Optimizing Wireless Ad Hoc Communication Resources with a Knowledge Plane

Daniel F. Macedo*, Aldri L. dos Santos†, Guy Pujolle*, José Marcos S. Nogueira‡
*Laboratoire d'Informatique Paris VI
Université Pierre et Marie Curie-Paris6, France
† Informatics Department
Federal University of Paraná, Brazil
‡ Computer Science Department
Federal University of Minas Gerais, Brazil
E-mails: Daniel.Macedo@lip6.fr, aldri@inf.ufpr.br, Guy.Pujolle@lip6.fr, jmarcos@dcc.ufmg.br

***Abstract*— In mobile wireless ad hoc networks (MANETs), layering is frequently broken in order to deal with changes on the wireless medium. Layering is also violated in order to implement autonomic behavior, which depends on correlating data scattered on various layers to identify relevant events. This paper proposes a new networking plane, called *MANET knowledge plane* (MANKP), that stores information concerning all layers of the protocol stack. This plane is used to improve network performance, as protocols may employ a broader range of inputs on their algorithms. Moreover, being an information repository, MANKP allows the creation of self-optimizing and self-configuring mechanisms for all the protocol stack. To showcase our solution, we propose new transmission power control aware protocols, where the physical, MAC and routing layers cooperate using MANKP. A preliminary evaluation using asymptotic complexity shows that MANKP significantly reduces the amount of messages sent, decreasing energy consumption.**

I. Introduction

Mobile devices are used in situations where there is no pre-installed infrastructure, or the existing infrastructure cannot be used due to catastrophic events. Hence, the devices must organize themselves to form a mobile wireless ad hoc network (MANET) acting as source and data sinks, but also as data forwarders [1], [2]. MANETs face several challenges that are not found in wired networks. Node mobility, the dynamism of the wireless medium and limited battery and resources must be considered on protocol design.

Due to the high dynamism of the physical medium, several protocols in the literature require the collaboration among layers. They are termed *cross-layer*, as they violate the layered approach [3]. The amount of existing cross-layer protocols shows that the layering model should be revised. At the same time, the concept of autonomic networking demands the creation of an entity that oversees the operation of the protocol stack as a whole [4]. This entity should evaluate the state of all networking layers in order to identify significant events and suggest actions based on the current configuration of the protocols. Thus, MANETs demand a mechanism that allows collaboration among layers.

Autonomic networks, thus, must have two new planes, as Figure 1 shows. The *data plane* encompasses the existing protocol stack, while the *knowledge plane* (KP) stores information concerning the state of the network. The *control plane*, placed between both, acts as an autonomic manager, using the data on the KP to define actions to be executed by the data plane. Several autonomic architectures and middlewares have been proposed, however none of them describe how information is gathered from protocols and applications, assuming that information is available, and hence focusing on control algorithms and architectures. Though, since existing protocols were not created with measurement and management in mind, all the required information is kept hidden inside data structures on each protocol. Further, we must develop mechanisms that make this knowledge available to all protocol layers and to autonomic solutions.

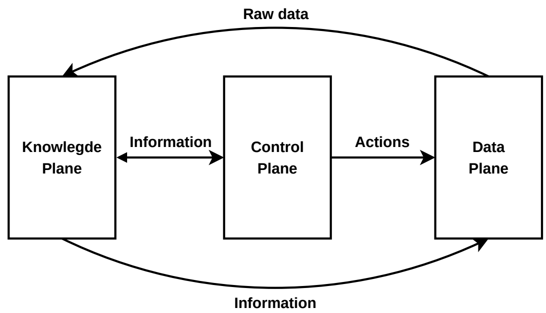

Fig. 1. The three planes on autonomic networks.

In this paper we propose a *knowledge plane* for MANETs, called MANKP, that concentrates the information concerning the operation of MANETs. MANKP implements mechanisms to retrieve and update the stored information, thus allowing protocols to improve their performance using augmented inputs. We perform a preliminary analysis of the benefits of MANKP using a case study. We propose and asymptotically evaluate a set of transmission power control (TPC) aware protocols that employ MANKP. TPC protocols adjust the transmission power on each link in order to decrease energy consumption and contention. They also build energy-aware routes that take into account the energy consumption on each link. Such protocols require a tight cooperation among layers, thus being a showcase for MANKP. Our analysis suggests that MANKP simplifies their implementation and reduces the energy consumption of the network.

The rest of this paper is organized as follows. Section II details the requirements of a KP for MANETs. Section III depicts the related works. MANKP is detailed in section IV, followed by a case study. Section VI presents our conclusions and future work.

II. Requirements for a Knowledge Plane on MANETs

Distributed (DS): The KP must be distributed, in order to avoid central points of failure or bottlenecks on the network. The distributed approach also allows easier recovery from network partitions.

No replicated effort (RE): The KP must expose information from layers whenever possible, avoiding replicated efforts to obtain the same data. In P2P networks, for example, nodes use PING messages to assess if their neighbors are still reachable, a task already done in routing.

Push and pull access modes (PP): The KP must support two data access types: The *pull* access, where the consumer requests information from the producer. This access is fine for scarce queries, however the KP must reduce periodical querying (polling). Thus, it must also support *push* accesses, where the KP warns the consumer when an event occurs.

Support hardware-based information (HI): The KP must expose hardware information, such as node location, speed or remaining energy, to protocols. Routing algorithms, for example, may use energy and geographic coordinates to optimize routing [5]–[7].

Multi-protocol packets (MP): The KP must aggregate messages from several layers, reducing the amount of packets sent and thus the energy consumption. For example, we could aggregate MAC messages that assess the reliability of the links with route reconstruction messages into one single packet [8].

Full information of the neighborhood (IN): The KP must store information concerning the condition of each node in a neighborhood, instead of having an aggregated view of the network. By aggregated we mean, for example, "the average queue occupation is 10%". An aggregated view is not suitable for tasks such as transmission power aware routing, which requires the transmission power value for all one-hop neighbors [9].

Variable-sized neighborhood (VN): Due to scalability issues, nodes will not store information concerning the entire network. Thus, the KP must store information from a limited range of nodes, based on the needs of each protocol. While MAC and routing protocols require information from one-hop neighbors, transport protocols must take into account all nodes that form the paths. In XCP, for example, the data-rate is dictated by the available bandwidth on the bottleneck link [10].

Secure access (SA): The KP must avoid data corruption caused by ill-intentioned or buggy applications. Thus, the KP must implement authentication and access control policies. Also, sensitive information, such as encryption keys and certificates, must be hidden from unauthorized applications.

External queries (EQ): Some services might demand the access to information stored on nodes outside the scope of the KP, due to its rare use.

III. Related Work

Since cross-layer design is common in MANETs, Kawadia and Kumar describe the pros and cons of layered and cross-layer design. They argue that cross-layering might induce a high layer interdependency, producing less modular code [11]. Several papers propose cross-layer algorithms to optimize network performance based on augmented information [5]–[7], [10]. Those propositions assume information availability, while in fact they are restricted only to the protocols that produce them. Autonomic networking architectures create an autonomic management plane to control protocols as a whole [4], [12]–[16]. Autonomic behavior can be dictated by policies that implement adaptability to context. Policies are either low-level, describing events and actions to be taken [17]–[19], or high level, describing business rules [20]. High level policies are later translated into lower-level, device-specific policies.

The works above suppose the existence of mechanisms that fetch and produce information concerning the operation of the network stack on each networked element, but no such thing exists nowadays. Hence, our work is complementary to such propositions, because MANKP will provide the substrate over which policy-based systems, autonomic planes and cross-layer protocols will be built. Other works propose the creation of a plane to gather the information produced by all protocols, as this paper proposes. However, their applicability on MANETs is limited due to their lack of information concerning nodes other than the current node, as we show below.

Turi proposed the creation of event-based cross-layer interfaces to notify protocols of significant events on other layers, simplifying the design of cross-layer protocols while at the same time keeping the benefits of layering [21]. The authors demonstrate their KP on a cross-layer version of Gnutella, where the neighbors of a peer are determined by the connectivity on the routing level. This approach has several limitations. First, there is still replication of information, since protocols cannot see each others' data unless they are explicitly exported. No push-based access is provided, and the information is not available to other nodes. Razzaque et al. improve this vision by creating a new plane that is dynamically fed by the protocol stack [22]. Protocols use *contextors* to insert and query information from the KP. However, the stored information is still limited to the current node.

Wolf created a KP for the POEM project [23]. This plane is divided into the local and the global views. The local view stores information about the host node, while the global view provides a glimpse on the overall state of the network. As an example, the global view could indicate that the average queue utilization is 10%. Each node periodically propagates its local view to its neighbors, which feed this information into aggregation algorithms to update their global view. Although more flexible than previous ones, this KP still lacks information from individual nodes, which is essential to deal with more complex problems such as transmission power control.

The access and modification rights on the KP must be controlled, as buggy or ill-intentioned applications might interact with the KP. Apart from the Haggle architecture [14], any previous work secures information access. In Haggle, each data has an associated access control list, however the authors do not show how those permissions are created and managed.

Table I summarizes the capabilities of each proposed KP. None implements all the requirements described on Section II. Thus, we decided to design MANKP, implementing all the requirements mentioned earlier.

TABLE I
SUMMARY OF RELATED KPS FOR MANETS

Proposal	DS	RE	PP	HI	MP	IN	VN	SA	EQ
Turi [21]	√	√	X	X	X	X	X	X	X
Razzaque [22]	√	√	√	X	X	X	X	X	X
Wolf [23]	√	√	√	√	√	X	X	X	X

IV. THE MANKP PLANE

MANKP is a distributed KP, where each node stores information concerning itself and its neighbors. The MANKP module installed on each node is composed of two blocks. The first block, called Networking-level Knowledge Plane (NKP), stores information on the layers one through six of the protocol stack[1]. The second block, called Application-level Knowledge Plane (AKP), allows secure queries from applications or from external sources. Figure 2 shows the organization of MANKP.

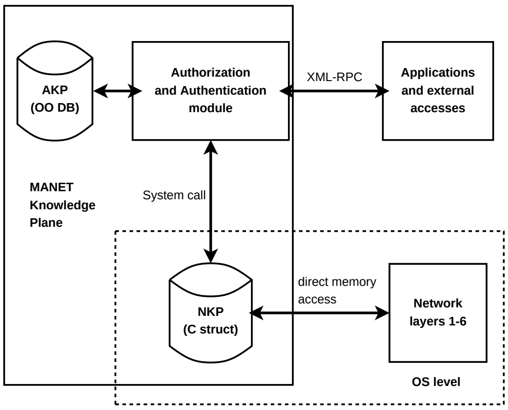

Fig. 2. Diagram of the MANKP module installed on each MANET node.

A. General Organization

As the NKP block provides information to the networking layer, it must be responsive and simple, because such information may be accessed many times per incoming/outgoing packet. Thus, we model the NKP as a C-like kernel-level structure where protocols have full read and write access. This approach avoids costly operations, such as context changes and XML marshaling. MANKP considers that layers 1-6 are trustworthy. It assumes that the OS verifies the authenticity and integrity of the code before loading it. Thus, protocol security is out of the scope of this work.

The AKP block stores the information concerning applications and overlay networks. It is object-oriented, allowing the storage of complex information. The AKP acts as a proxy among the applications and the NKP, allowing read-only access to the information stored on the NKP and hiding sensitive information. To allow a consistent view of application and kernel-level data, the AKP translates the information stored in the NKP into an object-oriented representation. Applications access the AKP using XML-RPC queries, which are well-suited to mobile devices [24]. Using this mechanism, MANKP allows queries from anywhere on the network and guarantees that applications have an homogeneous interface, if they are running on the same node than MANKP or on another node.

The division of MANKP into two blocks gives fast access for networking protocols and allows secure and reliable information sharing for the applications. Applications are considered untrusted entities, and thus have read/write access to their own data and read-only access to data on the NKP. All application queries are controlled by the Authorization and Authentication module. Protocols also access AKP data, however such requests will be slower due to the use of XML.

B. Interaction with MANKP

The MANKP module on each node periodically sends broadcast messages in order to update the stored information of the MANKP modules on neighbor nodes. Before sending such messages, MANKP requests the registered protocols to embed their own data, allowing each level of the stack to define what information must be sent. When the message arrives on the neighbors, their MANKP module notifies the concerned layers. Those layers process the received information, extracting useful data and updating their own MANKP.

Figure 3 shows the organization of the MANKP module, stored in each node, in classes. We describe below how protocols interact with the NKP using the classes specified above. The same access patterns are used on the AKP, however they are performed using XML-RPC calls and requests must be approved by the Authentication and Authorization module.

The class `MANKP` stores network information in a public field, allowing push access and easy data update[2]. The method `kpChange` on the `Layer` class, which is a parent class for all layers, allows pull access. To activate pull queries, layers call the `attach` method, indicating what event they wish to monitor. In order to update the stored information on each node, MANKP periodically executes the `sendKPPacket` function, which calls the `sendPacket` function on all layers, allowing them to piggy-back their data (Algorithm 1). Upon the reception of a KP packet, MANKP verifies the payload of

[1]From this moment on, *protocols* are the distributed algorithms that run below the application layer.

[2]On the AKP level, the access is based on XML-RPC `get` and `set` methods.

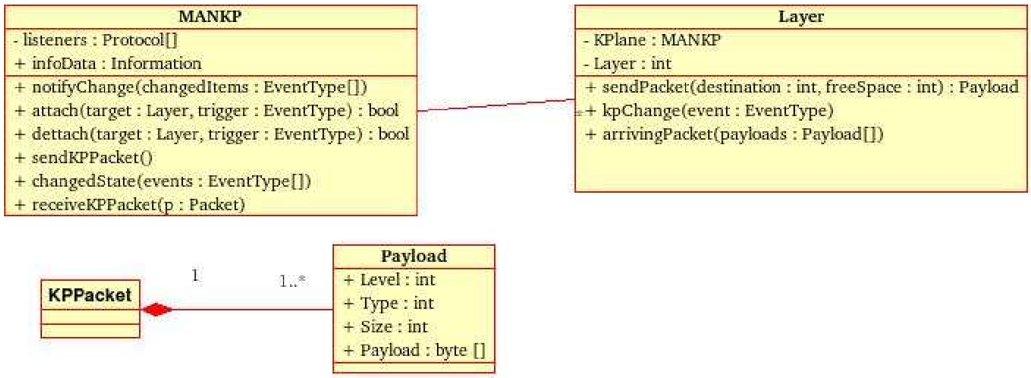

Fig. 3. Class diagram for the interaction among protocol layers and MANKP.

the message, which might contain data from one or more layers. It calls the `arrivingPacket` function of the concerned layer based on the identification level of the payload. The layer, in turn, processes the data and updates the information in the NKP. Protocols register each information change by calling the `changedState` function. After all layers have treated their payloads, MANKP notifies listening layers for change events. For example, if the MAC layer updates the expected link reliability of a given neighbor and the routing layer is registered to watch changes on this field, MANKP will notify routing using the `kpChange` method. This interaction is modeled upon the Observer design pattern [25].

Algorithm 1 The main functions of the MANET KP.

```
class MANKP
  function sendKPPacket( )
    foreach L in Layers do L.sendPacket()
  function receiveKPPacket( )            # Process KP packets
    foreach Payload in Packet do
      L = findLayer(Payload)
      L.arrivingPacket(Payload)
    foreach e in Events do               # Notify listeners
      List = getListOfListeners()
      foreach L in List do L.kpChange()
class Layer
  function arrivingPacket( )
    Event[] changes = processData()
    KPlane.changedState(changes)
```

C. Interoperability with Regular Nodes

Due to the significant number of mobile devices already deployed, MANKP-based nodes will interact with nodes that do not employ a KP, called *regular nodes*, during a significant transition period. Thus, MANKP-aware protocols will implement both KP-based information gathering and traditional methods. If no regular node is on the neighborhood of a MANKP-aware node, it will employ the MANKP approach. However, if a MANKP-aware node receives traditional update messages (e.g. a route request message), it will respond to them using regular messages, based on the information stored on MANKP. MANKP-aware nodes in contact with regular nodes will operate on compatibility mode, while other nodes communicate using KP messages. Regular nodes will discard KP messages, because no registered application is able to treat it. Further, interoperability can be achieved, however at a higher energy usage due to the use of two information dissemination paradigms.

V. Case Study: Transmission Power Control

In order to justify the use of MANKP in the management tasks of MANETs, this section describes a family of TPC-aware protocols based on MANKP. The problem of transmission power control (TPC) is to adjust the transmission power in order to minimize energy consumption in the communication. TPC involves the MAC, PHY and routing layers, because the transmission power (a PHY issue) is calculated based on measurements made on the MAC layer (the correct reception of a packet) and on the PHY layer (the reception power) [26]. Further, routing must be aware of the transmission power defined in each link in order to build minimum energy consumption routes [9].

In order to avoid cross-layering, Kawadia and Kumar proposed four TPC-aware protocols that solve the problem strictly on the routing layer [9]. Each node sends one beacon packets at each available transmission power. The ideal transmission power is determined by the smallest transmission value from the set of the recently received beacon packets. We propose to optimize this process, using one MANKP packet to calculate the ideal transmission power for all neighbors.

Suppose that node i sends a MANKP packet, and node j receives it. The PHY and MAC layers on node j will collaborate to calculate the transmission power from node i to j ($P_{i \to j}$) based on information contained on the MANKP packet, such as transmission power, and from hardware information, such as the reception power of the packet. The result of this calculation will be stored on MANKP. Next, the MAC layer on node j will embed $P_{i \to j}$ on the next MANKP packet that j sends. Hence, when node i receives the MANKP packet from j, it will obtain $P_{i \to j}$. This strategy is repeated for all nodes. Nodes will also embed routing information on MANKP packets, saving up on the transmission of routing packets. Thus, our TPC solutions exercise the following requirements of MANKP: (i)

access to hardware readings; (ii) Detailed information of the neighborhood, such as the individual transmission power of each link; (iii) multi-protocol packets.

In a layered approach, as mentioned before, each node would send one packet for each transmission power. Thus, the network would require at most $2n \times p$ packets, where n is the number of nodes and p is the number of transmission powers available. Each node will send p beacon packets, which must be replied by each node. Using MANKP, we would reduce it to $2n$ packets, as one packet per node is enough to calculate the transmission power, and another is required to diffuse this information. Clearly, the use of a KP is a huge advantage. The ideal transmission power would be calculated using the collaboration of KP-aware PHY and MAC layers, while routing would act as a consumer of this information, using it to optimize the routes.

VI. Conclusions and Future Work

This paper proposed a knowledge plane for ad hoc networks, called MANKP. Each node has a MANKP module, that stores information from all layers and from other nodes on the network in one centralized place, accessible to protocols and applications. Thus, protocols and applications may use algorithms that take into account information from other layers to improve their performance. Using transmission power control aware protocols as a case study, we made the point for the use of MANKP on MANETs showing that MANKP-aware protocols are more integrated, and thus will have a reduced energy consumption when compared to traditional approaches.

As future work, we intend to define the network-level information stored at the control plane and evaluate the gains of MANKP using simulations. We will also investigate how the correlation of this information can improve the diagnosis of important events on the network, proposing automated algorithms to identify and treat them.

Acknowledgements

We would like to thank CNPq, an organization from the Science and Technology Ministry, Brazil, for partially funding this research.

References

[1] B. Manoj and A. H. Baker, "Communication challenges in emergency response," *Communications of the ACM*, vol. 50, no. 3, pp. 51–53, 2007.

[2] V. Kumar, D. Rus, and S. Singh, "Robot and sensor networks for first responders," *IEEE Pervasive Computing*, vol. 3, no. 4, pp. 24–33, October-December 2004.

[3] V. Srivastava and M. Motani, "Cross-layer design: A survey and the road ahead," *IEEE Communications Magazine*, vol. 43, no. 12, pp. 112–119, December 2005.

[4] D. Clark, C. Partridge, J. C. Ramming, and J. T. Wroclawski, "A knowledge plane for the internet," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, 2003, pp. 3–10.

[5] D. Ferrara, L. Galluccio, A. Leonardi, G. Morabito, and S. Palazzo, "Macro: an integrated mac/routing protocol for geographic forwarding in wireless sensor networks," in *IEEE INFOCOM 2005*, vol. 3, March 2005, pp. 1770–1781.

[6] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, August 2000, pp. 243–254.

[7] S. Wu and K. S. Candan, "GPER: Geographic power efficient routing in sensor networks," in *12th IEEE International Conference on Network Protocols (ICNP)*, 2004, pp. 161–172.

[8] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proceedings of the first international conference on Embedded networked sensor systems*, 2003, pp. 14–27.

[9] V. Kawadia and P. R. Kumar, "Principles and protocols for power control in wireless ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 76–88, Jan 2005.

[10] D. Katabi, M. Handley, and C. Rohrs, "Congestion control for high bandwidth-delay product networks," in *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, 2002, pp. 89–102.

[11] V. Kawadia and P. R. Kumar, "A cautionary perspective on cross layer design," *IEEE Wireless Communications*, vol. 12, no. 1, pp. 3–11, February 2005.

[12] N. Niebert, H. Abramowicz, G. Malmgren, J. Sachs, U. Horn, C. Prehofer, and H. Karl, "Ambient networks: an architecture for communication networks beyond 3G," *IEEE Wireless Communications*, vol. 11, no. 2, pp. 14–22, April 2004.

[13] A. Manzalini and F. Zambonelli, "Towards Autonomic and Situation-Aware Communication Services: the CASCADAS Vision," in *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS)*, June 2006, pp. 383–388.

[14] J. Scott, P. Hui, J. Crowcroft, and C. Diot, "Haggle: A networking architecture designed around mobile users," in *Proceedings of the Third Annual IFIP Conference on Wireless On-demand Network Systems and Services (WONS 2006)*, January 2006.

[15] A. new components to the knowledge plane GMPLS over WDM networks, "Anna urra and eusebi calle and j. l. marzo," in *IEEE Workshop on IP Operations and Management (IPOM)*, October, pp. 82–86.

[16] A. Malatras and G. Pavlou, "Context-driven self-configuration of mobile ad hoc networks," in *Second International IFIP Workshop on Autonomic Communication (WAC)*, 2005, pp. 71–85.

[17] P. Lago, "A policy-based approach to personalization of communication over converged networks," in *Third International Workshop on Policies for Distributed Systems and Networks (POLICY)*, 2002, pp. 169–180.

[18] R. Chadha, H. Cheng, Y.-H. Cheng, J. Chiang, A. Ghetie, G. Levin, and H. Tanna, "Policy-based mobile ad hoc network management," in *Proceedings of the Fifth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY)*, 2004, pp. 35–44.

[19] A. M. Hadjiantonis, A. Malatras, and G. Pavlou, "A context-aware, policy-based framework for the management of manets," in *Proceedings of the Seventh IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY)*, Washington, DC, USA, 2006, pp. 23–34.

[20] S. Davy, K. Barrett, S. Balasubramaniam, S. van der Meer, B. Jennings, and J. Strassner, "Policy-based architecture to enable autonomic communications - a position paper," in *3rd IEEE Consumer Communications and Networking Conference (CCNC)*, Jan 2006, pp. 590–594.

[21] M. Conti, E. Gregori, and G. Turi, "A cross-layer optimization of gnutella for mobile ad hoc networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, 2005, pp. 343–354.

[22] M. Razzaque, S. Dobson, and P. Nixon, "A cross-layer architecture for autonomic communications," in *Autonomic Networking*, November 2006, pp. 25–35.

[23] R. Winter, J. Schiller, N. Nikaein, and C. Bonnet, "Crosstalk: a data dissemination-based crosslayer architecture for mobile ad-hoc networks," in *5th Workshop on Applications and Services in Wireless Networks (ASWN)*, June 2005.

[24] G. Pavlou, P. Flegkas, S. Gouveris, and A. Liotta, "On management technologies and the potential of web services," in *IEEE Communications Magazine*, vol. 42, no. 7, July 2004, pp. 58–66.

[25] E. Gamma, R. Johnson, R. Helm, and J. Vlissides, *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley, 1994.

[26] L. H. A. Correia, D. F. Macedo, A. L. dos Santos, A. A. F. Loureiro, and J. M. S. Nogueira, "Transmission power control techniques for wireless sensor networks," *Elsevier Computer Networks*, (to appear).