Journal of Computer Applications ISSN: 0974 – 1925,
Volume-5, Issue EICA2012-5, February 10, 2012

Implementing a Cooperative MAC Protocol for Wireless LANs

Mahalakshmi M
PG Scholor,Department of CSE,
SSM College of Engineering,
Komarapalayam, India
E-mail: mahalakshmi537@gmail.com.

Chitra P
Heads of Department
Department of CSE,
SSM College of Engineering,
Komarapalayam, India.

Abstract-We evaluate the performance of cooperative transmission, where nodes in a sending cluster are synchronized to communicate a packet to nodes in a receiving cluster. In our communication model, the power of the received signal at each node of the receiving cluster is a sum of the powers of the transmitted independent signals of the nodes in the sending cluster. The increased power of the received signal, vis-à-vis the traditional single-node-to-single-node communication, leads to overall saving in network energy and to end-to-end robustness to data loss. We propose an energy-efficient cooperative protocol, and we analyze the robustness of the protocol to data packet loss. When the nodes are placed on a grid, it reduces the probability of failure to deliver a packet to destination.

In Energy Efficient Protocol for Cooperative networks, transmitting and receiving nodes recruit neighboring nodes to assist in communication. We model a cooperative transmission link in wireless networks as a transmitter cluster and a receiver cluster. Up to 80% in energy savings can be achieved for a grid topology, while for random node placement our cooperative protocol can save up to 40% in energy consumption relative to the other protocols. The reduction in error rate and the energy savings translate into increased lifetime of cooperative sensor networks.

Keywords: *Clustering, Cooperative transmission protocol, Sensor networks, High load operation.*

I. INTRODUCTION

Wireless Sensor Network (WSN) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants [14]. The development of wireless sensor In this paper, we investigate how threat modeling can be used as foundations for the specification of security requirements (SSR). It also discusses the importance of implementing web application security at design time. Threat modeling is a process to ensure that application security is implemented at design time. Threat modeling is a structured activity for identifying and evaluating application threats and vulnerabilities. This How To presents a question-driven approach to threat modeling that can help us to identifies security design problems early in the application design process. This approach allows you to quickly create a basic threat model for your web application scenario. Then you can use this threat model to help refine your application's design early and for communication among team members. The threat modeling approach is presented here focuses on identifying and addressing vulnerabilities. The security objectives, threats, and attacks that we identify in the early steps of the activity are the scoping mechanisms designed to help you find vulnerabilities in our

web application. We can use the identified vulnerabilities to help shape our design and direct and scope your web application security testing. Networks was motivated by military applications such as battlefield surveillance. They are now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control. In addition to one or more sensors, each node in a sensor network is typically equipped with a radio transceiver or other wireless communications device, a small microcontroller, and an energy source, usually a battery.

A sensor node might vary in size from that of a shoebox down to the size of a grain of dust although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor node is similarly variable, ranging from hundreds of dollars to a few pennies, depending on the size of the sensor network and the complexity required of individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and bandwidth. A sensor network normally constitutes a wireless ad-hoc network, meaning that each sensor supports a multi-hop routing algorithm (several nodes may forward data packets to the base station).

Sensor networks have emerged as a promising tool for monitoring (and possibly actuating) the physical worlds, utilizing self-organizing networks of battery-powered wireless sensors that can sense, process and communicate. In sensor networks, energy is a critical resource, while applications exhibit a limited set of characteristics. The requirements and limitations of sensor networks make their architecture and protocols both challenging and divergent from the needs of traditional Internet architecture.

The basic goals of a WSN are to:

i. Determine the value of physical variables at a given location,
ii. Classify a detected object, and
iii. Track an object.

A sensor network is a network of many tiny disposable low power devices, called nodes, which are spatially distributed in order to perform an application-oriented global task. These nodes form network by communicating with each other either directly or through other nodes. One or more nodes among them will serve as sink(s) that are capable of communicating with the user either directly or through the existing wired networks. The primary component of the network is the sensor, essential for monitoring real world physical conditions such as sound, temperature, humidity, intensity, vibration, pressure, motion, pollutants etc. at different locations. The tiny sensor nodes, which consist of sensing, on board processor for data processing, and communicating components, leverage the idea of sensor networks based on collaborative effort of a large number of nodes. Figure 1 shows the structural view of a sensor network in which sensor nodes are shown as small circles.

Each node typically consists of the four components: sensor unit, central processing unit (CPU), power unit, and Communication unit. They are assigned with different tasks. The sensor unit and ADC (Analog to Digital Converter). The sensor unit is responsible for collecting information as the ADC requests, and returning the analog data it sensed.

ADC is a translator that tells the CPU what the sensor unit has sensed, and also informs the sensor unit what to do. Communication unit is tasked to receive command or query from and

transmit the data from CPU to the outside world. CPU is the most complex unit. It interprets the command or query to ADC, monitors and controls power if necessary, processes received data, computes the next hop to the sink, etc.

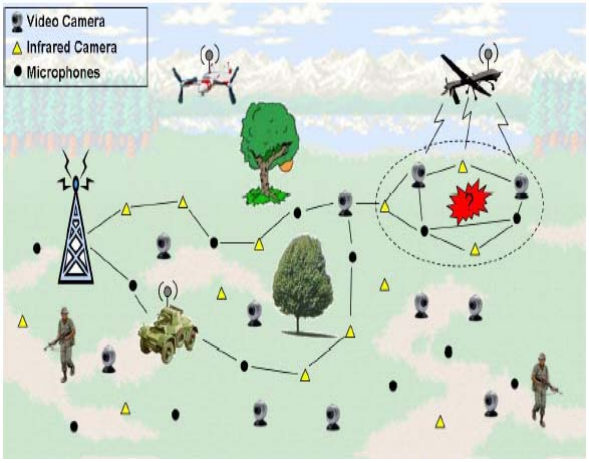

Figure1: Structural View of Sensor Network

Power unit supplies power to sensor unit, processing unit and communication unit. Each node may also consist of the two optional components namely Location finding system and mobilizer. If the user requires the knowledge of location with high accuracy then the node should pusses Location finding system and mobilizer may be needed to move sensor nodes when it is required to carry out the assigned tasks.

Instead of sending the raw data to the nodes responsible for the fusion, sensor nodes use their processing abilities to locally carry out simple computations and transmit only the required and partially processed data. The sensor nodes not only collect useful information such as sound, temperature, light etc., they also play a role of the router by communicating through wireless channels under battery-constraints.

Sensor network nodes are limited with respect to energy supply, restricted computational capacity and communication bandwidth. The ideal wireless sensor is networked and scalable , fault tolerance, consume very little power, smart and software programmable, efficient, capable of fast data acquisition, reliable and accurate over long term, cost little to purchase and required no real maintenance.

II. ROUTING PROTOCOLS

A routing protocol is a protocol that specifies how routers communicate with each other, disseminating information that enables them to select routes between any two nodes on a computer network, the choice of route being done by routing algorithm.

2.1 Ad-Hoc on Demand Distance Vector (AODV)

The Ad Hoc On-demand Distance Vector Routing (AODV)[11] protocol is a reactive unicast routing protocol for mobile ad hoc networks. As a reactive routing protocol, AODV only needs to maintain the routing information about the active paths. In AODV, routing

information is maintained in routing tables at nodes. Every mobile node keeps a next-hop routing table, which contains the destinations to which it currently has a route.

A routing table entry expires if it has not been used or reactivated for a pre-specified expiration time. Moreover, AODV adopts the destination sequence number technique used by Destination Sequence Distance Vector (DSDV in an on-demand way. In AODV [11], [24], when a source node wants to send packets to the destination but no route is available, it initiates a route discovery operation. In the route discovery operation, the source broadcasts route request (RREQ) packets.

A RREQ includes addresses of the source and the destination, the broadcast ID, which is used as its identifier, the last seen sequence number of the destination as well as the source node's sequence number. Sequence numbers are important to ensure loop-free and up-to-date routes. To reduce the flooding overhead, a node discards RREQs that it has seen before and the expanding ring search algorithm is used in route discovery operation.

In AODV, each node maintains a cache to keep track of RREQs it has received. The cache also stores the path back to each RREQ originator. When the destination or a node that has a route to the destination receives the RREQ, it checks the destination sequence numbers it currently knows and the one specified in the RREQ. The redundant RREP packets or RREP packets with lower destination sequence number will be dropped. In AODV, a node uses hello messages to notify its existence to its neighbors. Therefore, the link status to the next hop in an active route can be monitored. When a node discovers a link disconnection, it broadcasts a ROUTE_ERROR packet to its neighbors, which in turn propagates the ROUTE_ERROR packet towards nodes whose routes may be affected by the disconnected link. Then, the affected source can re-initiate a route discovery operation if the route is still needed.

2.2 Dynamic source routing (DSR)

Dynamic Source Routing (DSR) is a reactive unicast routing protocol that utilizes source routing algorithm [19-20]. In source routing algorithm, each data packet contains complete routing information to reach its destination.

Otherwise, the source node initiates a route discovery operation by broadcasting route request packets. A route request packet contains addresses of both the source and the destination and a unique number to identify the request. Receiving a route request packet, a node checks its route cache. If the node doesn't have routing information for the requested destination, it appends its own address to the route record field of the route request packet. Then, the request packet is forwarded to its neighbors.

To limit the communication overhead of route request packets, a node processes route request packets that both it has notseen before and its address is not presented in the route record field. If the route request packet reaches the destination or an intermediate node has routing information to the destination, a route reply packet is generated. When the route reply packet is generated by the destination, it comprises addresses of nodes that have been traversed by the route request packet.

The second possibility is that the network has symmetric (bi- directional) links. The route reply packet is sent using the collected routing information in the route record field, but in a reverse order.

In the last case, there exists asymmetric (uni- directional) links and a new route discovery procedure is initiated to the source. The discovered route is piggybacked in the route request packet. In DSR, when the data link layer detects a link disconnection, a ROUTE_ERROR packet is sent backward to the source. After receiving the ROUTE_ERROR packet, the source node initiates another route discovery operation.

2.3Destination Sequence Distance Vector (DSDV)

The Destination Sequence Distance Vector (DSDV) is a proactive unicast mobile ad hoc network routing protocol, DSDV is also based on the traditional Bellman-Ford algorithm.
In routing tables of DSDV, an entry stores the next hop towards a destination, the cost metric for the routing path to the destination and a destination sequence number that is created by the destination. Sequence numbers are used in DSDV to distinguish stale routes from fresh ones and avoid formation of route loops. The route updates of DSDV can be either time-driven or event-driven.

Every node periodically transmits updates including its routing information to its immediate neighbors. While a significant change occurs from the last update, a node can transmit its changed routing table in an event-triggered style. Moreover, the DSDV has two ways when sending routing table updates.

One is "full dump" update type and the full routing table is included inside the update. A "full dump" update could span many packets. An incremental update contains only those entries that with metric have been changed since the last update is sent. Additionally, the incremental update fits in one packet.

II. RECRUIT AND TRANSMITTING PHASE

The example in Figure.3(a)-(f)demonstrates the operation of the "recruiting-and-transmitting" phase. In the current hop, node2 is the sending cluster head and has a packet to be sent to node5. Node2 sends are quest-to-recruit RR) packet to node5 [Fig.3(a)],causing node5 to start the formation of the receiving cluster, with node5 as the cluster head. From the routing phase, node5 knows that then ext-hop node is node8. Node5 broadcasts to its neighbors a recruit (REC) packet [Fig.3(b)]. The REC packet contains: the id of the previous node(2), the id of then extnode (8), and the maximum time to respond, denoted as T_r.

Each node that receives the REC packet, which we call *potential recruits* (nodes4and6inour example), computes the sum of the link costs of the following two links: a link from the sending cluster head to itself (the *receiving link*) and a link from itselftothenextnode, suchas the receiving cluster head or the sink node (the *sending link*).In our example, node4 computes the sums of the energy costs of the links (2,4) and (4,8), i.e., $C_{2,4}C_{4,8}$, while node6 computes the sum of the energy costs of the links (2,6) and (6,8),i.e., $C_{2,6}+C_{6,8}$.

A *potential all recruit* replies to the REC packet it has grant (GR) packet that contains the computed sum [Figure.3(c)] after a random back off time drawn uniformly from(0,). The GR packets inform the cluster head that the nodes are available to cooperate in receiving on the current hop andinsendingonthenexthop.

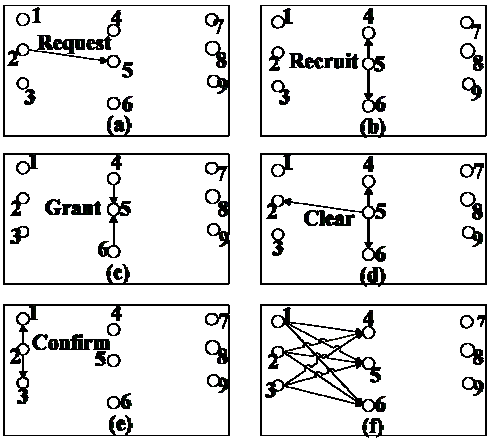

Figure3:ExampleoftheRecruitingPhaseOperation.(a)Request-to-recruit(RR)packet. (b)Recruit(REC)packet.(c)Grant(GR)packet.(d)Clear(CL)packet.(e)Confirm(CF)packet.(f) Transmission of the datapacket.

III. COOPERATIVE TRANSMISSION PROTOCOL

Let the nodes in the cluster be indexed from 0 to m-1. We denote the transmission pattern of nodes in a sending cluster by a binary representation $b_{m-1}\ldots b_1 b_0$according to which node j transmits if b_j and does not transmit if b_j. A node does not transmit when it receives a packet in error from the previous hop. We denote the reception pattern of nodes in a receiving cluster by a binary representation$b_{m-1}\ldots b_1 b_0$ according to which node$j$correctly receives the packet if b_j and receives the packet in error if b_j. Forexample, for m = 4, the binary representation of 1010 of the sending cluster and the binary representation of 0101 of the receiving cluster means that nodes 1 and 3 in the sending cluster transmit the packet, while in the receiving cluster nodes 0 and 2 correctly receive the packet and nodes 1 and 3 in correctly receive the packet. Let g_I^Jbe the probability that nodes with binary representation transmit a packet of length bits to nodes with binary representation $b_{m-1}\ldots b_1 b_0$ across a single hop, and let be the SNR of the received signal at node jThen

$$\text{ER} = f\left(\text{SNR}_j, \sum_{i=0} u_i\right)$$

$$g_I^J = \prod_{j=0}^{m-1} [(1 \quad b_j)(1 \quad (1 \quad \text{BER})^L) + b_j(1 \quad \text{BER})^L].$$

Let vector $v(i)$ be the binary representation of integer. We define:$g_{v(0)}^J v(0)$. Let be the Probability that a packet reaches the kth hop to nodes with binary representation, given that atleast one copy reaches hop$k$, then

$$A_{Jk}=\sum_{I=1}^{2^m-1} g_{v(I)}^{J} A_{v(I)k-1}$$

Now, let B_{CwR}^{h} be the probability of failure of a packet to reach any node by the hth hop

$$B_{CwR}^{h}=\sum_{k=1}^{h} A_{v(0)k}.$$

IV. EXPECTED OUTCOMES

Number of packet transmission is increased because of the classical route from a source node to a sink node is replaced with a multihop cooperative path, and the classical point-to-point communication is replaced with many-to-many cooperative communication. Minimizes the energy consumption, The increased power of the received signal, vis-à-vis the traditional single-node-to-single-node communication, leads to overall saving in network energy and to end-to-end robustness to data loss. It increases the transmission reliability and also increases the lifetime of the cooperative sensor network.

V. CONCLUSION

Cooperative transmission, where nodes in a sending cluster are synchronized to communicate a packet to nodes in a receiving cluster. In our communication model, the power of the received signal at each node of the receiving cluster is a sum of the powers of the transmitted independent signals of the nodes in the sending cluster. The increased power of the received signal, vis-à-vis the traditional single-node-to-single-node communication, leads to overall saving in network energy and to end-to-end robustness to data loss. Weighted load balanced routing selects a routing path by maximizing the weight among the feasible paths. There are three parameters in WLBR that are used to calculate the weight of the feasible path: the aggregate interface queue length, the route energy, and the hop count. Route selection is based on the weight value of each feasible path. In a feasible path, the higher the weight value, the higher is its suitability for traffic distribution to forming a cluster.

REFERENCES

1. Khandani, J. Abounadi, E. Modiano, and L. Zheng, “Cooperative routing in static wireless networks,” IEEE Trans. Commun., vol. 55, no. 11, pp. 2185–2192, Nov. 2007.
2. N. Shankar, C. Chun-Ting, and M. Ghosh, “Cooperative communication MAC (CMAC)—A new MAC protocol for next generation wireless LANs,” in Proc. IEEE Int. Conf.WirelessNetw., Commun., Mobile Comput., Maui, HI, Jul. 2005, vol. 1, pp. 1–6.
3. T. Korakis, S. Narayanan, A. Bagri, and S. Panwar, “Implementing a cooperative MAC protocol for wireless LANs,” in Proc. IEEE ICC, Istanbul, Turkey, Jun. 2006, vol. 10, pp. 4805–4810.
4. Chou, J. Yang, and D. Wang, “Cooperative MAC protocol with automatic relay selection in distributed wireless networks,” in Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops, White Plains, NY, Mar. 2007, pp. 526–531.

5. J. Mirkovic, G. Orfanos, H. Reumerman, and D. Denteneer, “A MAC protocol for MIMO based IEEE 802.11 wireless local area networks,” in Proc. IEEE WCNC, Hong Kong, Mar. 2007, pp. 2131–2136.
6. Sendonaris .A, E. Erkip, and B. Aazhang, “User cooperation diversity Part I: System description,” IEEE Trans. Commun., vol. 51, no. 11, pp. 1927–1938, Nov. 2003.
7. Sendonaris A, E. Erkip, and B. Aazhang, “User cooperation diversity Part II: Implementation aspects and performance analysis,” IEEE Trans. Commun., vol. 51, no. 11, pp. 1939–1948, Nov. 2003.
8. J. Laneman, D. Tse, and G.Wornell, “Cooperative diversity in wireless networks: Efficient protocols and outage behavior,” IEEE Trans.Inf. Theory, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.
9. [9] A. Stefanov and E. Erkip, “Cooperative information transmission in wireless networks,” in Proc. Asian–Eur. Workshop Inf. Theory, Breisach, Germany, Jun. 2002, pp. 90–93.
10. T. Hunter and A. Nosratinia, “Diversity through coded cooperation,” IEEE Trans. Wireless Commun., vol. 5, no. 2, pp. 283–289, Feb. 2006.
11. J. Laneman and G. Wornell, “Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks,” IEEE Trans. Inf. Theory, vol. 49, no. 10, pp. 2415–2425, Oct. 2003.
12. P. Herhold, E. Zimmermann, and G. Fettweis, “Cooperative multi-hop transmission in wireless networks,” Comput. Netw., vol. 49, no. 3, pp. 299–324, Oct. 2005.
13. Nosratinia A, T. E. Hunter, and A. Hedayat, “Cooperative communication in wireless networks,” IEEE Commun. Mag., vol. 42, no. 10, pp. 74–80, Oct. 2004.
14. H. Shen and S. Kalyanaraman, “Asynchronous cooperative MIMO communication,” in Proc. IEEE WiOpt, Limassol, Cyprus, Apr. 2007, pp. 1–9.
15. D. Hoang and R. Iltis, “An efficient MAC protocol for MIMO-OFDM ad hoc networks,” in Proc. IEEE Asilomar Conf. Signals, Syst. Comput., Pacific Grove, CA, Oct. 2006, pp. 814–818.
16. K. Sundaresan, R. Sivakumar, M. Ingram, and T. Chang, “A fair medium access control protocol for ad-hoc networks with MIMO links,” in Proc. IEEE INFOCOM, Hong Kong, Mar. 2004, vol. 4, pp. 2559–2570.
17. J. E. Kleider, G. Maalouli, and X. Ma, “Timing synchronization in distributed mobile MISO Rayleigh fading channels,” in Proc. IEEE MILCOM, Orlando, FL, Oct. 2007, pp. 1–7.
18. Y. Yuan, M. Chen, and T. Kwon, “A novel cluster-based cooperative MIMO scheme for multi-hop wireless sensor networks,” EURASIP J. Wireless Commun. Netw., vol. 2006, no. 2, pp. 38–38, Apr. 2006.
19. W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocols for wireless microsensor networks,” inProc. IEEE Int. Conf. Syst. Sci., Jan. 2000.
20. C. Perkins and E. Royer, “Ad-hoc on-demand distance vector routing,” in Proc. IEEE WMCSA, New Orleans, LA, Feb. 1999, pp. 90–100.