Proceedings of the World Congress on Engineering 2008 Vol III
WCE 2008, July 2 - 4, 2008, London, U.K.

SV's Energy Efficient Network Design for Real Time Wireless Networks for Unmanned Blimp

Sandip Vijay*, S.C.Sharma

***Abstract*—Current development in multidisciplinary activities provides real time access to mechtronics systems based on wireless networks. Blimp Project is such an approach in this multidisciplinary area emphasizes the integration of hands-on experience and theoretic thinking in engineering education by *college of engineering*, a constituent of *Rowan University*. Blimp Project covers piece of wireless communication. This paper proposes the possible development of Real Time Wireless Sensor Ad-hoc Network for Blimp project to build a computer-controlled unmanned blimp. New network topologies for wireless real-time networks leading to enhanced performance and new applications. The first implementation of the gateway allows the decentralized collection of data, the transmission from the sensor to the gateway via Bluetooth, and the transmission from the gateway to the central server via GSM/GPRS.**

***Index Terms*—Blimp, SVEEN (Sandip Vijay's Energy Efficient Network) , Ad-hoc Networks, geographic forwarding.**

I. INTRODUCTION

Highlight In the initial design of the blimp, RU used small portion of communication engineering. In their initial design step the image obtained by the on-board camera was sent out via a 2.4 GHz wireless video transmitter. A video receiver would capture the video stream, then send it to the computer to display it on the screen, and if necessary, to conduct image processing. The digital control signal generated by the computer was converted to pulse-width-module (PMW) signal and broadcasted with an R/C radio controller. The R/C receiver in the gondola then decoded the signal that would control the propeller and servo of the blimp. At the second phase, RU introduced sonar as the second sensor of the blimp to help the blimp to maintain the height. This further replaced the R/C controller with a pair of the Basic Stamp with radio frequency transceiver. See Figure 1 and 2. The transceiver pair enabled us to send signals both to and from the on-board equipment, and provided extra capability for future enhancement of the blimp. For example, we could install sonar for backup motion or a pair of microphone and speaker for communication.

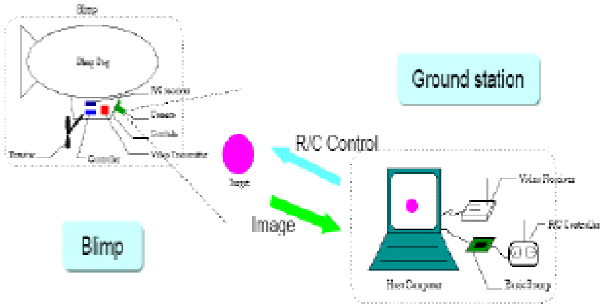


Fig.1 Initial Design of blimp [1]

In the second iteration of the Blimp project sensors and sonar will build a strong communication network for controlling and Maintaining backup for Blimp. We introduced an energy efficient Wireless Ad-hoc network named as SVEEN for Controlling, monitoring and Coordinating blimp Sensor ad-hoc network in the transmission and reception of the information.

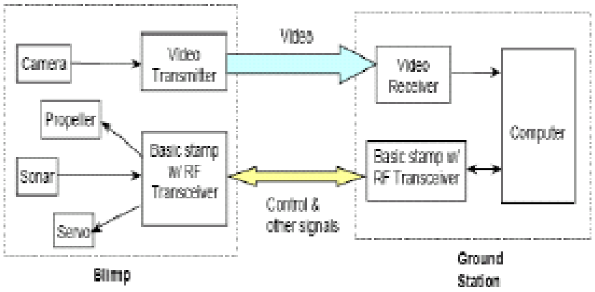


Fig 2. Second iteration of the blimp design [1]

II. SVEEN DESIGN

A. Review of SVEEN

SVEEN adaptively elects "controllers, coordinators, monitors and connectors" from all nodes in the network. SVEEN coordinators stay awake continuously and perform multi-hop packet routing and switching routing within the ad hoc network, while other nodes remain in power-saving mode and periodically check if they should wake up and become a coordinator. SVEEN achieves five goals. First, it ensures that atleast coordinators are elected so that every node is in radio range of at least one coordinator, one connector and one controller. Second, it rotates the coordinators in order to ensure that all nodes share the task of providing global connectivity roughly equally. Third, it attempts to minimize the number of nodes elected as coordinators, thereby increasing network lifetime, but without suffering a significant loss of capacity or an increase in latency. Fourth, it elects coordinators using only local information in a decentralized manner – each node

---

Sandip Vijay, B.Sc(Engg.), M.Tech., Member of IEEE, Life Member of ISTE, Member of NSBE (USA), Member of ISOC-Internet Society(USA), MIEANG, (snvijdpt@iitr.ernet.in) has more than thirty five international and national journal / proceedings publication in field of Wireless & Digital Communication and networks. He has served Department of Electronics & Communication Engg. as a Professor of G.E.I.T. He has more than Eight years of experience in field of teaching and research. He has supervised twelve M.Tech Dissertation. Currently he is pursuing research work in Wireless Ad hoc Network at I.I.T., Roorkee under Ministry of HRD, Government of India fellowship. He has won several research awards in national and International conferences. He is the active reviewer of IEEE Sensor Journal. Currently, he is member of technical committee ACM- CS conferences of ICAIT 2008, Shenzhen, China (www.ICAIT.org) .

S.C .Sharma (scs60fpt@iitr.ernet.in**)** received M.Tech. (Electronics & Communication Engg. and Ph.D. (Electronics & Computer Engg.) in 1983 and 1992 respectively from IIT Roorkee (erstwhile University of Roorkee). He started his career as R & D Engineer in 1983 then joined teaching profession in Jan. 1984 in IIT-Roorkee and continuing till date. He has published over Seventy research papers in national and international journals/conferences and supervised more than 30 projects/dissertation of PG students. He has supervised several Ph.D. in the area of Computer Networking, Wireless Network, Computer Communication and continuing supervising Ph.D. in the same area. Currently, he is supervising Six Ph.D. Scholars. He has successfully completed several major research projects independently and many minor research projects related to Communication and SAW filter Design sponsored by Government of India. IIT-Roorkee has awarded him the Khosla annual research prize with best research paper in the year 2000. His many research papers has been awarded by National and International Committees. He has worked as research scientist at FMH, Munchen, Germany in the year 1999. He has chaired several national and International Conferences. He is the active reviewer of IEEE Sensor Journal and Chief Editor of two reputed International Journals and Editor of National Journal(BITS,PILANI). Currently, he is member of technical committee ACM CS conferences of ICAIT 2008, Shenzhen, China (www.ICAIT.org) and ISN-2008, India. He is the Honorary member of NSBE, ISOC, IAENG, USA. He has also worked as Group leader of Electronics & Instrumentation Engg. department of BITS-Pilani-Dubai Campus, from Aug. 2003 to Aug. 2005. Presently he is continuing as Associate Professor at IIT Roorkee.

only consults state stored in local routing tables during the election process. Fifth , it selects monitor which can look after the periodical broadcasts and receive messages. SVEEN is Hybrid: each node periodically broadcasts HELLO messages and send ACK's that contain the node's status (i.e., whether or not the node is a coordinator), its current connector, and its current neighbors. From these HELLO messages, each node constructs a list of the node's neighbors and coordinators, and for each neighbor, a list of its neighbors and coordinators. As shown in figure 3, SVEEN runs above the link and MAC layers and interacts with the routing protocol.

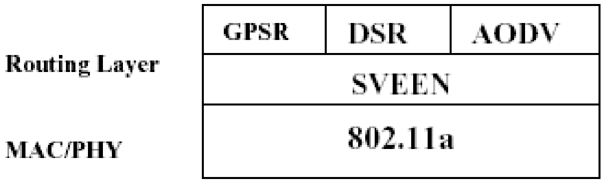


Fig 3. SVEEN is a protocol that operates under the routing layer and above the MAC and physical layers.

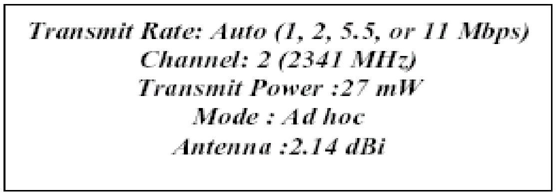
SVEEN controls, coordinates and connects the routing algorithm. We chose to implement geographic forwarding primarily because of its simplicity; this structuring allows SVEEN to take advantage of power-saving features of the link layer protocol, while still being able to affect the routing process. How a node decides that it should announce that it is a coordinator, and how it decides that it should withdraw from being a coordinator?

B. Motivation for SVEEN Design

Poor network performance suggested that the distance vector routing protocol[1,2], which has no notion of link quality or loss rate, was forwarding traffic over links of low or intermediate quality. To determine how common such links were in the testbed, we performed a series of experiments to measure the loss rate[3] between each pair of nodes in the testbed. The remainder of this paper details the experimental procedure and results. Table 4: 802.11 settings:

Table 4. SVEEN testbed parameters.

In SVEEN, which uses local HELLO messages to propagate topology information, it does not depend on them for correctness: when HELLO messages are lost[4], SVEEN elects more coordinators, but does not disconnect the backbone. Figure 3 shows the result of our election algorithm at a random point in time on a network of 100 nodes in a 1000 m x 1000 m area, where each radio has an isotropic circular range with a 50 m radius. Solid lines connect coordinators that are within radio range of each other.



Transmit Rate: Auto (1, 2, 5.5, or 11 Mbps)
Channel: 2 (2341 MHz)
Transmit Power :27 mW
Mode : Ad hoc
Antenna :2.14 dBi

C. Simulator Implementation

As This section describes our implementation of SVEEN, geographic forwarding, the 802.11 power saving mode (with our own improvements), and the energy model we used in our simulations. We ran our SVEEN implementation in the *ns*-2 network simulator environment.

A. SVEEN geographic forwarding

Our implementation uses a geographic forwarding algorithm. We chose to implement geographic forwarding [5]primarily because of its simplicity; SVEEN can be used with other routing protocols as Well.

SVEEN's election algorithm requires each node to advertise its coordinators, its neighbors, and if it is a coordinator, a tentative coordinator[6], or a non- coordinator. To reduce protocol overhead, we piggyback SVEEN HELLO information (see section) onto the broadcast updates required by geographic forwarding (see table 2). Each node enters all the information it receives in broadcast updates into a *neighbor table*. Consequently, this neighbor table contains a list of neighbors and coordinators, and for each neighbor, a list of its neighbors and coordinators. Geographic forwarding forwards packets using a greedy algorithm. The source node annotates each packet with the geographic location of the destination node. Upon receiving a packet for a node not in radio range, a coordinator forwards the packet to a neighboring coordinator that is closest to the destination. If no such coordinator exists, the packet is forwarded to a non-coordinator that is closer to the destination. Otherwise, we know that a packet has encountered a void, and so it is dropped. (We did not implement an idea like GPSR [13], which ameliorates the effects of voids.).

Our simulations do not use a location service. Instead, each sender uses the GOD module of ns to obtain the geographic location of the destination node. Hence, our simulation results may be better than one might expect with a real location service, such as GLS [15].

Our geographic forwarding algorithm also implements MAC-layer failure feedback and interface queue traversal [1, 13]. These mechanisms allow the routing layer to readily remove unresponsive nodes from its routing table and rescue packets using these nodes as the next hop.

III. COORDINATOR ELECTION

A node uses information from its neighbor table to determine if it should announce or withdraw itself as a coordinator[7,22]. Figure 4 shows the coordinator announcement algorithm. A non-coordinator node periodically calls check-announce coordinator to determine if it should become a coordinator or not. Check-announce-coordinator first computes *C*, the number of additional neighbor pairs that would be connected if the node becomes a coordinator, using connect-pair. If *C* > 0, the node computes *delay* and waits for *delay* seconds before re-computing *C*. If C continues to be greater than 0 after *delay* seconds, the node announces itself as a coordinator. Connect-pair calculates the number of would-be connected neighbor pairs by iterating through the node's neighbors in the neighbor table.

A similar routine exists for checking if every pair of neighbor nodes can reach each other via one or two other neighbors. That routine is used by the withdraw algorithm. In addition to the coordinator election algorithm shown in figure 4, we implemented a special case for electing coordinators. The geographic routing algorithm can readily detect that a coordinator has left the region through MAC layer failure feedback[9,28]. However, the SVEEN election algorithm may not react fast enough to elect new coordinators. In the worst case, nodes must wait until the old coordinator information has expired in the neighbor table before a new coordinator can be elected. Because geographic forwarding falls back to using non-coordinators to route packets if coordinators do not exist, a non-coordinator node announces itself as a coordinator if it has received a large number of packets to route in the recent past. If this coordinator turns out to be redundant, the coordinator withdraw algorithm will force the node to withdraw itself as a coordinator soon after.

IV. 802.11 AD HOC POWER-SAVING MODE

SVEEN determines when to turn a node's radio on or off, but depends on the low level MAC layer to support power saving functions, such as buffering packets for sleeping nodes. We have implemented SVEEN on top of the 802.11 MAC and physical layers with ad hoc power saving support [11]. 802.11 ad hoc power-saving mode uses periodic *beacons* to synchronize nodes in the network. Beacon packets contain timestamps that synchronize nodes' clocks. A beacon period starts with an ad hoc traffic indication message window (*ATIM window*), during which all nodes are listening, and pending traffic transmissions are advertised. A node that receives and acknowledges an advertisement for unicast or broadcast traffic directed to itself must stay on for the rest of the beacon period. Otherwise, it can turn itself off at the end of the ATIM window[10,17], until the beginning of the next beacon period. After the ATIM window, advertised traffic is transmitted. Since traffic cannot be transmitted during the ATIM window, the available channel capacity is reduced.

When the 802.11 MAC layer is asked to send a packet, it may or may not be able to send it immediately, depending on which ATIM's have been sent and acknowledged in the immediately preceding or current, ATIM window. If the packet arrives at the MAC during the ATIM window, or if the advertisement for the packet has not been acknowledged, it needs to be buffered. In our implementation, we buffer packets for two beacon periods. Packets that have not been transmitted after two beacon periods are dropped. The beacon period and ATIM window size greatly affect routing performance [21]. While using a small ATIM window may improve energy savings, there may not be enough time for all buffered packets to be advertised. Using an ATIM window that is too large not only decreases available channel utilization, it may also not leave enough room between the end of the ATIM window and the beginning of the next beacon period to transmit all advertised traffic. We have experimentally determined that a beacon period of 200 ms and an ATIM window size of 40 ms result in good throughput and low loss rate[24]. Aside from decreased channel capacity, 802.11 power saving mode (without SVEEN) also suffers from long packet delivery latency: for each hop that a packet traverses, the packet is expected to be delayed for half a beacon period[25].

V. IMPROVING 802.11 USING SVEEN

Using SVEEN on top of 802.11 ad hoc power saving mode can improve routing throughput and packet delivery latency. Because coordinators do not operate in power saving mode, packets routed between coordinators do not need to be advertised or delayed. To further take advantage of the synergy between SVEEN and 802.11 power saving mode, we have made the following modifications to our simulation of 802.11 power saving mode.

Loss Experiment

The cumulative distribution of delivery rates across all links for each of the two packet sizes. The two directions between each node pair are considered separate links. The figure shows that about 50% of the links deliver no packets, while the best 20% of links deliver more than 95% of their packets. The delivery rates of the remaining 30% of links are approximately evenly distributed. Other experiments on different days and at different times confirm that there are always many links in the network which provide intermediate delivery rates. The small packet tests exhibit higher delivery rates than the large packet tests because there is less chance of radio interference at the receivers during shorter transmissions. Regardless, the number of links with intermediate loss rates and the distribution of those loss rates is what is relevant, and nearly identical between the two types of tests.

Motivated by statistics showing high traffic death rates occurring near these areas, a project aimed at implementing a wirelessly controlled speed limiting system is herein presented. The proposed system is based on microcontroller technology for collecting data related to speed and transmitting it through a transceiver to a base station that analyzes the transmitted data and takes appropriate decisions related to speed limit and control requirements. The transmitting/receiving base station is to be located strategically at the entrance of populated areas where speed is to be maintained. Upon entry to these areas, the speed control is switched from the car driver to the speed limiting system. A model of the actual system is designed and built.

Coordinator election

A node uses information from its neighbor table to determine if it should announce or withdraw itself as a coordinator. Program1 shows the coordinator announcement algorithm. A non - coordinator node periodically calls check – announce - coordinator to determine if it should become a coordinator or not.

Check - announce - coordinator first computes *C*, the number of additional neighbor pairs that would be connected if the node becomes a coordinator, using connect-pair. If *C* > 0, the node computes *delay* using equation (2) and waits for *delay* seconds before recomputing *C*.

If *C* continues to be greater than 0 after *delay* seconds, the node announces itself as a coordinator. connect-pair calculates the number of would-be connected neighbor pairs by iterating through the node's neighbors in the neighbor table. A similar routine exists for checking if every pair of neighbor nodes can reach each other via one or two other neighbors. That routine is used by the withdraw algorithm. In addition to the coordinator election algorithm shown in figure 4, we implemented a special case for electing coordinators. The geographic routing algorithm can readily detect that a coordinator has left the region through MAC layer failure feedback. However, the SVEEN election algorithm may not react fast enough to elect new coordinators. In the worst case, nodes must wait until the old coordinator information has expired in the neighbor table before a new coordinator can be elected. Because geographic forwarding falls back to using non-coordinators to route packets if coordinators do not exist, a non-coordinator node announces itself as a coordinator if it has received a large number of packets to route in the recent past. If this coordinator turns out to be redundant, the coordinator withdraw algorithm will force the node to withdraw itself as a coordinator soon after.

```
// a non-coordinator node periodically calls this routine
to see if it should become a coordinator[1]
check-announce-coordinator()
C = connect-pairs( )
If C > 0 {
Calculate delay using equation (2), using C as Ci
Wait delay
if connect-pairs( ) > 0 {
Announce itself as a coordinator
}
}
// returns number of neighbor pairs a node can connect
```

```
if it becomes a coordinator
connect-pairs( )
n = 0
for each neighbor a in neighbor table {
    for each neighbor b,b > a, in neighbor table {
        if share-other-coordinators(a, b) == false {
            n->n + 1
        }
    }
}
return n
```

```
// returns true if neighbors a and b are connected by one or two other coordinators
share-other-coordinators(a, b)
// coordinator lists are kept in the neighbor table
for each coordinator c_a in a's coordinator list {
    if c_a equals self {
        continue
    }
    else if c_a in b's coordinator list {
        return true
    }
    // try to see if we know a path from a to b via two coordinators
    else if c_a in neighbor table {
        for each coordinator c_c_a in c_a's coordinator list {
            if c_c_a equals self {
                continue
            }
            else if c_c_a in b's coordinator list {
                return true
            }
        }
    }
}
return false
```

Program 1: Coordinator announcement algorithm.

Connector and monitoring section

A non-connected sleeping node uses as connectors for unconnected path till coordinator chose it in alternate path . A non-connected node periodically calls check-announce-monitoring to determine if it should become a connector and allow monitor or not. Check-announce-connector first computes *d*, the number of additional neighbor pairs that would be connected if the node becomes a connector to monitor the sleeping nodes, using connect-pair. If d > 0, the node computes *Pseudo- broadcast* using and waits for ACK *call* before reconstituting d. If d continues to be greater than 0 after *ACK call*, the node announces itself as a connector. connect-pair calculates the number of unconnected neighbor pairs by iterating through the node's neighbors in the neighbor table. A similar routine exists for checking if every pair of neighbor nodes can reach each other via one or two other *neighbors.*

VI. PERFORMANCE EVALUATION

To measure the effectiveness of SVEEN, we simulated SVEEN, with geographic forwarding, on several static and mobile topologies. Simulation results show that SVEEN not only performs well by extending network lifetime, it out-performs unmodified 802.11 power saving network in handling heavy load, per-packet delivery latency, and network lifetime.

Simulation environment

We simulated SVEEN in the NS-2 & OPNET[17] network simulator using the CMU wireless extensions [5]. The geographic forwarding algorithm, as described in section 4.1, routes packets from source to destination. SVEEN runs on top of the 802.11 MAC layer with power saving support and modification.

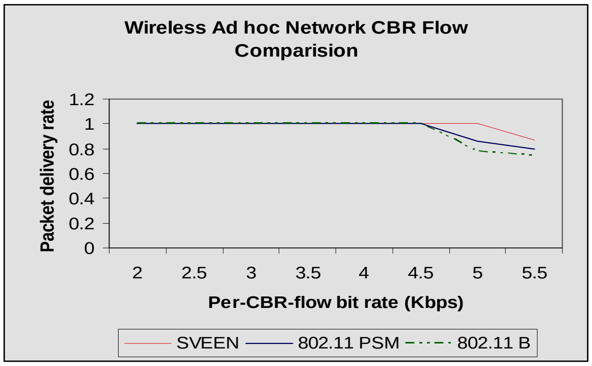


Fig. 5 Packet delivery rate as a function of per-CBR-flow bit rate. Each packet traverses six hops. Under higher traffic load, SVEEN delivers more packets than 802.11 PSM, but slightly less than 802.11B

Effects of mobility

Figure 6 shows the effects of mobility on packet loss rate. In these simulations, an area of 1000 m * 1000 m is used. Each simulation lasts 400 s.

Nodes follow the random waypoint motion model, and the length of the pause time reflects the degree of mobility. The degree of mobility does not significantly affect routing with Span backbones. Span consistently performs better than both 802.11 PSM and 802.11. Most packet drops in these simulations are caused by temporary voids created by mobility. Because geographic forwarding with Span encounters fewer voids, its loss rate is lower.

Backbone election

Ideally, SVEEN would choose just enough backbones to preserve connectivity and capacity, but no more; any backbones above this minimum just waste power. This section compares the number of backbones Span chooses with the number that would be required to form a hexagonal grid layout, shown in figure 7; the hex grid layout of nodes, while perhaps not optimal, produces a connected backbone in every direction with very few backbones. The hexagonal grid layout of backbones places a backbone at each vertex of a hexagon. Every backbone can communicate with the three backbones that it has connected to through an edge of a hexagon, which is 250 m long (the radio range). Each hexagon has six backbones, but each backbone has shared by three hexagons. Therefore, each hexagon is only responsible for two backbones. Each hexagon has an area of 162,380 m^2.

Figure 8 shows backbone density as a function of node density. For each node density, backbone density has computed from the average number of backbones elected by Span over 500 s of five mobile simulations.

Controller election

SVEEN would chose just one controller to centrally monitor the backbones, within the backbones and keep the table of unfaithful nodes, which must be in the mid of geographical area of the network.

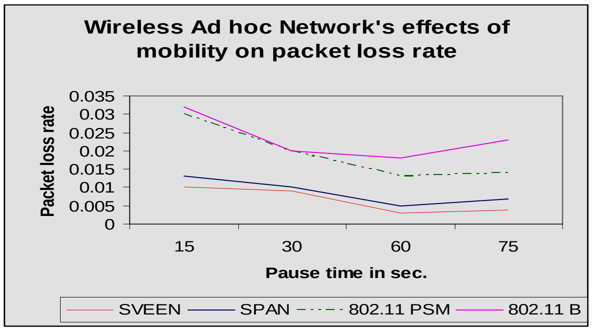


Fig. 6 Packet loss rate as a function of pause time. The simulation area is 1 Km* 1 Km square. Mobility does not affect SVEEN very much, and Geographic forwarding with SVEEN delivers more packets than with 802.11 PSM and 802.11 because it encounters fewer voids.

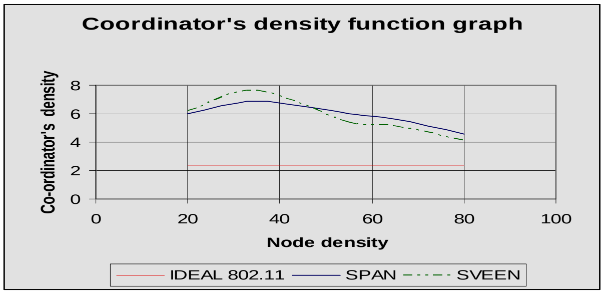


Fig. 8 Ideal and actual backbone density as a function of node density. The ideal curve represents an approximate lower bound on the number of Backbones needed. SVEEN elects more backbones than the ideal case Because of lower node density, backbone rotation, and announcement collision.

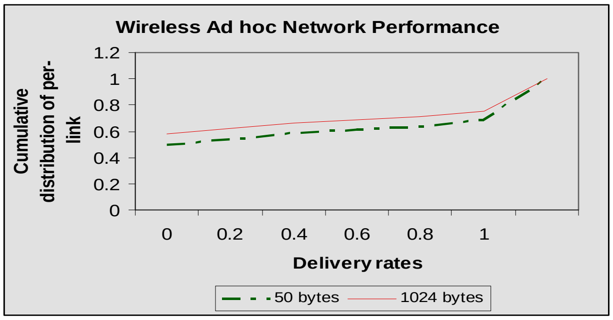


Fig. 9 shows the network performance in case of 50 bytes Vs. 1024 bytes.

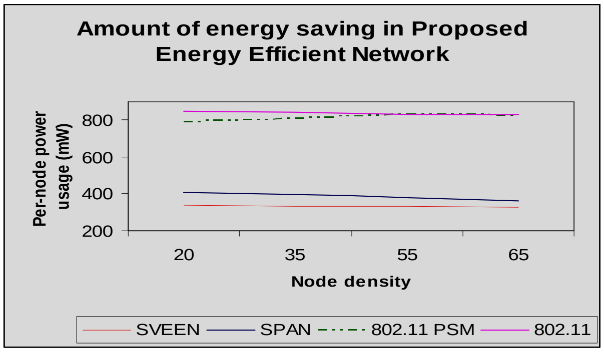


Fig. 10 Per-node power usage. SVEEN provides significant amount of Savings over 802.11 PSM and 802.11.

VII. CONCLUSION

This paper presents application of *SVEEN for sensor networks* for implementation of a blimp communication network , it is a distributed as well as centralized coordination technique for multi-hop ad hoc wireless networks that reduces energy consumption without significantly diminishing the capacity or connectivity of the network. SVEEN adaptively elects *backbones* from all nodes in the network, and rotates them in time. SVEEN backbones stay awake and perform multi-hop packet routing within the ad hoc network, while other nodes remain in power-saving mode and periodically check if they should awaken and become a backbone. With SVEEN, each node uses a random backoff delay to decide whether to become a backbone. This delay is a function of the number of other nodes in the neighborhood that can be bridge using this node and the amount of energy it has remaining. To identify the unfaithful nodes in the particular geographic region and to control backbone, we introduced a controller from the backbones, which must be fall in the middle of geographical area of complete network. Our results show that SVEEN not only preserves network connectivity, it also preserves capacity, decreases latency, and provides significant energy savings. For example, for a practical range of node densities and a practical energy model, our simulations show that the system lifetime with SVEEN is more than a factor of two better than without SVEEN. The amount of energy that SVEEN saves increases only slightly as density increases. This is largely because the current implementation of SVEEN uses the power saving features of 802.11, in which nodes periodically wake up and listen for traffic advertisements. Section V shows that this approach can be extremely expensive. This warrants investigation into a more robust and efficient power saving MAC layer, one that minimizes the amount of time each node in power saving mode must stay up.

These results indicate that any successful Sensor ad hoc routing protocol will need to consider loss rate during route selection. This will require some estimation of loss rate in both directions of each network link. The estimate must rapidly adapt to changing conditions or node mobility. Measurements of signal strength and quality reported by the 802.11 firmware show that neither of these metrics is sufficient alone.

REFERENCES

[1] J. Broch, D. Johnson and D. Maltz, The dynamic source routing protocol for mobile ad hoc networks, Internet draft, IETF Mobile Ad Hoc Networking Working Group (December 1998).

[2] J. Broch, D. Maltz, D. Johnson, Y. Hu and J. Jetcheva, A performance comparison of multi-hop wireless ad hoc network routing protocols, in: *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Dallas, TX (August 1998).

[3] J. Chang and L. Tassiulas, Energy conserving routing in wireless ad hoc networks, in: *Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Network (MobiCom)*, Dallas, TX (August 1998).

[4] G. Chesson, XTP/protocol engine design, in: *Proceedings of the IFIP WG6.1/6.4 Workshop*, Rüschlikon (May 1989).

[5] CMU Monarch extensions to *ns*, http://www.monarch.cs.cmu.edu/

[6] B. Das and V. Bharghavan, Routing in ad-hoc networks using minimum connected dominating sets, in: *Proceedings of the IEEE International Conference on Communications (ICC'97)* (June 1997).

[7] L. Feeney and M. Nilsson, Investigating the energy consumption of a wireless network interface in an ad hoc networking environment, in: *Proceedings of IEEE INFOCOM*, Anchorage, AK (2001).

[8] W. Fenner, Internet Group Management Protocol, Version 2, RFC-2236 (November 1997).

[9] S. Floyd, V. Jacobson, S. McCanne, C.G. Liu and L. Zhang, A reliable multicast framework for lightweight sessions and application level framing, in: *Proceedings of the ACM SIGCOMM*, Boston, MA (September 1995) pp. 342–356.

[10] W.R. Heinzelman, A. Chandrakasan and H. Balakrishnan, Energyefficient communication protocols for wireless microsensor networks, in: *Proceedings of the Hawaaian International Conference on Systems Science* (January 2000).

[11] IEEE,Wireless LANMedium Access Control and Physical Layer specifications, IEEE 802.11 Standard, IEEE Computer Society LAN MAN Standards Committee (August 1999).

[12] C. Intanagonwiwat, R. Govindan and D. Estrin, Directed diffusion: A scalable and robust communication paradigm for sensor networks, in: *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA (August 2000).

[13] B. Karp and H.T. Kung, GPSR: Greedy Perimeter Stateless Routing for wireless networks, in: *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA (August 2000).

[14] R. Kravets and P. Krishnan, Application-driven power management for mobile communication, in: *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Dallas, TX (October 1998).

[15] J. Li, J. Jannotti, D.D. Couto, D. Karger and R. Morris, A scalable location service for geographic ad-hoc routing, in: *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)* (August 2000).

[16] W. Mangione-Smith, P.S. Ghang, S. Nazareth, P. Lettieri, W. Boring and R. Jain, A low power architecture for wireless multimedia systems: Lessons learned from building a power hog, in: *1996 International Symposium on Low Power Electronics and Design Digest of Technical Papers*, Monterey, CA (August 1996).

[17] *ns* notes and documentation (2000) http://www.isi.edu/vint/nsnam/

[18] C. Raghavendra and S. Singh, PAMAS: Power Aware Multi-Access Protocol with Signaling for ad hoc networks, ACM Computer Communication Review (July 1998) 5–26.

[19] R. Ramanathan and R. Rosales-Hain, Topology control of multi-hop wireless networks using transmit power adjustment, in: *Proceedings of IEEE INFOCOM*, Tel Aviv, Israel (March 2000).

[20] V. Rodoplu and T.H. Meng, Minimum energy mobile wireless networks, in: *Proceedings of the IEEE International Conference on Communications(ICC)*, Vol. 3, Atlanta, GA (June 1998) pp. 1633–69.

[21] David B. Johnson. Routing in ad hoc networks of mobile hosts. In Proc. of the IEEE Workshop on Mobile Computing Systems and Applications, pages 158{163, December 1994.

[22] Jain-Shing Liu and Chun-Hung Richard Lin.: Energy-efficiency clustering protocol in wireless sensor networks *Ad Hoc Networks, Volume 3, Issue 3*, May 2005, Pages 371-388

[23] Σινγη, Ραγηαϖενδρα, Στεπανεκ : Power-Aware Broadcasting in Mobile Ad Hoc Networks - (1999)

[24] M. Stemm, R. Katz, and S. Seshan. A Network Measurement Architecture for Adaptive Applications. Proceedings of IEEE Infocom 2000, March 2000.

[25] F. Kuhn, R. Wattenhofer, and A. Zollinger. Ad-hoc networks beyond unit disk graphs. In Proceedings of the 2003. http://citeseer.ist.psu.edu/kuhn03adhoc.html

[26] Wu. J. and Gao M. : On Calculating Power Aware Connected Dominant Set For Efficient Routing In Ad Hoc Wireless Networks. In proceedings of International conference on parallel processing, sept, 2001.

[27] J. Wu, F. Dai, M. Gao, and I. Stojmenovic, On calculating power aware connected dominating sets for efficient routing in ad hoc wireless networks, IEEE/KICS Journal of Communication Networks**,** Vol. 4, No. 1, March 2002, 59-70.

[28] Spiewla Jacek K.: Dynamic Routing Protocols and Energy Efficient Communication in Mobile Ad Hoc Networks, www.ece.mtu.edu/ee/faculty/cchigan/ee4272/projects/Spring03-paper1.pdf

[29] Susumu Ishihara and Tokuya Inagaki: A Scheme for Expanding Grid Size of Geographical Adaptive Fidelity, http://ieeexplore.ieee.org/iel5/4297370/4297371/04297433.pdf

[30] *Ivan Stojmenovic, Prosenjit Bose and Pat Morin and Jorge Urrutia:* Greedy Face Greedy (GFG) Routing with guaranteed delivery in ad hoc wireless networks, DIALM '99: Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications, Seattle, Washington, United States, pages 48-55, 1999, ISBN 1-58113-174-7.

[31] *Sandip Vijay, and S. C. Gupta*: SVEEN a comparative protocol design, Proceeding of IEEE 2nd International Conference on Wireless & Sensor Networks, IIITA, India.