Design and Implementation of AIRF: A Anycast-Based Integrated Routing Framework for WSNs

Guofeng Yan[a,b,c], Yuxing Peng[b,c], Shuhong Chen[d], Junyi Liu[c]
[a] School of Computer and Communication, Hunan Institute of Engineering, Hunan, 411104, China
Email: gfyan@nudt.edu.cn, gfyan@mail.csu.edu.cn
[b]Science and Technology on Parallel and Distributed Processing Laboratory, National University of Defense Technology, Changsha 410073, China
Email: pyx@nudt.edu.cn
[c] School of Computer, National University of Defense Technology, Changsha 410073, China
Email: jyliu@nudt.edu.cn
[d] School of Information Science and Engineering, Central South University, Changsha 410083, China
Email:shchenannabell@mail.csu.edu.cn

***Abstract*—In wireless sensor networks (WSNs), asynchronous sleep wake scheduling protocols can significantly reduce energy consumption without incurring the communication overhead. However, the savings could come at a significant cost in delay performance. In this paper, we consider anycast-based integrated routing framework (AIRF) to reduce the cost in delay performance of communications in multihop WSNs. Without tight time synchronization or known geographic information, AIRF provides low-delay cost route. We implement a low-overhead AIRF module in TinyOS kernel by modifying BLIP protocol stack, i.e., the Berkeley Low-power IP stack; as demonstrated, this implementation can be incorporated into existing routing protocols with the least effort. We describe in detail the format of AIRF message, the dynamic updating process of mapping record information, and anycast data flow under TinyOS. And then, we present the anycast group management system. Finally, we analyze the application performance of AIRF in balancing network load, network lifetime, and delay performance.**

***Index Terms*—Integrated routing, TinyOS, Multihop WSN, Anycast protocol, Lifetime**

I. INTRODUCTION

WIRELESS sensor networks (WSNs) create the technological basis for building pervasive, large-scale distributed systems, which can sense their environment in great detail, communicate the relevant information gathered from the monitored field through wireless links, reason collectively upon the observed situation and react according to the application-specific goals [1]–[3]. Therefore, WSNs can be deployed for a wide range of applications [4]–[6] such as environmental monitoring, health care, military applications and so on. These applications employ a large number of low-cost sensing devices equipped with wireless communication and computation capabilities. Due to multimedia applications typically produce a huge volume of data that require extensive processing power and high transmission rates, power consumption becomes a fundamental concern in WSNs since the physically small nature of sensors. More particularly, transmitting a video stream using the shortest path will drain the node of its energy along the path, thereby shortening the network lifetime. Therefore, a critical problem is to develop new routing protocols to maximize the network lifetime without decreasing the quality of service (QoS). Recently, anycast routing protocol has become more and more important for WSNs due to its utilization. In anycast-based forwarding schemes, nodes maintain multiple candidates of next-hop nodes and forward their packet to the first candidate node that wakes up. Hence, an anycast forwarding scheme can substantially reduce the one-hop delay over traditional schemes, especially when nodes are densely deployed, as is the case for many WSN applications [7]–[9]. In this paper, we study how to improve the performance of network lifetime, delay and network load balance by using a novel anycast scheme for WSNs, i.e., anycast-based integrated routing framework (AIRF) that exploits the broadcast nature of the wireless medium.

In order to improve the tradeoff of energy-savings and QoS, we focus on how to reduce the amount of request and reply packets through AIRF. In AIRF, a router employs different routing strategies for various anycast data flows, respectively. In this paper, we mainly consider two types of anycast data flows, i.e., single-packet anycast flow and multi-packet anycast flow. In our routing approach, a AIRF client receives only one response message from its closest or best service provider. Responses from other service providers will be discarded by the controlling gate in gateway nodes. On the other hand, we distribute effectively network traffic to different servers by AIRF mechanism. In other words, our anycast routing approach reduces the control overhead and is

This work was supported by the National High Technology Research and Development Program of China under Grant Number 2011AA0A202, National 973 Basic Research Program of China under Grant Number 2011CB302601, and China Postdoctoral Science Foundation under Grant Number 2013M542561.

doi:10.4304/jsw.9.3.569-575

suitable for large-scale WSNs.

The remainder of the paper is organized as follows. We discuss the related work in Section II. Then we elaborate the design of AIRF algorithm in Section III, Section IV further presents the implementing issues of AIRF. We analyze the performance of our scheme in Section V. Finally, we conclude this paper in Section VI .

II. Related work

Anycast becomes more and more important, owing to its utilization in peer-to-peer systems (P2P), DNS service, or content delivery networks (CDNs) [7], [10], [11]. However, anycast-based solutions [12]–[16] have been proposed that do not synchronize awake times at all. Rather, they exploit the path redundancy (i.e., multiple next hop nodes) available in the network to opportunistically maximize the network lifetime and reduce sleep latency. In anycast mechanism, a single anycast address is assigned to service providers within an anycast group [17], [18]. When a client sends packets to an anycast address, routing node will deliver the packets to the closest service provider which matches the anycast address, based on some criteria, such as location of the client node, delay, or QoS. Some applications can be simplified by using anycast service mechanism. Fox example, the efficiency of locating a suitable one from a group of available servers through anycast mechanism is higher than multicast mechanism [19]. The most effective way to maximize the network lifetime is to carefully stagger the awake times of all nodes along the path towards the destination [20]–[24]. However, in a realistic network with many nodes, due to unavoidable clock drift inherent in cheap sensor motes [25]–[27], many current protocols end up expending excessive energy synchronizing the wake-up times. In [28], the authors present a DSR-based anycast scheme, i.e., A-DSR. which modified the Dynamic Source Routing (DSR) protocol and added an anycast flag in reserved field of the DSR header. A-DSR uses this anycast flag to join, leave, and find the anycast group.

To select the best forwarding node, current anycast-based solutions either probe the neighborhood systematically [29], [30] or periodically exchange routing information among one hop neighbors [31] before forwarding. Therefore, anycast-based solutions incur either a high overhead at low traffic due to exchanging route information or a long delay with a high traffic load for probing the neighbour nodes. As a results, although anycast provides the appropriate mechanism to opportunistically reduce the delay in a duty-cycled network, current anycast-based solutions cannot handle the diverse traffic expected in WSNs.

III. Algorithm framework of AIRF

In the paper, we define the packets that arrive at a router to be receiving packets, and packets that leave from a router to be sending packets. AIRF routing system disposes receiving packets and sending packets according to different routing principles, respectively. Therefore, AIRF is composed of two subsystems: receiving routing subsystem and sending routing subsystem. In receiving routing subsystem, a routing node looks up mapping records corresponding to the destination address **A** in MAP table when a packet with address **A** arrives. If not finding the corresponding item, AIRF won't process it and forward directly to routing module. Otherwise, AIRF maps the address **A** into a corresponding unique identification address **U**, and obtain the flow information (i.e., indicate the source address/port number and the destination address) by seeking the linked list of flow information. If finding a suitable match, the packet is sent to the corresponding next-hop. If the packet belongs to a new data flow, AIRF will not find the corresponding records, and then add the data flow information with a corresponding next-hop information into MAP table, where the next-hop is the closest router according to the local routing strategy. If a router has more than one routing to the same destination, AIRF chooses randomly a forwarding route from these paths. Therefore, the method routes the packets of the same anycast flow to the same service host. For a new anycast data flow, AIRF recalculates the path according to routing information. We implement this process by adding a routing module, $a2irf\_treeRouting()$, in tree routing module of BLIP protocol stack of TinyOS. The algorithm framework of revised $a2irf\_treeRouting()$ function is shown in Algorithm 1.

For the sending routing algorithm, the processing path is different from the receiving routing algorithm, but their processing mechanisms are basically the same. Another, the sending routing algorithm does not necessary to recalculate the checksum for a routing packet.

IV. Implementation details

A. Data structure design

To implement AIRF, we firstly make a transition between an anycast IP address and an unicast IP address. In addition, our routing system needs to obtain the information about router interface of every anycast data flow. Hence, we define two types of data structures to implement AIRF protocol, i.e., MAP table and FIS. The MAP table is used to transform an anycast IP address into an unicast IP address, and the FIS is mainly used to record routing information of router interface corresponding to an anycast flow. The record items of the two data structures are shown in TABLE I and TABLE II, respectively.

B. The format design of AIRF message

AIRF exchanges routing information between different subsystems through various types of messages. We mainly define three types of messages. In kernel, the AIRF routing system exchanges dynamically the routing information with MAP table. The MAP table management system in kernel communicates with flow label management system using AIRF protocol messages. We execute these operations under user mode. AIRF mainly use

Algorithm 1 The algorithm framework of $a2irf\_treeRouting()$ in AIRF

```
BEGIN
if the destination address A is not an anycast address then
    RETURN;
else
    for each node i of MAP table do
        Compare A and the anycast address of node i
        if A is a registered anycast address then
            if the number of hop in MAP table ≠ 1 then
                Replace A with the corresponding unique unicast address of next-hop host;
            else
                Replace A with the corresponding anycast server address;
            end if
            if existing a corresponding record of source address and port number in FIS then
                Send the packet through the corresponding routing interface index using the function skb− > dst− > input();
                Return;
            else
                Add new flow node record into FIS;
            end if
            Call ip_route_input() function to look up route table;
            Modify the routing interface index of the corresponding flow node as the selected routing interface;
            Restore the unicast address of anycast server to A;
            Call skb− > dst− > input() to send the packet;
        end if
    end for
end if
Reconstruction packet checksum;
END
```

TABLE I
MAP TABLE STRUCTURE

symbol	description
$*ptr_{afl}$	Pointer to anycast flow structure
$*ptr_{prev}$	Pointer to prior network node
$*ptr_{next}$	Pointer to next network node
$node(i)_{anycast}$	Anycast address of node i
$nexthop_{unicast}$	Unique unicast address of next-hop
$server_{unicast}$	Unique unicast address of anycast server
$server_{anycast}$	Anycast address of anycast server
hop_{number}	Number of hop

TABLE II
FLOW INFORMATION STRUCTURE (FIS) OF ANYCAST DATA

symbol	description
$label_{flow}$	Anytcast flow label
$*ptr_{prev}$	Pointer to prior network node
$*ptr_{next}$	Pointer to next network node
$source_{unicast}$	Unique unicast address of source node
$source_{port}$	Port number of source node
$destination_{unicast}$	Unique unicast address of destination node
$router_{interface}$	Index of router interface

three types messages, i.e., message between flow structure records, MAP table message, and communication message between kernel AIRF routing system and external user process. The structures of these AIRF messages are shown in Fig. 1, Fig. 2, and Fig. 3.

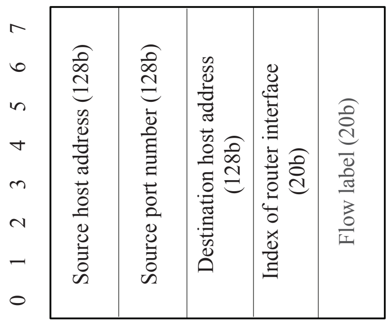

Fig. 1. Message format of FIS record

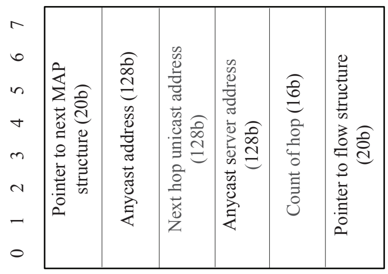

Fig. 2. Message format of MAP table record

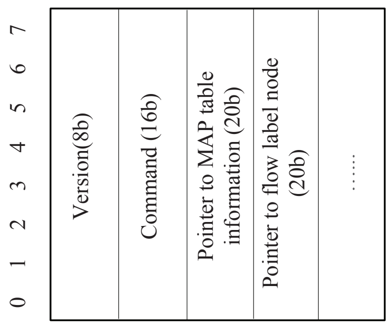

Fig. 3. Format of communication message between kernel AIRF and external process

The packet of AIRF maybe becomes invalid. Hence, AIRF has to estimate the validity of current packet according to the value of version field of FIS record message. The current packet is allowed to be exchanged only when its version value is the same as the version field values of the other packets of the same data flow. Otherwise, AIRF discards the packet. In addition, AIRF distinguishes the types of different messages according to the value of command field in FIS record message.

C. Dynamic Updating subsystem between MAP table and Flow Information Structure (FIS)

Dynamic update subsystem (DUSS) of mapping records in MAP table mainly includes add, $edit$, and $delete$ operations. Moreover, DUSS can modify records in add function. When adding a record, DUSS first to allocate memory space for the new record, and set the values of corresponding variables according to the parameters. Then, DUSS seeks an appropriate position for the new record by traversing chain list from the starting node of MAP table. In DUSS, $delete$ function first to find a relevant record in MAP table, and then, deletes it. On the other hand, DUSS can also manage flow information records of Flow Information Structure (FIS). When adding a mapping record, add operation establishes a new corresponding node in FIS chain. Similarly, $delete$ operation removes the corresponding node from FIS chain when DUSS deletes a record of MAP table. For each flow information record, DUSS sets a maximum value of survival time, i.e, TTL and the flow information record automatically be deleted when TTL exceeds the maximum value.

AIRF routing system provides an external system call interface, i.e, $ip\_anycast\_ioctl()$. AIRF can obtain the information of MAP table and flow in TinyOS kernel through $ip\_anycast\_ioctl()$ and external daemon, and then, update dynamically the records of MAP table and FIS according to the command field value and version field value, respectively, which can ensure the validity of records in kernel MAP table and FIS.

D. Dynamic management of anycast group member

Dynamic management subsystem of anycast group member mainly implements the following operations, such as $join$ and $leave$ of anycast group member, initiative $register$, $leave$ of anycast server host, and ask and $answer$ to router, ect. The subsystem includes two parts: information communication module of anycast group members, and user console module. User console module allows user to deploy statically the information of anycast group member and communicate with the information communication module of anycast group member, which reads configuration files, and sends these information to router at regular intervals. During the operation, information communication system of anycast group member automatically reads, and sends configuration information according to console commands. When an anycast server host leaves, it will inform the corresponding forwarding node.

E. Software implementation and memory usage

The software implementation of AIRF requires moderate program storage and memory usage. Considering the memory limitation and usage overhead on sensor nodes, the MAP table and FIS keep the forwarded message intervals for up to 30 source nodes, with up to 10 non-overlapped intervals for each individual source node. The MAP table and FIS on a node keep adding items for new source node until it is full.

AIRF routing process is implemented by modifying BLIP protocol stack of TinyOS 2.0. We embed a new routing module into tree routing system of TinyOS 2.0. Then, we implemented a typical TinyOS data collection application, i.e., MultihopOscilloscope, based on our new routing protocol. The MultihopOscilloscope application periodically makes sensing samples and sends out the sensed data to a root via multiple routing hops with certain modified sensing parameters for our performance evaluation. These information are shown through an application system in the directory of $TOSROOT/apps/Oscilloscope/$ named $Oscilloscope$. The enabling of AIRF in MultihopOscilloscope requires the size of ROM by around 4.5KB and the size of memory by around 6.0KB, respectively.

V. Performance analysis

AIRF is designed to improve the quality of service (QoS) of WSNs. In the paper, we mainly analyze the QoS performance on network load balance, lifetime, and delay of AIRF through networks experiments in WSNs. We select the other three WSN routing protocols to evaluate the performance of AIRF, i.e., flooding, directed diffusion [34], and SPIN [35]. Flooding is chosen because it gives an indication of the worst case routing. Directed diffusion is chosen due to the fact that it is very popular in the literature and many protocols have been based on it. SPIN is chosen since it is also a source initiated protocol. In addition, network lifetime is defined as the time until the first node fails. The different type of messages and its sizes for experiments are given in TABLE III.

TABLE III
Packet sizes for experiments

Type of message	Size (bits)
Data	100
Broadcast	60
Interest	60
Energy	55
Advertise	50
Request	50

A. Experimental setting

In experiments, we use the MICAz wireless mote equipped with a $MDA300$ data acquisition board, both manufactured by Memsic Corporation. The mote's onboard radio power is adjustable within a range of -25 to 0dBm, the maximum and minimum power allocations, respectively. The MICAz $CC2420$ radio transceiver [32] uses IEEE 802.15.4 protocol [33] and transmits data in the $2.4GHz$ frequency band with a maximum data rate of $250kbits/s$. Other investigations have already looked at the effects of the transmission power on mote connectivity and battery power. A series of experiments were conducted in the environment to evaluate the effect of AIRF on the network load balance, network lifetime and sleep latency. Our study is more focused on how the data collection rate and multihop network functionality affects network lifetime and sleep latency. The default 0dBm (1mW) radio power setting in the $2.405 - 2.425$GHz frequency channel was used in this study. To simulate more realistic scenarios, we randomly deploy 600 nodes in a 1km $\times$ 1km area and we set the transmission range to 70m. The data message interval (DMI) and the data transmission interval, which are designated to be the same value, can be set to any multiple of seconds. Intervals were chosen from 10 to 800 seconds.

B. Experimental results and analysis

To obtain the experimental results, we generate 50 packets at each node and take the average on the measured delay in experiments. We define $f(x)$ to be a density function based on the number of bytes handled by server under load unbalanced, where x is calculated by formula (1):

$$x = \frac{\max(u) - \min(u)}{avg(u)} * 100 \quad (1)$$

where u is the number of bytes handled by each server during the interval of three seconds, $max(u)$, $min(u)$, and $avg(u)$ are the maximum, minimum, and average value of u, respectively. Fig. 4 shows the performance difference of the different strategies on network load balance. Experimental results show that, in most case, the non-balance values of the maximum load do not exceed 108% of the average. However, for directed diffusion and SPIN, the non-balance values of the maximum load exceed to 200%, even up to 400% of the average. But for AIRF, the non-balance values scarcely exceed 108% of the average. Therefore, we can obtain from the above experiments that the performance of AIRF on network load balance is more excellent than HOPS and DD methods.

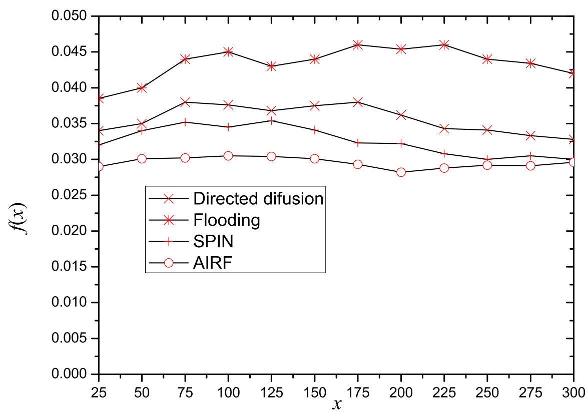

Fig. 4. Performance analysis on network load balance

In order to analyze the lifetime performance of AIRF, we assume that each anycast address is shared by $N(N = 1, 2, \cdots, 10)$ nodes in experiments. By changing the number of destination nodes, we analyze the lifetime performance of flooding, directed diffusion, SPIN, and AIRF, respectively. Experimental results are shown in Fig 5. Fig 5 shows that AIRF obtains an improvement of several orders of magnitude better than the other protocol tested. The reason is that the fact that messages are sent along a single routing path which eliminates energy consuming transmissions. The improvement in network lifetime for the network of 600 nodes is only three times that of the other protocols, due to the fact that every node in the network sends its data through the nodes surrounding the sink. Therefore, 599 messages are transmitted by eight nodes every fifteen minutes.

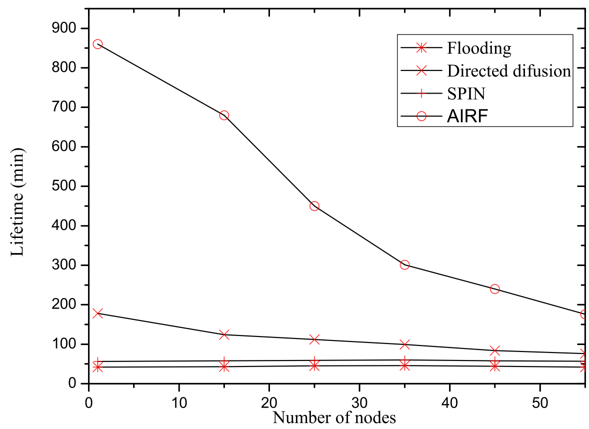

Fig. 5. Performance evaluation of lifetime

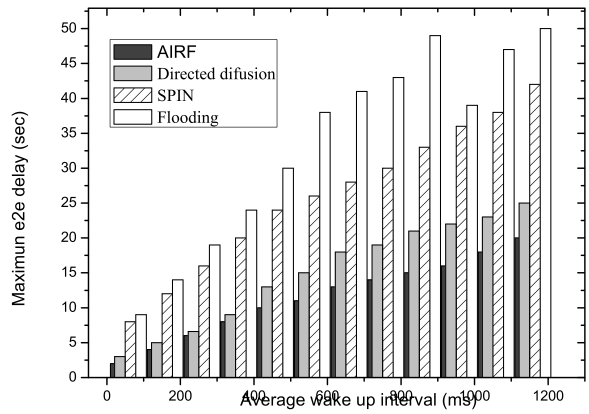

Fig. 6. Performance evaluation of transmission time

Finally, we compare the maximum expected end-to-end (e2e) delay over all nodes under different wake-up rates. The experimental results are shown In Fig. 6. We observe that AIRF significantly reduces the e2e delay compared with the other algorithms, such as flooding, directed diffusion and SPIN. Our results consistently show that AIRF is optimal on the performance of e2e delay.

VI. Conclusion and Future Work

We have designed and implemented AIRF algorithm, a robust anycast-based integrated routing framework for WSNs, to improve quality of service (QoS) of WSNs. AIRF focuses on network load balance, lifetime, and delay, which are vital to the survival of a WSN in a QoS-aware environment. With the idea of MAP table and FIS management, AIRF enables a node to keep track of the efficient route of its neighbors and thus to select a reliable route. The performance of AIRF is evaluated through experiments and experimental results show that AIRF has the better performance on lifetime, end-to-end delay, and network load than the other routing protocols, such as flooding, directed diffusion, and SPIN in highly dynamic scenarios.

Future work includes application of the proposed anycast-based integrated routing approach to incorporate multiple channels WSNs for each router to reduce co-channel interference. In addition, when several hosts sharing the same anycast address are connected to the same network link, we must modify ARP protocol or neighbor discovery protocol to insure data link layer address be mapped to a sharing anycast address, which is the emphasis in our further research.

Acknowledgment

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, Volume 38, issue 4, pp. 393-422, 2002.

[2] C. Buratti, A. Conti, D. Dardari, and R. Verdone, "An Overview onWireless Sensor Networks Technology and Evolution," *Sensors*, pp. 6869-6896, 2009.

[3] Y. Li, "A Quality of Service Anycast Routing Algorithm Based on Improved Ant Colony Optimization," *Journal of Computers*, Vol 8, No 4, pp. 968-974, 2013.

[4] S. Chen, H. Lee, C. Chen, H. Huang, and C. Luo, "Wireless Body Sensor Network With Adaptive Low-Power Design for Biometrics and Healthcare Applications," *IEEE Systems Journal*, pp. 398-409, 2009.

[5] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wireless Communications*, Vol. 11, pp. 6-28, 2004.

[6] F. A. Ian, M. Tommaso, and R.C. Kaushik, "A survey on wireless multimedia sensor networks," *Computer Networks*, Vol. 51, pp. 921-960.

[7] K. Kim, X. Lin, and Ness B. Shroff, "Optimal Anycast Technique for Delay-Sensitive Energy-Constrained Asynchronous Sensor Networks," *IEEE/ACM Transactions on Networking*, Vol. 19, Issue 2, pp. 484-497, 2011.

[8] H. Lu, X. Wang, and J. Su, "CCS: Collaborative Malware Clustering and Signature Generation using Malware Behavioral Analysis," *International Journal of Hybrid Information Technology*, Vol. 5, No. 2, pp. 147-152, April, 2012.

[9] S. Ren, H. Han, B. Li, J. Lu, C. Peng, and W, Dou, "An Improved Wireless Sensor Networks Routing Protocol Based on AODV," *IEEE 12th International Conference on Computer and Information Technology (CIT)*, pp. 742-746, 2012

[10] J. Kim, X. Lin, Ness B. Shroff, and P. Sinha, "Minimizing Delay and Maximizing Lifetime for Wireless Sensor Networks With Anycast," *IEEE/ACM Transations on Networking*, Vol. 18, No. 2, pp. 515-528, 2010.

[11] J. Kim, X. Lin, N. B. Shroff, and P. Sinha, "On Maximizing the Lifetime of Delay-Sensitive Wireless Sensor Networks with Anycast," *in Proceedings of IEEE INFOCOM*, pp. 807-815, Pheonix, AZ, April 2008.

[12] M. Zorzi and R.R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: multihop performance," *IEEE Transactions on Mobile Computing*, Volume 2, issue 4, pp. 337-348, 2004.

[13] S. Liu, K. Fan, and P. Sinha, "CMAC: An energy-efficient MAC layer protocol using convergent packet forwarding for wireless sensor networks," *ACM Transactions on Sensor Networks*, Volume 5, issue 4, Article No. 29, 2009.

[14] H. Tan and M. Chan, "A^2-MAC: An Adaptive, Anycast MAC Protocol for Wireless Sensor Networks," *in the IEEE Wireless Communications and Networking Conference (WCNC)*,pp. 1-6, 2010.

[15] R. R. Choudhury, and N. H. Vaidya, "MAC-layer anycasting in ad hoc networks," *SIGCOMM Comput. Commun. Rev.*, vol. 34, pp. 75-80, Jan. 2004.

[16] F. Ashraf, Nitin H. Vaidya, and Robin H. Kravets, "Any-MAC: Extending Any Asynchronous MAC with Anycast to Improve Delay in WSN," *in Proceedings of the 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2011)*, pp. 19-27, Salt Lake City, UT, USA, 2011.

[17] S. K. Chen and P. CH. Wang, "Design and Implementation of an Anycast Services Discovery in Mobile Ad Hoc Networks," *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 6, No. 1, Article 2: 1-9, 2011.

[18] S. K. Chen and P. C Wang, "An anycast-based emergency service for healthcare wireless sensor networks," *IEICE Trans. Comm.*, E93-B, 4, pp. 858-861, 2010.

[19] E. Guttman, "Service location protocol: automatic discovery of IP network services," *IEEE Internet Computing*, Vol. 3, Issue 4, pp. 71-80, 1999.

[20] G. Lu, B. Krishnamachari, and C. S. Raghavendra, "An adaptive energyefficient and low-latency MAC for data gathering in wireless sensor networks," *in the 18th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2004.

[21] S. Du, A. Saha, and D. Johnson, "RMAC: A routing-enhanced dutycycle mac protocol for wireless sensor networks," *in the IEEE 26th Conference on Computer Communications (INFOCOM)*, pp. 1478-1486, 2007.

[22] Y. Sun, S. Du, O. Gurewitz, and D. Johnson, "DW-MAC: a low latency, energy efficient demand-wakeup MAC protocol for wireless sensor networks," *in the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 53-62, 2008.

[23] J. Jiang, Y. Tseng, C. Hsu, and T. H. Lai, "Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks," *Mobile Networks and Applications*, Volume 10, issue 1-2, pp. 169-181, 2005.

[24] A. Keshavarzian, H. Lee, and L. Venkatraman, "Wakeup scheduling in wireless sensor networks," *in the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 322-333, 2006.

[25] K. Römer, "Time synchronization and calibration in wireless sensor networks," *In IFIP Mediterranean Workshop on Ad-Hoc Networks*, pp. 131-142, June 2003.

[26] Q. Lu, W. Luo, J. Wang, and B. Chen, "Low-complexity and energy efficient image compression scheme for wireless sensor networks," *Computer Networks*, Vol. 52, Issue 13, pp. 2594-2603, 2008.

[27] F. Liu, F. Liu, and H. Zhang, "An Energy-efficient and Real-time Anonymous Routing Protocol for Ad hoc Networks," *Journal of Computers*, Vol 7, No 4, pp. 900-912, 2012.

[28] J. Wang, Y. Zheng, C. Leung, and W. Jia, "A-DSR: A DSR-based Anycast Protocol for IPv6 Flow in Mobile Ad Hoc Networks," *Proceeding of Vehicular Technology Conference*, Vol. 5, pp. 3094-3098, 2003.

[29] M. Zorzi and R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: energy and latency performance," *The IEEE transactions on Mobile Computing (TMC)*, Volume 2, issue 4, pp. 349-365, 2003.

[30] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc Networks*, Volume 1, issue 4, pp. 351-369, 2003.

[31] V. Paruchuri, S. Basavaraju, A. Durresi, R. Kannan, and S. Iyengar, “Random asynchronous wakeup protocol for sensor networks,” *in The Fourth International Conference on Broadband Communications, Networks and Systems (BROADNETS)*, pp. 710-717, 2004.
[32] A. S. Chipcom, “SmartRF CC2420 Preliminary Datasheet,” 2004, http://inst.eecs.berkeley.edu/ cs150/Documents/CC2420.pdf.
[33] IEEE-SA Standards Board, “IEEE 802.15.4 Standard-2003, Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs),” 2003.
[34] C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” *in Proc. of the 6th International conference on Mobile Computing and Networking (Mobicom)*, Boston, USA, PP. 56-67, Aug. 2000.
[35] W. Heinzelman, J. Kulik, and H. Balakrishnam, “Adaptive protocolsfor information dissemination in wireless sensor networks,” *in Proc. of the 5th International conference on Mobile Computing and Networking (Mobicom)*, Seattle, USA, PP. 174-185, Aug. 1999.
[36] W. Wang, J. Zhang, Y. Tang, Y. Xiang, and T. Yang, “Leveraging 1-hop Neighborhood Knowledge for Connected Dominating Set in Wireless Sensor Networks,” *Journal of Computers: Special Issue: Parallel Algorithms, Scheduling and Architectures*, Vol 7, No 1, pp. 11-18, 2012.

Guofeng Yan received Ph.D. degrees from Central South University, Changsha, China in 2011. He is currently an Associate Professor of School of Computer and Communication, Hunan Institute of Engineering and National University of Defense Technology. His research interests are routing protocol for wireless networks, performance analysis on end-to-end communication in wireless networks, and cloud computing.

Yuxing Peng received the BS degree in computer science from Beihang University, China, in 1983, and the MS and DSc degrees in computer science from the National University of Defense Technology in 1986 and 1996, respectively. He is currently a full professor in the School of Computer Science at the National University of Defense Technology. His research interests include wireless networks, distributed systems, and P2P networks.

Shuhong Chen received her MS degree in computer science from Central South University, Changsha, China in 2004. She is currently a Ph.D. candidate of School of Information Science and Engineering at Central South University. Her research interests range over the various areas of wireless communication networks: trust computing, wireless resource optimization in sensor, and routing.

Junyi Liu received his B.S. degree in computer science from the National University of Defense Technology, China, in 2011. He is currently working towards his M.S. degree in computer science at National University of Defense Technology, China. His current research interest includes wireless network security and cloud computing.