EFFICIENT APPROACH FOR WIRELESS SENSOR NETWORKS TO IMPROVE LIFE TIME USING ANYCAST

Anusha Priyadharshini,Rashmi M R ,Shilpashree R L,Nirmala S, and Anitha S

Abstract**—** This paper describe the topic based on minimizing the delay and maximizing the lifetime of event-driven wireless sensor networks, for which events occur infrequently. In such systems, most of the energy is consumed when the radios are on, waiting for an arrival to occur. Sleep-wake scheduling is an effective mechanism to prolong the lifetime of these energy-constrained wireless sensor networks. However, sleep-wake scheduling could result in substantial delays because a transmitting node needs to wait for its next-hop relay node to wake up. An interesting line of work attempts to reduce these delays by developing any cast.-based packet forwarding schemes, where each node opportunistically forward]s a packet to the neighboring node that wakes up among multiple candidate nodes. In this paper, we first study how to optimize the any cast forwarding schemes for minimizing the expected packet-delivery delays from the sensor nodes to the sink. Based on this result, we then provide a solution to the joint control problem of how to optimally control the system parameters of the sleep-wake scheduling protocol and the any cast packet-forwarding protocol to maximize the network lifetime, subject to a constraint on the expected end to end packet-delivery delay. Our numerical results indicate that the proposed solution can outperform prior heuristic solutions in the literature, especially under the practical scenarios where there are obstructions, e.g., a lake or a mountain, in the coverage area of wireless sensor networks.

Keywords-component; event-driven; energy minimizing maximize; Anycast.

I. INTRODUCTION

We consider a wireless sensor network whose main function is to detect certain infrequent alarm events, and to forward[1] alarm packets to a base station, using geographical forwarding. The nodes know their locations, and they sleep-wake cycle, waking up periodically but not synchronously. In this situation, when a node has a packet to forward to the sink, there is a trade-off between how long able neighbor to wake up and the progress the packet makes towards the sink once it is forwarded to this neighbor. Hence, in choosing a relay node, we consider the problem of minimizing[8] average delay subject to a constraint on the average progress. By constraint relaxation, we formulate this next hop relay selection problem as a Markov decision process (MDP). The exact optimal solution (BF (Best Forward)) can be found, but is computationally intensive. Next, we consider a mathematically simplified model for which the optimal policy (SF (Simplified Forward)) turns out to be a simple one-step-look-ahead rule. Simulations show that SF is very close in performance to BF, even for reasonably small node density. We then study the end-to-end performance of SF in comparison with two extremal policies: Max Forward (MF) and First Forward [1](FF), and an end-to-end delay minimizing policy[1,8]. We find that, with appropriate choice of one hop average progress constraint, SF can be tuned to provide a favorable trade-off between end-to-end packet delay and the number of hops in the forwarding path.

Anycast[3] routing[6] is very useful for many applications such as resource discovery in Delay Tolerant Networks (DTNs)[2]. In this paper, based on a new DTN model, we first analyze the anycast semantics for DTNs. Then we present a novel metric named EMDDA (Expected Multi-Destination Delay for Any cast) and a corresponding routing algorithm for anycast routing in DTNs. Extensive simulation results show that the proposed EMDDA routing scheme can effectively improve the efficiency of anycast routing in DTNs. It outperforms another algorithm, Minimum Expected Delay (MED) algorithm, by 11.3% on average in term of routing delays and by 19.2% in term of average max queue length.

Anusha Priyadharshini is student of IIIrd Sem M.Tech Computer Science Engineering Dept,,VTU University at AMC Engineering College(e-mail: anu_priya07@yahoo.co.in).

Rashmi M R is student of IIIrd Sem M.Tech computer Science and Engineering ,VTU University at AMC Engineering College.

Ms.Shilpashree is student of IIIrd sem M.Tech Computer Science and Engineering ,VTU university at AMC Engineering College Bangalore INDIA

Ms Nirmala is Associate Professor in Dept of Computer Science and Engineering AMC Engineering College Bangalore INDIA (e-mail: Esthernirmala70jc@yahoo.co.in)

Ms Anitha is Assistant Professor in Dept of Computer Science and Enigneering AMC Engineering College Bangalore INDIA (e-mail: sbanithait@gmail.com)

II. SYSTEM MODEL

The energy[7] consumption throughout the lifespan of a network node, an interesting observation is that, in current surveillance systems, most of the energy is "wasted" in the sense that it is used in operations that do not actively fulfill the system's purpose. For example, a node without power management is always turned on, but there is no target most of the time. Hence, most of its energy is dissipated in a waiting status. Fig. 1 shows the distribution of energy consumption in a typical surveillance network. As the figure shows, only one per cent of the energy is used in actually tracking targets, the other 99% of the energy is used in waiting for targets to show up. With a rotation based power management, the energy efficiency is much better. Fig. 2 shows the energy distribution. Throughout the network's lifespan, 21% of the energy is used in really tracking targets, and 7% is used in sleep [5] mode. However, 72% of energy is still wasted in a waiting status because the node periodically wakes up to listen to potential wake-up signals, or continuously operates in a low-power[5] stand-by listening mode.

Figure 1. Analysis of Energy Distribution with the Always-On Scheme

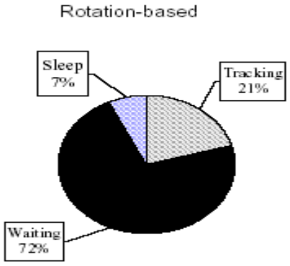


Figure 2. Analysis of Energy Distribution with the Rotation Based Scheme

is that, in current surveillance systems, most of the energy is "wasted" in the sense that it is used in operations that do not actively fulfill the system's purpose. For example, a node without power management is always turned on, but there is no target most of the time. Hence, most of its energy[7] is dissipated in a waiting status. Fig. 1 shows the distribution of energy consumption[5] in a typical surveillance network, of the energy saving should come from eliminating the energy consumed in the waiting status. Aiming at this goal, we propose another approach to power management. The events of interest in a system often contain energy, and the moment when an event happens is exactly the moment when we want the system to enter wake-up mode. So we can potentially use the energy in the event to trigger the transition of the system from sleep[5] mode to wake-up mode.

A. Anycast Forwarding[1][3] and Sleep[5],[4]-Wake Scheduling Policies In this model, there are three control variables that affect the network lifetime and the end-to-end delay[2] experienced by a packet: wake-up rates, forwarding set, and priority.1) Wake-up rates: The sleep-wake schedule is determined by the wake-up rate ¸j of the Poisson process with which each node j wakes up. If ¸j increases, the expected one-hop delay will decrease, and so will the end-to-end delay of any routing[6] paths that pass though node j. However, a larger wake-up rate leads to higher energy consumption and reduced network lifetime.In the rest of the paper, it is more convenient to work with the notion of awake probability which is a function of ¸j . Suppose that node i sends the first beacon signal at time 0. If no nodes in Fi have heard the first m¡1 beacon and ID signals, then node i transmits the m-th beacon and ID signal in the time-interval [(tB +tC +tA)(m¡1); (tB +tC + tA)(m¡1)+tB+tC]. For a neighboring node j to hear the m- th signals and to recognize the sender, it should wake up during [(tB +tC +tA)(m¡1)¡tA¡tC; (tB +tC +tA)m¡tA¡tC]. Therefore, provided that node i is sending the m-th signal, the probability that node j 2 Ci wakes up and hears this signal is pj = 1 ¡ e¡¸j (tB+tC+tA): (1) We call pj the awake probability of node j.

2) Forwarding Set: The forwarding set Fi is the set of candidate nodes chosen to forward a packet at node i. In principle, the forwarding set should contain nodes that can quickly deliver the packet to the sink. However, since the end-to-end delay depends on the forwarding set of all nodes along the possible routing[6] paths, the optimal choices of forwarding sets of these nodes are correlated. We use a matrix A to represent the forwarding set of all nodes collectively, as follows: A , [aij ; i = 1; :::;N; j = 1; :::;N]; where aij = 1 if j is in node i's forwarding set, and aij = 0 otherwise. We call this matrix A the forwarding matrix. Reciprocally, we define Fi(A) as the forwarding set of node i under forwarding matrix A, i.e., Fi(A) = fj 2 Cijaij = 1g. We let A denote the set of all possible forwarding matrices. With anycast[3] a forwarding matrix determines the paths that packets can potentially traverse. Let g(A) be the directed graph G(V;E(A)) with the set of vertices V = N, and the set of edges E(A) = f(i; j)jj 2 Fi(A)g. If there is a path in g(A) that leads from node i to node j, we say that node i is connected to node j under the forwarding matrix A.Otherwise, we call it disconnected from node j. An acyclic path is the path that does not traverse any node more than once. If g(A) has any cyclic path, we call it a cyclic graph, otherwise we call it an acyclic graph.

3) Priority: Let bij denote the priority of node j from the viewpoint of node i. Then, we define the priority assignment of node i as ~bi = (bi1; bi2; ¢ ¢ ¢ ; biN), where each node j 2 Ci is assigned a unique number bij from 1; ¢ ¢ ¢ ; jCij, and bij = 0 for nodes j =2 Ci. When multiple nodes send an acknowledgement after the same ID signal, the source node i will pick the highest priority node among them as a nexthop node. Although only the nodes in a forwarding set need priorities, we assign priorities to all nodes to make the priority assignment an independent control variable from

forwarding matrix A. Clearly, the priority assignments of nodes will also affect the expected delay[2]. In order to represent the global priority decision, we next define a priority matrix B as follows: B , [bij ; i = 1; :::;N; j = 1; :::;N]; We let B denote the set of all possible priority matrices.

III. MINIMIZATION OF END-TO-END DELAYS

A. Anycasting in Mobile Networks

Anycast technology and related dynamic routing functionality provide significant improvements to mobile network architectures. A major difficulty that exists in today's military and commercial mobile networks is managing mobile nodes and services under dynamic changing conditions. Also, with progress of internetworking technology, more distributed services are being deployed and relied upon by end users and applications. Anycasting helps provide a robust means of dynamically managing the end user requirement of finding "one service point out of a set".

Modern networks often include multiple levels and types of distributed services and applications that end users need to periodically contact, potentially exchange data with, and/or continuously provide data reports to. These services and applications may provide such functions as security key management, application directory services, name/address resolution or data collection and fusion.

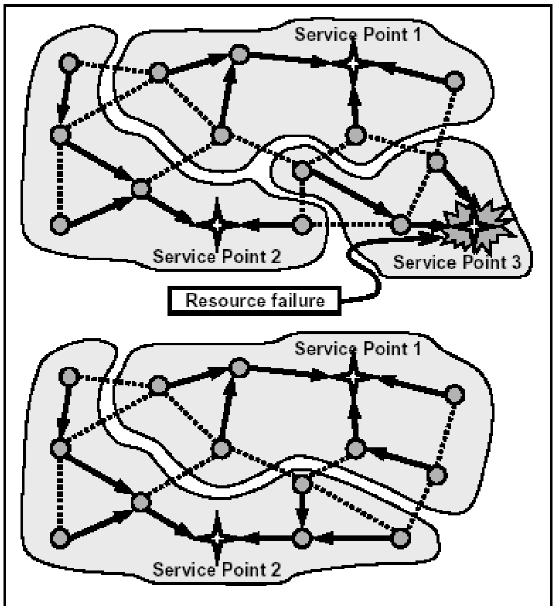
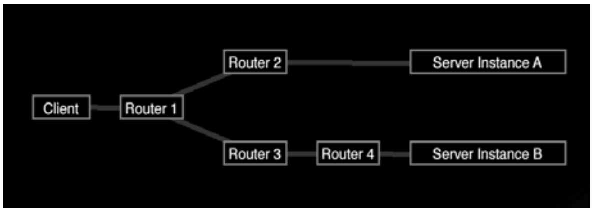
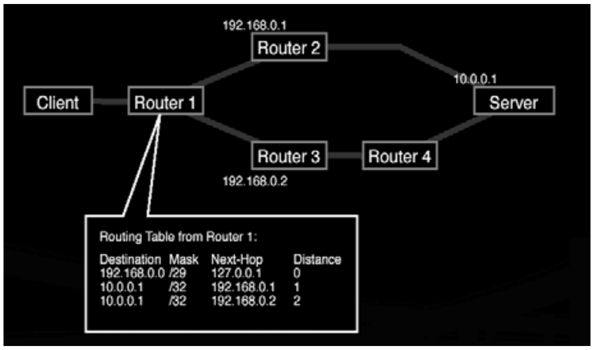


Fig. 3.1. Conceptual illustration of the dynamic service areas defined by the anycast routing[6] protocol before and after a resource failure.

B. How Does Anycast Work?

The basic idea is extremely simple: Multiple instances of a service share the same IP address. The routing infrastructure directs any packetto the topologically nearest instance of the service. What little complexity exists is in the optional details.





Routing Table from Router 1:

Destination	Mask	Next-Hop	Distance
192.168.0.0	/29	127.0.0.1	0
10.0.0.1	/32	192.168.0.1	1
10.0.0.1	/32	192.168.0.2	2

Fig 3.2 Anycast Work

IV. MAXIMIZATION OF NETWORK LIFETIME

1. In this System we are using one Client, five Routers and one Server.

2.A Client can able to send a file to server and the file will pass through any one of the Router using sleep and wake up method and any cast method:

TABLE I ROUTING TABLE

Router	*IP.Address*	*Port*	*Mode*	*File_Flag*	*Cur_Fla*
A			*Live*	*0*	*0*
B			*Live*	*1*	*1*
C			*Live*	*1*	*0*
D			*Live*	*0*	*0*
E			*Live*	*0*	*0*

Sleep and Wakeup Method

1.All the Router are inbuilt with sleep and awake up technology, that means if the router is not getting the packet for certain interval of time it will go to sleep[5]mode, when need arise the routers are switch back to wake-up mode.

2.We need a monitoring program. It will check the routing table every 60 seconds.

a .If file_flag is 0 then make the status of respective router "sleep".

b. Reset the file_flag for all the router to 0

Any cast

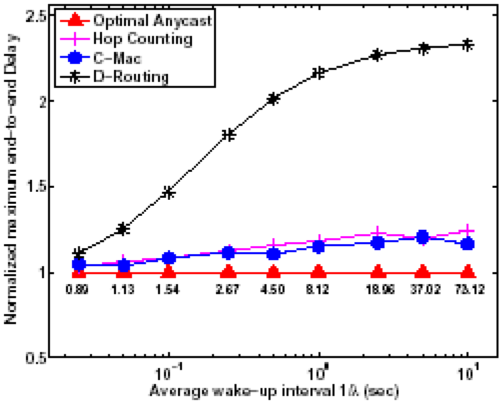
1.When Client is sending a file to server, this any cast system will check the Router which is in Live mode, suppose there are many Routers are available then it will calculate the distance (cost) of the Router and pick the Router which having less distance. If all Routers are in sleep mode means, this method will calculate the distance for the all Router and wakeup[4] (i.e converting the status from "sleep" to "Live") the Router which are having less distance, through the file will transfer to the server.

2 .Note: Set the "Cur_Flag" to 1 for the Router through which the file has to be transfer and Set 0 to all other Routers.

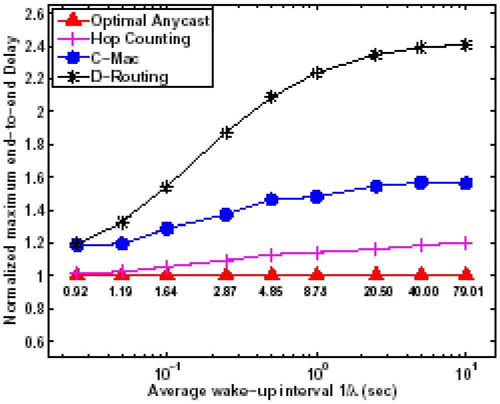
3. Set the "File_Flag" to 1 for the Router through which the file has to be transfer.

V. SIMULATION RESULTS

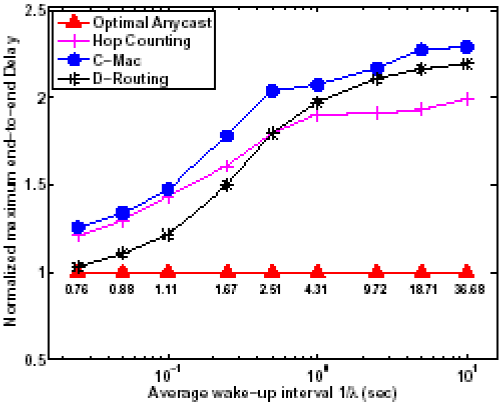
In this section, we provide simulation results to compare the performance of the optimal anycast[3] algorithm and the following algorithms.



(a) Nodes with the same wake-up rate are uniformly deployed



(b) Nodes with the same wake-up rate are non-uniformly deployed



(c) Nodes with the different wake-up rates are non-uniformly deployed

Fig. 3. The maximum end-to-end delay under each algorithm normalized by that under 'Optimal anycast' when (a) 400 nodes with the same wake-up rate ¸are uniformly deployed, (b) 391 nodes with the same wake-up rate ¸ are deployed forming a connectivity hole, and (c) 391 nodes with different wake-up rates are deployed forming a connectivity hole. The numbers beneath the line of 'Optimal Anycast' are the delay values (sec) under the optimal anycast algorithm..

C-MAC: The C-MAC algorithm proposed is an anycast-based heuristic that exploits geographic information to reduce the delay[2] from each node. Let di be the Euclidean distance from node i to sink s. Further, let rij be the geographical progress toward the sink, i.e., if node i forwards the packet to node j, the progress is defined as rij = di ¡ dj . If a node has a packet, let D be the one-hop delay from node i to a next-hop node, and let R be the progress between two nodes. Since node i selects the next-hop node probabilistically, both D and R are random variables. The objective of the CMAC algorithm is to find the forwarding set that minimizes the expectation of normalized one-hop delay, i.e., E[D R].The idea behind this algorithm is to minimize the expected delay per unit distance of progress, which might help to reduce the actual end-to-end delay.

Hop-counting Algorithm: We also compare with a heuristic hop-counting algorithm that we have developed that exploits the hop count (the minimum number of hops to reach the sink) of neighboring nodes to reduce the end-to-end delay[2].The objective of this algorithm is to minimize the time for a packet to advance one hop closer to the sink. This algorithm is inspired by the original hop-counting algorithms in [19], [20]. 5 If an h-hop node i has a packet to transmit, it waits until any (h¡1)- or h-hop neighboring node wakes up. If an (h¡1)-hop node wakes up first, then the packet is transmitted to the (h ¡ 1)-hop node. If an h-hop node j wakes up first, node i has to decide whether it transmits the packet to node j or it waits for an (h ¡ 1)-hop node to wake up. Such a decision is made by comparing the corresponding expected delays. If node j is chosen, the expected delay is given by tD + tI 1¡Qj02C (h¡1) j(1¡pj0) + tD. (The three terms in the summation correspond to the

time to transmit the packet to node j, the expected time for node j to wait for another (h¡1)- hop neighboring node to wake up, and the time to transmit the packet to the (h ¡ 1)-hop node, respectively, where C(h¡1) j is the set of (h ¡ 1)-hop neighboring nodes of node j.) If node i waits for an (h ¡ 1)-hop node, the expected delay is
tI 1¡Qj02C (h¡1) i (1¡pj0) +tD. Hence, node i chooses the decision with the smaller expected delay. Deterministic Routing[6] (D-Routing): By deterministic routing, we mean that each node has only one designated nexthop forwarding node. To find the delay-optimal routing path, we use the well-known Bellman-Form algorithm, in which the length of each link (i; j) is given by the expected one-hop delay tI=pj + tD. Comparing this algorithm with the others, we will study how exploiting path diversity can help to reduce the end-to-end delay. A. Case 1: Uniformly deployed homogeneous nodes We first simulate a wireless sensor network with 400 uniformly deployed nodes over an 1km-by-1km area with the sink s located at the lower-left corner. We assume that the transmission range from each node i is a disc with radius 100m. The parameters tI and tD are set to 6ms and 30ms, respectively. We also assume that all nodes are homogeneous

VI. CONCLUSION

We develop an anycast packet-forwarding[1] scheme to reduce the event-reporting delay and to prolong the lifetime of wireless sensor networks employing asynchronous sleep-wake scheduling. Specifically, we study two optimization problems. First, when the wake-up rates of the sensor nodes are given, we develop an efficient and distributed algorithm to minimize the expected event-reporting delay from all sensor nodes to the sink. Second, using a specific definition of the network lifetime, we study the lifetime-maximization problem to optimally control the sleep-wake scheduling policy and the anycast policy, in order to maximize the network lifetime subject to a upper limit on the expected end-to-end delay. Our numerical results suggest that the proposed solution can substantially outperform prior heuristic solutions in the literature under practical scenarios where there are obstructions in the coverage area of the wireless sensor network.

REFERENCES

[1] Tunable Locally-Optimal Geographical Forwarding in Wireless Sensor Networks with Sleep-Wake Cycling Nodes K.P. Naveen and Anurag Kumar Dept. of E.C.E., Indian Institute of Science, Bangalore 560 012, India.

[2] Anycast Routing in Delay Tolerant Networks,Yili Gong†, Yongqiang Xiong, Qian Zhang, Zhensheng Zhang, Wenjie Wang, Zhiwei Xu.

[3] Anycast Routing for Mobile Networking.Vincent D. Park and Joseph P. Macker,Naval Research Laboratory,Washington, DC 20375.

[4] Performance Analysis of Sensor Nodes in a WSN With Sleep/Wakeup Protocol ,Wuyi Yue1 Qingtian Sun2 Shunfu Jin2,1Department of Intelligence and Informatics,Konan University, Kobe 658-8501, Japan.

[5] Radio-Triggered Wake-Up forWireless Sensor Networks,Lin Gu and John A. Stankovic,Department of Computer Science, University of Virginia.

[6] Best Practices in IPv4 Anycast Routing Version 1.0 August, 2002 Bill Woodcock Packet Clearing House.

[7] Optimal Anycast Technique for Delay-Sensitive Energy[7]-Constrained Asynchronous Sensor Networks,Joohwan Kim□, Xiaojun Liny, and Ness B. Shroffz.

[8] Minimizing Delay and Maximizing Lifetime for Wireless Sensor Networks With Anycast Joohwan Kim, Xiaojun Lin, Member, IEEE, Ness B. Shroff, Fellow, IEEE, and Prasun Sinha.