the corresponding node: associated, dissasociated or rejected.

The election of the manager by the agents is made by means of a cost function which encompasses three parameters: number of hops p_{hops}, associated agents p_{agents}, and previous state of the association (this latter parameters tries to maintain the already established associations). The agents, after gathering information about the available managers, obtains, for each entry, a weighted sum of the three parameters, using Eq. 3. Each of the ω_j represents the relative weight given to parameter j (we have established that their sum equals 1.0), while $(p_j)^i$ represents the value of the corresponding parameter for the i^{th} manager.

$$(f_{\text{cost}})^i = \max \sum_{j=0}^{C-1} \omega_j \cdot (p_j)^i \quad (3)$$

$$(p_{\text{hops}})^i = \frac{(\text{hops})^{\max} + 1 - (\text{hops})^i}{(\text{hops})^{\max}} \quad (4)$$

$$(p_{\text{agents}})^i = \frac{(\text{agents})^{\max} - (\text{agents})^i}{(\text{agents})^{\max}} \quad (5)$$

In order to model the aforementioned parameters: number of hops and associated agents, we use the established maximum values, $(\text{hops})^{\max}$ and $(\text{agents})^{\max}$, respectively, by means of a linear relationship, which takes the best value (1.0) for one hop and zero agents and decrease until it reaches 0.0 for the worst-case values. For the third parameter (the previous state of the association), we model it as a binary variable, depending on whether the agent was already associated to a particular manager. As was said before, this parameter avoids the unstability which could cause continuous manager changes.

Considering the high variability of the network topology (either because of node mobility or appearance/dissapearance of nodes), the nodes should carry out the association procedures on a periodic fashion, and we define the *Refresh Manager Table* timer (t_{RMT}). In addition, considering that the nodes might

Table 1. Information carried by the discovery packets

Packet type	Source	Dest	Hops	SeqNum	AssocAgents
Manager Announcement	√	√	√	√	√
Manager Request	√	√	√	√	
Association Request	√	√		√	
Association Confirm	√	√		√	
Association Reject	√	√		√	

not be available simultaneously, both modes of operation implement a *back-off* procedure on this timer, which is activated when agents are not aware of any manager.

Table 1 depicts the information carried by each of the discovery packets, which will be thoroughly described afterwards. All of them include the source and destination addresses, as well as a sequence number, used to discard broadcast packets which had been forwarded before. Furthermore, broadcast packets include the number of hops which it has gone through and *Manager Announcement* also includes the number of associated agents.

4.1 Proactive Mode

As has been said before, this mode of operation (Figures 1(a) and 1(b)) is based on the periodic broadcast of *Manager Announcements* (MA), the *manager announcement* timer (t_{MA}) is used for triggering these transmissions and the sequence number is increased for each of them. The first transmission is randomized (within the *manager start interval*, MSI), so as to avoid unwanted synchronizations. MA are propagated throughout the network by the agents (managers silently discard them). After storing the information of the available managers, the agent starts the association process (after the expiration of t_{RMT}, whose first value is the sum of MSI and the *agent start interval*, ASI). The association process implies the selection of the manager which maximizes the cost function introduced before and the transmission of an *association request* to the corresponding manager. When the manager is more than one hop away, the subjacent routing mechanism is in charge of delivering the packets to the appropriate destination. Upon the reception of this request, the manager accepts or rejects the association (*association confirm* or *association reject*), depending on the current number of associated agents.

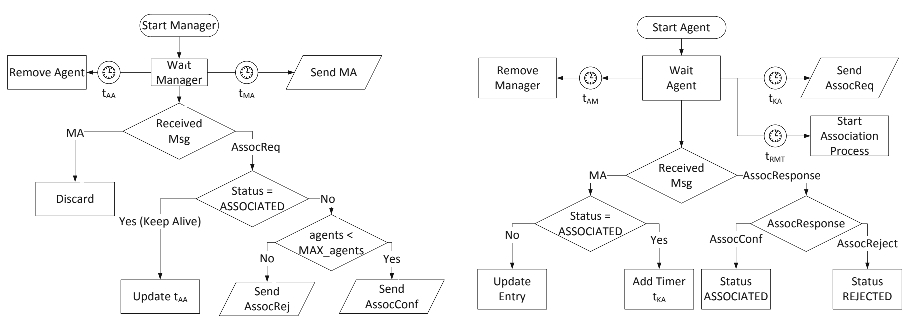
Once the association is completed, the maintenance is performed by means of the responses of the agent to the MA that its manager periodically broadcasts. Upon receiving this packet, the agent starts a random *keep alive* timer (t_{KA}) so as to send the corresponding association request (packet used to maintain the association). This timer avoids the synchronization between the transmissions of all the agents controlled by the same manager after the reception of the MA.

It is important to highlight that the association process which is triggered after the t_{RMT} does not necessarily generate more traffic, since it just checks in the corresponding table whether there is a better manager than the current one. If this was not the case, the process would silently finish, without any further action. Finally, both managers and agents maintain timers (*Alive Agent* - t_{AA} - and *Alive Manager* - t_{AM} -, respectively) to keep track of the accessible ones.

4.2 Reactive Mode

In this case (Figures 2(a) and 2(b)), the discovery is initiated by the agents which, after expiring the t_{RMT}, trigger a searching procedure; the first value is also randomized whithin the ASI interval. A clear difference with the previous mode is that, in this case, agents gather the information about the available managers during this searching procedure, since managers do not announce their presence. Agents broadcast *manager request* packets (which also carry a sequence number). When a manager receives them, they answer with a MA (which is, in this case, sent -unicast- to the corresponding agent); this way, the agents gather the required information. Afterwards, after waiting a time long enough to guarantee the reception of enough information elements (*wait manager announcement* timer, t_{WMA}), the agent chooses the best manager, to which it sends the *association request* (the manager answers as it was described for the proactive mode). After the completion of the association, the agent initiates a *keep alive* timer (t_{KA}) which will be used so as to maintain the association: everytime it expires, the agent sends an *association request* to the manager, which responses with an *association confirm.*

As opposed to the proactive mode, every time a new association procedure is triggered (t_{RMT} expires), the agent is not aware of the available managers, so it must trigger a new searching procedure. In addition, both the t_{AA} and t_{AM} are also used in this operation mode, as can be seen on Figure 2.



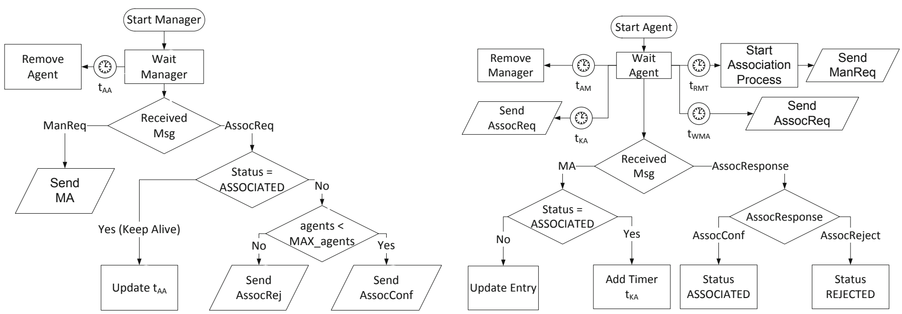
(a) Manager (b) Agent

Fig. 1. Flow diagrams for the proactive mode of operation

5 Discussion of Results

This section discusses the main results which were obtained during the analysis of the different strategies. It discusses both those which were defined in Section 3 (static measurements), as well as the particular behavior of the discovery protocols (dynamic measurements).

In order to carry out the static measurements we have used two complementary approaches: the first one implies a fundamental study based on a proprietary simulator, while the second is based on the *ns-2* platform. The results of the fundamental analysis were thoroughly described in [6], and can therefore be compared with the ones obtained with the network simulation, so as to assess the validity of the implementation. The parameters which will be analyzed are the coverage probability and the β parameter, which were introduced in Section 3. The dynamic measurements compare the performance, in terms of traffic and time, of the two modes of operation of the discovery protocol; for this study we will use the management framework implementation which was integrated in the *ns-2* platform. In this latter case, the *DYMO* protocol was used so as to enable the communications in the subjacent *mesh* topology.



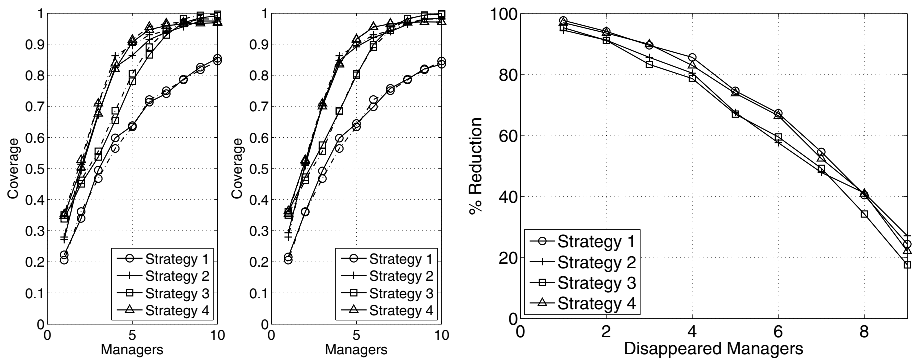
(a) Manager (b) Agent

Fig. 2. Flow diagrams for the reactive mode of operation

In order to perform the analysis, we start from the following parameters: 80 nodes which are randomly deployed within a $100 \times 100\ m^2$ square area. Each of the nodes is equipped with the same *radio access technology* (RAT), having a coverage of 15 m[1]. From this basic topology, the number of managers was modified, and were deployed according to the four strategies. In order to ensure statistical validity of the results[2], 100 independent runs were executed (each of them comprising 600 s) for each of the combinations (managers/agents and strategies). Furthermore, for the particular case of the fourth strategy, we have established not to manage those subgraphs with 2 nodes or fewer (they will not be considered when solving the *p-median*).

1 An ideal circle propagation model has been assumed.

2 Confidence intervals were obtained, but are not presented hereinafter, so as to improve the readability of the corresponding graphs.



(a) Coverage probability for the proactive (left) and reactive (right) modes of operation. Dashed lines represent the results obtained with the fundamental analysis

(b) Coverage after deleting a number of managers (initial setup: 80 nodes and 10 managers)

Fig. 3. Coverage behavior for the different strategies

5.1 Static Measurements

Figure 3(a) shows the coverage probability, which accounts for the number of covered agents against the overall number of agents. We represent, in addition to the results observed with the two modes of operation, those which were obtained with the fundamental analysis, so as to corroborate their validity. It can be seen that the two modes of operation offer the same results than the fundamental analysis, being very similar to each other. Regarding the differences between the strategies, it is clear that strategy 1 is the one which offers the worst behavior, which is sensible, due to its random character. On the other hand, strategies 2 and 4 offer very similar values, while strategy 3 yields a slightly smaller coverage until the number of deployed managers is sufficiently large (when the number of deployed managers is greater than 8, strategy 3 outperforms strategy 4 in terms of coverage); strategy 4 keeps a constant value due to the agents which are contained on the subgraphs with 2 or fewer nodes.

To complement the previous results, we have also analyzed the coverage which would result when a certain percentage of managers dissapear from the original network. Figure 3(b) represents the loss of coverage with respect to the original one (obtained with all the managers); we have used the situation in which there were 10 managers, which are being deleted. This can be used to assess the reliability of the strategies upon the loss of nodes and the number of managers which might be lost before carrying out a reassignment of roles. The values were obtained with the fundamental analysis, since (as was seen before) there are not differences with the results obtained with the *ns-2* implementation.

As can be seen, strategies which get more affected by the loss of managers are #2 and #3, while the topological sub-optimal deployment (strategy 4) shows a

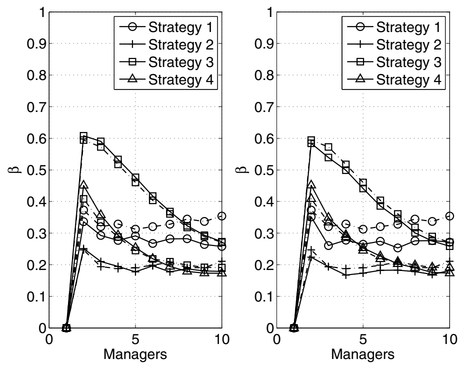


Fig. 4. β parameter for the proactive (left) and reactive (right) modes of operation. Dashed lines represent the results obtained with the fundamental analysis.

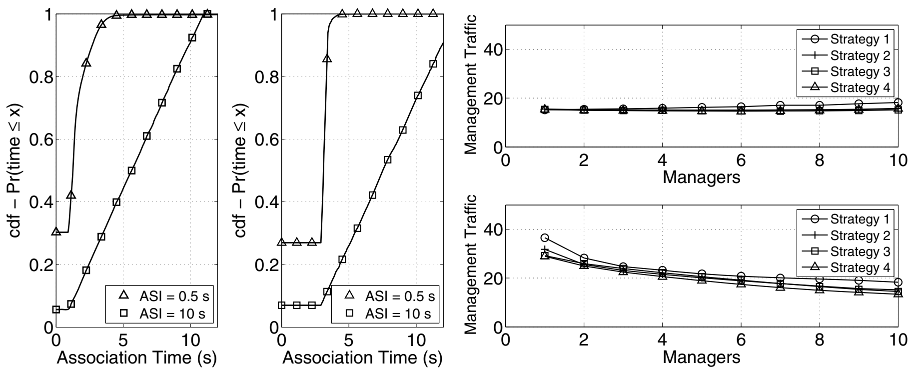
behavior similar to the one exhibited by the random case. It is worth highlighting the behavior of strategy 2, which albeit yielding a performance similar to strategy 4 (in terms of coverage), it shows almost a 10% difference after manager dissapearance. Last, but not least, it is important to remark that although Figure 3(b) might give the impression that the random deployment has a good behavior, this is the one which yields the lowest coverage and thus, the loss of coverage upon manager dissapearance is not (in relative terms) very remarkable.

Another characteristic which is desirable for the management architecture is a fair distribution of the agents between the managers; as was previously discussed, we introduce the β parameter for analyzing this aspect. As can ben seen on Figure 4 there are not remarkable differences between the behavior obtained with the fundamental analysis with that observed using the *ns-2* implementation, with the exception of strategy 1, likely due to the random manager assignment.

The results on Figure 4 shows a clear difference between strategies 3 and 4, since they yield rather distinct behaviors (in terms of the β values), although both of them consider the network topology. Since it does not cover the subgraphs with 1 and 2 nodes, the sub-optimal deployment leads to a fairer distribution of the management burden. Regarding the random deployment, it is worth saying that the β values it yields are even better than those seen for strategy 3, which reflects the penalization of the *p-median* method and its goal to cover all the demand. Last, it can be seen that strategy 2 shows the best performance, likely due to the fair distribution of managers within the scenario.

5.2 Dynamic Measurements

The set of results which are presented in this subsection have been all obtained with the *ns-2* implementation and can be used so as to analyze the intrinsic behavior and performance of the discovery protocols. All the measurements have been carried out over a network topology comprising 80 nodes (10 of them taking the manager role).



(a) cdf of the association time for the proactive (left) and reactive (right) modes of operation

(b) Management traffic overhead for the proactive (top) and reactive (bottom) modes of operation

Fig. 5. Dynamic measurements for the two modes of operation

First, we analyzed the time required for any agent to associate with a manager, seeing the relationship with the random intervals which were introduced before (MSI and ASI). The measurements which were made showed little differences between the various strategies, so we will only use strategy 4 in this case. We have studied the complementary distribution function (*cdf*) of the time required to complete the first association (if this happens correctly), and we have analyzed the influence of the ASI interval. The MSI interval was fixed to 1 second, while the t_{WMA} interval was 3 seconds.

In the proactive mode, an agent will receive the MA from the managers during the MSI interval, and once this expires, it will trigger the association process with the *best* manager, in a time which is randomly selected within the ASI interval. In this sense, the association time should be uniformly distributed in the interval $[MSI, MSI + ASI]$, and therefore, the corresponding *cdf* should be a straight line with a slope of $\frac{1}{ASI}$ in such interval. Figure 5(a) shows that for a high ASI value (10 seconds), the association time matches the expected cdf, while the behavior gets more unpredictable if we reduce the ASI. In this case, it is important to remark that the value of the cdf for $t = 0$ corresponds to the probability of being uncovered, which is much higher than the expected value (note that we intentionally disabled the back-off procedure of the association process this time, since we wanted to study the time for the first association).

In the reactive mode, agents start with the transmission of *manager request* packets within the ASI interval; then, they wait until the expiration of t_{WMA}, when they send the association request. Therefore, the association time should be uniformly distributed within the interval $[t_{WMA}, t_{WMA} + ASI]$. As can be seen, the results yield a better performance this time, even for low values of ASI (0.5 seconds) the *cdf* matches quite well the expected behavior (it shows a linear trend), although the coverage probability is again penalized.

Last measurement studies the influence of the number of managers on the management traffic (relative to the number of covered agents), so as to analyze the efficiency of the different strategies. Figure 5(b) shows the management packets which are transmitted per minute and covered agent. In this case, there is not a relevant difference between the various strategies, but the two modes of operation shows rather different behavior. It can be seen that this parameter shows a constant value for the proactive case (with a light increasing tendency); on the other hand, for the reactive mode of operation, the overhead is much higher when the number of managers is low, but it sharply reduces as long as we increase them; it reaches the values which were observed for the proactive case, but it seems to keep the decreasing tendency. In the proactive case, the overhead is constant, since the managers periodically announce their presence (and this does not depend on the number of managers); for the reactive case, when the number of managers is low, the agents would invoke the association process (and the corresponding searching procedure) periodically, thus causing the high overhead values which we can see on the figure.

6 Conclusions

This work has analyzed the behavior of an autonomous management architecture over a wireless multi-hop scenario (mesh network). We started from a hiearchical/decentralized organizational model, since it reduces the penalization that management tasks can cause on the subjacent network.

To reach this goal, we have proposed a set of manager assignment strategies, based on a number of figures of merit. The presented results (which have been obtained with a more analytical study and also with an implementation within the *ns-2* framework), can be used to establish various main conclusions. First, it is important to ensure an appropriate manager selection, since there might be remarkable differences depending on the particular selection strategy; furthermore, the novel heuristic which was proposed to enhance the *p-median* performance offers very interesting results, since it yields better behavior (in terms of agent distribution), without major decrease on the coverage probability.

From a more realistic application perspective, once the managers have been selected/deployed, agents must discover them so as to complete the association. For this, we have proposed two discovery mechanisms, which are fundamental to ensure the autonomous behavior which is being pursued. The discovery protocol (with the proactive and reactive operation modes) has been designed and implemented within the *ns-2* framework and, using such tool, we have analyzed their behavior in terms of the stabilizing and self-learning capabilities (association time) and of the extra management traffic overhead which is generated. The obtained results show that both operation modes offer similar performances to the one which was assessed with the fundamental analysis, being the differences (in terms of coverage probability and agent distribution) almost negligible. Regarding discovery protocols we have seen that, for the particular characteristics of the analyzed scenario, the reactive mode has a slightly better behavior, since it shows a greater stability against the starting interval for the agents and, which is even more important, is able to reduce the overhead caused by the discovery protocol, as long as we increase the number of managers within the scenario.

From this work, we can open various lines of research, some of which are already started. On the one hand, it would be interesting to analyze the appropriateness of the deployment strategies, considering other application scenarios, like the connection to an infrastructure network, using the managers, which would take a *gateway* role in this case. Besides, another interesting aspect to strengthen is to benefit from the implemented framework to analyze management procedures over mesh networks (e.g. optimum channel assignment, transmit power selection, etc).

Acknowledgements. The authors would like to express their gratitude to the Spanish government for its funding in the following project: "Cognitive, Cooperative Communications and autonomous SErvice Management", C3SEM (TEC2009-14598-C02-01).

References

1. IEEE 802.16 air interface for fixed and mobile broadband wireless systems. amendment 1: Multiple relay specification (IEEE 802.16j) (June 2009)
2. IEEE 802.11 wireless LAN medium access control (MAC) and physical layer (PHY) specification - ESS mesh networking (IEEE 802.11s) - draft 8.0 (December 2010)
3. Ashraf, U., Abdellatif, S., Juanole, G.: Gateway selection in backbone wireless mesh networks. In: Wireless Communications and Networking Conference, WCNC 2009. IEEE (2009)
4. Ferreira, L., De Amorim, M., Iannone, L., Berlemann, L., Correia, L.: Opportunistic management of spontaneous and heterogeneous wireless mesh networks. IEEE Wireless Communications 17(2), 41–46 (2010)
5. Hoebeke, J., Moerman, I., Dhoedt, B., Demeester, P.: Analysis of decentralized resource and service discovery mechanisms in wireless multi-hop networks. Comput. Commun. 29, 2710–2720 (2006)
6. Irastorza, J.A., Agüero, R., Muñoz, L.: Manager Selection over a Hierarchical/Distributed Management Architecture for Personal Networks. In: Pentikousis, K., Agüero, R., García-Arranz, M., Papavassiliou, S. (eds.) MONAMI 2010. LNICST, vol. 68, pp. 210–224. Springer, Heidelberg (2011)
7. Matsuda, T., Nakayama, H., Shen, S., Nemoto, Y., Kato, N.: On gateway selection protocol for DYMO-based MANET. In: IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WIMOB 2008, pp. 32–37 (2008)
8. Mian, A., Baldoni, R., Beraldi, R.: A survey of service discovery protocols in multihop mobile ad hoc networks. IEEE Pervasive Computing 8(1), 66–74 (2009)
9. Pabst, R., Walke, B., Schultz, D., Herhold, P., Yanikomeroglu, H., Mukherjee, S., Viswanathan, H., Lott, M., Zirwas, W., Dohler, M., Aghvami, H., Falconer, D., Fettweis, G.: Relay-based deployment concepts for wireless and mobile broadband radio. IEEE Communications Magazine 42(9), 80–89 (2004)
10. Resende, M.G.C., Werneck, R.F.: A hybrid heuristic for the p-median problem. Journal of Heuristics 10, 59–88 (2004)

11. Schoenen, R., Halfmann, R., Walke, B.H.: MAC performance of a 3GPP-LTE multihop cellular network. In: Proceedings of the IEEE International Conference on Communications, ICC, pp. 4819–4824. IEEE (2008)
12. Setiawan, F., Bouk, S., Sasase, I.: An optimum multiple metrics gateway selection mechanism in manet and infrastructured networks integration. In: Wireless Communications and Networking Conference, WCNC 2008, pp. 2229–2234. IEEE (2008)
13. Ververidis, C., Polyzos, G.: Service discovery for mobile ad hoc networks: a survey of issues and techniques. IEEE Communications Surveys Tutorials 10(3), 30–45 (2008)