Sensor and Ad Hoc Wireless Networks*

(tutorial paper)

Martin Drozda
FG Simulation und Modellierung, Computer Science Dept.
Leibniz University of Hannover, Welfengarten 1, 30167 Hannover, Germany
drozda@sim.uni-hannover.de

Abstract

Sensor and ad hoc wireless networks lack a fixed infrastructure in the form of wireline, or base stations to support the communication. Instead, any participating wireless device can act as a router, when a direct communication is not possible.

Sensor and ad hoc wireless networks is an area of very active research. In this tutorial we discuss issues specific to this flavor of communication networks: MAC and routing protocols, protocol interactions, mobility models, misbehavior and misbehavior detection, and relevant graph-theoretic concepts. The current network simulation tools will also be discussed.

1 Topics Covered

Within this tutorial the following topics will be covered:

- MAC and routing protocols for ad hoc wireless networks: on-demand routing, hierarchical routing, routing based on gossiping, location based routing, power-aware MAC and routing protocols, multicast routing, secure routing.
- Cross-layer interactions and cross-layer design.
- Mobility models for ad hoc wireless networks.
- Graph-theoretic concepts for sensor and ad hoc wireless networks; distance-2 matching.
- Misbehavior in (sensor) ad hoc wireless networks and efficient solutions to misbehavior detection. Recent results in Artificial immune systems and their suitability for misbehavior detection.
- Current network simulation tools suitable for simulating (sensor) ad hoc networks. Glomosim/Qualnet, ns2, OMNeT++ and swans will be discussed.

2 Sensor and Ad Hoc Wireless Networks

Ad hoc wireless networks [24, 16] do not need any infrastructure in the form of base stations or wireline to operate. Data packets are forwarded by intermediate wireless devices until the destination is reached. Wireless devices in this setting are expected to be small and therefore battery powered.

Sensor networks are a specialized flavor of ad hoc wireless networks. A sensor network is understood to be a collection of small wireless devices (sensors) that are able to monitor environmental or physical conditions such as humidity, temperature, motion or noise. These sensors are suitably spatially distributed in the monitored area.

The self organizing feature of sensor and ad hoc wireless networks is often restated in graph theoretic framework as follows: an (sensor) ad hoc network is a net $N = (n(t), e(t))$ where $n(t), e(t)$ are the sets of nodes and edges at time t, respectively. Nodes correspond to mobile users or automated sensors that wish to communicate with each other.[1] An edge between two nodes A and B is said to exist when A is within the radio transmission range of B and vice versa. The imposed symmetry of edges is a usual assumption of many mainstream protocols. The change in the cardinality of sets $n(t), e(t)$ can be caused by the freedom that users have when they wish to switch on or switch off their communication device, or can be caused by mobility of users, signal propagation, link reliability and other factors.

Data exchange in a point-to-point (uni-cast) scenario often proceeds as follows: a node initiated data exchange leads to a route query at the network layer of the OSI pro-

*This work was supported by the German Research Foundation (DFG) under the grant no. SZ 51/24-1 (Survivable Ad Hoc Networks – SANE) and by the German Academic Exchange Service (DAAD) within the SAMANET project.

[1] We will use the term node, when a wireless device and/or a sensor is meant.

tocol stack.[2] A routing protocol at that layer attempts to find a route to the data exchange destination. This request may result in a path of non-unit length. This means that a data packet in order to reach the destination has to rely on successive forwarding by intermediate nodes on the path.

3 Protocol Stack

Sensor and ad hoc wireless networks have specific needs that must be considered when designing tailored communications protocols. Wireless devices are expected to have limited power supply. In case of sensor networks, it is often assumed that their maintenance will be sporadic and that their computational capabilities are *extremely limited*. For example, sensors marketed by Crossbow Inc. [8] use a low power 8-bit microprocessor with peak computational throughput of 16 MIPS[3], the program memory of 128 kB and the storage memory of 512 kB (Mica2 sensors).

The research of communications protocols evolved in several directions. Novel MAC (Medium access control) protocols such as PAMAS [29] or S-MAC [33] acknowledge the fact that a big portion of the available battery power is wasted when nodes participate in medium contention resolution. Therefore nodes got the freedom to turn themselves off in idle periods and a sleep-wake-up schedule mechanism has been proposed in order to minimize battery power consumption. The level of medium contention, that a network can face, can be expressed by the distance-2 matching [18]. The *distance-2 matching* is defined as follows: given a graph $G(V, E)$, find a set of edges $E' \subseteq E$ such that no two edges in E' are connected by another edge in E. An interesting problem is to find a maximum distance-2 matching; this problem is known to be NP-complete. This measure assumes that the contention resolution mechanism uses an RTS-CTS-DATA handshake[4] in order to reserve local medium for data transmission. In Figure 1, node A signals by sending an RTS his readiness to transmit a data packet. If node B is not busy with another transmission, it replies with a CTS packets; this packet is overheard by both A and C. Node C learns this way that B is going to receive a data packet and postpones its own possible data transmission. Under this mechanism, if any other node in the network wishes to transmit a data packet, it must be at least two edges away from B, or the data transmission must be delayed. Distance-2 matching can be suitably combined with other graph-theoretic concepts such as network diameter in order to characterize the given network's capability to transmit data packets [4].

In the area of routing protocols design it was acknowledged that routing protocols, that attempt to compute a path for each possible source-destination pair, are unable to keep up with the ever changing topology of ad hoc wireless networks. Such pro-active routing protocols do not consider whether a given routing path will ever be used and therefore consume an undesired share of the available bandwidth. As a consequence on-demand routing protocols that search for routes only when they are needed have been proposed. Examples of such protocols are DSR [15] and AODV [25]. When it is desired to establish a unicast (one-to-one) connection between a source and a destination, these protocols flood the network with Route request (RREQ) control packets that either travel all the way to the destination, or to the closest node that knows the path to the destination. These nodes then reply with a Route reply (RREP) packet that traverses the net in the opposite direction back to the source. In order to be able to react to broken links caused by movement or worsened link quality, these protocols also include mechanisms that are aimed at maintaining already found paths. An interesting example of a protocol that localizes the algorithmic reaction, should a path get broken, is TORA [21]. This protocol computes a directed acyclic graph rooted at the destination. Nodes are assigned weights that get recomputed if a node no longer has a directed path to the destination.

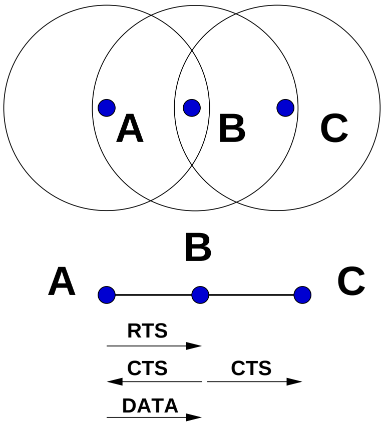

Figure 1. An RTS-CTS-DATA handshake. Node A sends RTS, node B sends CTS that is overheard by both A and C. Finally, A can send DATA. The circles approximate radio propagation.

Recently, many researchers advocated use of the Global Positioning System (GPS) in efficient routing. Based on GPS coordinates, LAR [17] computes a zone within which the destination node is believed to be located. This approach decreases routing overhead and communication complexity. The forwarding scheme of LAR is similar to DSR, however, the intermediate nodes are allowed to forward route request packets only to neighbors in the zone. Similar in design are

[2] The ISO Model of Architecture for Open Systems Interconnection.

[3] MIPS = million instructions per second.

[4] RTS = Ready to send, CTS = Clear to send, DATA = data packet.

DREAM [5], Fisheye routing (FSR) [22] and Zone routing (ZRP) [12] protocols. DREAM is based on the observation that "the greater the distance of two nodes, the slower they appear to be moving with respect to each other". In FSR the quality of routing decreases as the distance to destination increases. ZRP uses both pro-active and reactive routing; the flavor of routing changes with the distance to the source.

In hierarchical routing [23], nodes form clusters with an elected cluster head. Each cluster head is responsible for communication between nodes that lie in different clusters. Cluster heads use up more battery power than ordinary nodes, therefore a mechanism for cluster head rotation has to be available [13].

On-demand routing protocols such as DSR or AODV use broadcast for propagating the RREQ control packets around the network. This means that each nodes sends the first received RREQ to all its neighbors with probability equal 1.0. In gossiping based protocols [11], this probability is strictly smaller than 1.0. Since the available bandwidth in sensor and ad hoc networks is scarce, gossiping is an interesting alternative to broadcast.

The need for multicast communication emerges when one node wishes to send an identical message to a group of other nodes. It is usually assumed that nodes have the freedom to join and leave a multicast group. This decision is based on their desire to be either able to receive a specific type of messages or to be able to access a group of nodes with specific interests. A representative of multicast routing protocols is Multicast AODV (MAODV) [28]. This protocol builds and maintains a multicast tree whose members cooperate in forwarding multicast messages.

Since it is possible for any node that forwards a routing control packet to modify its headers, there has been an increased interest for protocols that would allow for secure broadcast and node authentication. The TESLA protocol [26] uses a hash-chain mechanism with a delayed secret key publish mechanism to achieve that.

Protocols at any layer of the OSI protocol stack have been traditionally designed in isolation. This has become a major source of throughput deterioration in (sensor) ad hoc wireless networks due to bandwidth limitations (as compared to wired networks). Therefore researchers have attempted to identify the extent of interlayer interactions [6] and to propose new solutions to this problem [30].

4 Mobility Models

Since nodes that form an ad hoc wireless network are expected to move freely, there has been a multitude of mobility models [7] introduced. These models control either the movement of individual nodes or the movement of groups of nodes. A well-known mobility model is the *Random waypoint model*. In this model, nodes move from the current position to a new randomly generated position at a predetermined speed. After reaching the new destination a new random position is computed. Nodes pause at the current position for a time period t before moving to the new random position. Other mobility models, commonly used, are: Random walk model, Random direction model, Gaussian-Markov model or Nomadic community model; see [7]. A special class of mobility models form those that are based on realistic traffic in an urban environment. Transims is an agent-based simulation system capable of simulating the second-by-second movements of every person and every vehicle through the transportation network of a large metropolitan area; see the Transims project [31].

5 Misbehavior and Misbehavior Detection

The battery power and computational capabilities of mobile devices (sensors) are limited. Some nodes might decide that in order to preserve their own battery power or to lower the computational load, they will not participate in mechanisms prescribed by communications protocols. These nodes might decide to drop data packets, change headers of control or data packets, increase or decrease their relative importance, create virtual nodes, selectively forward control or data packets, or attempt to skew the network topology [10]. An emerging solution to these problems are Artificial immune systems (AIS) [14, 9] that are motivated by the mechanism of the Human immune system (HIS). The HIS is a rather complicated mechanism that is able to protect humans against an amazing set of extraneous attacks. This system is remarkably efficient, most of the time, in discriminating between *self* and *non-self* antigens.[5] A non-self antigen is anything that can initiate an immune response; examples are a virus, bacteria, or splinter. The opposite to non-self antigens are self antigens; self antigens are human organism's own cells.

The process of T-cells maturation in thymus is used as an inspiration for learning in AIS. The creation of T-cells (detectors) in thymus is a result of a pseudo-random process. After a T-cell is created (see Figure 2),[6] it undergoes a censoring process called *negative selection*. During negative selection T-cells that bind self are destroyed. Remaining T-cells are introduced into the body. The recognition of non-self is then done by simply comparing T-cells that survived negative selection with a suspected non-self. This process is depicted in Figure 3. It is possible that the self set is incomplete, while a T-cell matures (tolerization period) in the thymus. This leads to producing T-cells that should have been removed from the thymus and can cause an autoimmune reaction, i.e. it leads to *false positives*.

To apply the above described learning process to (sensor) ad hoc wireless networks, it is necessary that each node

[5] Self and non-self in short.

[6] T-cells (detectors) are here represented as (binary) strings.

observes and evaluates data and control traffic that he forwards or that he overhears in the neighborhood. This traffic can be characterized by performance measures; it is important that these measures are easy to compute locally. Examples of such performance measures are e.g. the number of complete RTS-CTS-DATA handshakes, pattern of the routing control packets that a node forwards, or the observed willingness of neighboring nodes to forward data packets. These performance measures are often represented as bitstrings. Subsequently, they get concatenated (a single bitstring is created) and become subject to the negative selection process depicted in Figure 2. The drawback of this approach is that a misbehavior-free period is necessary (the set of self strings must be created). To solve this problem, researchers suggested the use of a danger signal [1]. Danger signal is a simple form of feedback that helps classify a detected anomaly as misbehavior.

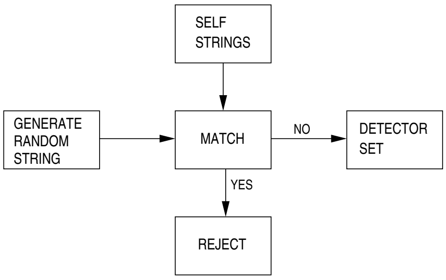

Figure 2. Detector generation by random-generate-and-test process. Only strings that do not match anything self become detectors.

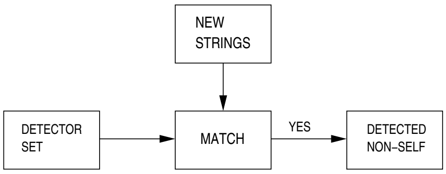

Figure 3. Recognizing non-self is done by matching detectors with suspected non-self strings.

6 Network Simulation Tools

A detailed simulation of an (sensor) ad hoc wireless network requires that key protocols and procedures get implemented in a programming (simulation oriented) language. To increase efficiency of simulations, researchers rely on one of the available network simulation tools. ns2 [19] is a tool that allows for an easy configuration through scripting in the OTcl language. Glomosim [2] is based on the parallel simulation language Parsec; the simulation parameters are entered through a simple text file. Qualnet [27] is the commercial version of Glomosim. swans [3] and OMNeT++ [20] are simulators that recently gained on popularity. A non-exhaustive comparison of these simulator tools is in Table 1. In general, it can be stated that ns2 and Qualnet offer the largest set of functionality in form of available models of protocols and support. The advantage of swans is that it is a Java based simulator. The authors of swans claim in [3] that their simulator performs well in comparison to Glomosim and ns2 when the simulation real time and memory consumption is considered. ns2 seems not to scale well beyond several hundred nodes. Glomosim has not been further developed since 2001 when Qualnet entered the market.

In [32] the authors claim to develop a simulator tool that scales up to roughly one million nodes. They demonstrated the capabilities of their tool on a realistic topology resembling the city of Los Angeles.

7 Conclusions

Sensor and ad hoc wireless networks remain an area of a very intense research activity. One of the important research challenges is cross-layer protocol design. *Mega protocols* optimized over several layers of the OSI protocol stack will hopefully be able to suppress interaction effects [6] that lead to performance deterioration.

The capability of large scale simulations is important in several ways. Currently, the degree of robustness of large ad hoc (sensor) networks is not well understood [4]. It is not clear how certain adverse effects can propagate through a network. It is therefore also not clear what impact would a possible misbehavior of nodes have on the overall performance. Consequently, it is not clear how to impose a higher degree of survivability on ad hoc (sensor) networks.

References

[1] U. Aickelin, P. Bentley, S. Cayzer, J. Kim, and J. McLeod. Danger Theory: The Link between AIS and IDS? *Proc. of the Second Internation Conference on Artificial Immune Systems (ICARIS-03)*, pages 147–155, 2003.

[2] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, and M. Gerla. GloMoSim: A Scalable Network Simulation Environment. *UCLA Computer Science Department Technical Report*, 990027, 1999.

[3] R. Barr. SWANS-Scalable Wireless Ad hoc Network Simulator Users Guide, 2004. `jist.ece.cornell.edu`.

[4] C. Barrett, M. Drozda, D. Engelhart, V. Kumar, M. Marathe, M. Morin, S. Ravi, and J. Smith. Understanding protocol performance and robustness of ad hoc networks through

Simulator	**Glomosim 2.03**	**Qualnet 4.0**	**ns2 2.31**	**swans 1.0.6**	**OMNeT++ 3.3**
Platform	Windows, Linux, Unix	Windows, Linux, Unix	Windows, Linux, Unix	All with Java virtual machine	Windows, Linux, Unix
Language	Parsec	Parsec	OTcl, C++	Java	C++, NED
Availability	Free for academic and non-profit use	Commercial with university program	Free for academic and non-profit use	Free for academic and non-profit use	Free for academic and non-profit use

Table 1. Network simulation tools: basic features.

structural analysis. *IEEE International Conference on Wireless And Mobile Computing, Networking And Communications (WiMob'2005)*, 3:65–72, 2005.

[5] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A distance routing effect algorithm for mobility (DREAM). *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, pages 76–84, 1998.

[6] C. Barrett, M. Drozda, M.V. Marathe and A. Marathe. Characterizing the interaction between routing and MAC protocols in ad-hoc networks. *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and computing*, pages 92–103, 2002.

[7] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5):483–502, 2002.

[8] Crossbow Technologies Inc. `www.xbow.com`.

[9] M. Drozda, S. Schaust, and H. Szczerbicka. Is AIS Based Misbehavior Detection Suitable for Wireless Sensor Networks? *Proc. IEEE Wireless Communications and Networking Conference (WCNC'07)*, 2007.

[10] M. Drozda and H. Szczerbicka. Artificial immune systems: Survey and applications in ad hoc wireless networks. *Proc. 2006 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'06)*, pages 485–492, 2006.

[11] Z. Haas, J. Halpern, and L. Li. Gossip-based ad hoc routing. *IEEE/ACM Transactions on Networking (TON)*, 14(3):479–491, 2006.

[12] Z. J. Haas and M. R. Pearlman. ZRP: a hybrid framework for routing in ad hoc networks. In *Ad hoc networking* [24], pages 221–253.

[13] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, pages 1–10, 2000.

[14] S. Hofmeyr and S. Forrest. Immunity by design: An artificial immune system. *Proceedings of the Genetic and Evolutionary Computation Conference*, 2:1289–1296, 1999.

[15] D. Johnson and D. Maltz. Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 353:153–181, 1996.

[16] H. Karl and A. Willig. *Protocols and Architectures for Wireless Sensor Networks.* John Wiley and Sons, 2005.

[17] Y. Ko and N. Vaidya. Location-Aided Routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6(4):307–321, 2000.

[18] V. Kumar, M. Marathe, S. Parthasarathy, and A. Srinivasan. Algorithmic aspects of capacity in wireless networks. *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 133–144, 2005.

[19] ns-2 Network Simulator. `www.isi.edu/nsnam/ns`.

[20] OMNeT++ Community Web Site. `www.omnetpp.org`.

[21] V. Park and M. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. *Proc. IEEE INFOCOM*, 3:1405–1413, 1997.

[22] G. Pei, M. Gerla, and T. Chen. Fisheye state routing: a routing scheme for ad hoc wireless networks. *IEEE International Conference on Communications (ICC'00)*, pages 70–74, 2000.

[23] G. Pei, M. Gerla, X. Hong, and C. Chiang. A wireless hierarchical routing protocol with group mobility. *IEEE Wireless Communications and Networking Conference (WCNC'99)*, pages 1538–1542, 1999.

[24] C. Perkins. *Ad hoc networking*. Addison-Wesley Longman Publishing Co., 2001.

[25] C. E. Perkins and E. M. Royer. Ad hoc On-Demand Distance Vector Routing. *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, 1999.

[26] A. Perrig, R. Canetti, J. Tygar, and D. Song. The TESLA Broadcast Authentication Protocol. *RSA CryptoBytes*, 5(2):2–13, 2002.

[27] Qualnet Simulator. `www.scalable-networks.com`.

[28] E. Royer and C. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 207–218, 1999.

[29] S. Singh and C. Raghavendra. PAMAS–power aware multi-access protocol with signalling for ad hoc networks. *ACM SIGCOMM Computer Communication Review*, 28(3):5–26, 1998.

[30] V. Srivastava and M. Motani. Cross-layer design: a survey and the road ahead. *Communications Magazine, IEEE*, 43(12):112–119, 2005.

[31] Transims project. `transims.tsasa.lanl.gov`.

[32] R. Waupotitsch, S. Eidenbenz, J. P. Smith, and L. Kroc. Multi-scale integrated information and telecommunications system (miits): first results from a large-scale end-to-end network simulator. *Proceedings of the 37th Winter simulation conference*, pages 2132–2139, 2006.

[33] W. Ye and J. Heidemann. Medium Access Control in Wireless Sensor Networks. *Wireless Sensor Networks*, pages 73–91, 2004.