

# A COMMUNICATION PARADIGM FOR HYBRID SENSOR/ACTUATOR NETWORKS

Wen Hu<sup>1</sup>, Nirupama Bulusu<sup>2</sup>, Sanjay Jha<sup>1</sup>

<sup>1</sup> The University of NSW and National ICT Australia Limited, {wenh, sjha}@cse.unsw.edu.au

<sup>2</sup> National ICT Australia Limited, {nbulusu}@cse.unsw.edu.au

**Abstract** - This paper investigates an anycast communication paradigm for a *hybrid* sensor/actuator network, consisting of both resource-rich and resource-impooverished devices. The key idea is to exploit the capabilities of resource-rich devices (called micro-servers) to reduce the communication burden on smaller, energy, bandwidth and memory constrained sensor nodes. The goal is to deliver sensor data to the nearest micro-server, which can (i) store it (ii) forward it to other micro-servers using out-of-band communication or (iii) perform the desired actuation. Our approach is to construct an anycast tree rooted at each potential event source, which micro-servers can dynamically join and leave. Our anycast mechanism is self-organizing, distributed, robust, scalable, and incurs very little overhead. ns-2 simulations show that our anycast mechanism can reduce network energy consumption by more than 50%, both the mean end-to-end latency of the transmission and the mean number of transmissions by more than 50%, and achieves 99% data delivery rate for low and moderate micro-server mobility rate.

## I. INTRODUCTION

This paper investigates an anycast service for hybrid sensor/actuator networks. In the last couple of years, sensor networks research has addressed the development of sensor platforms[5], application domains, and algorithms. Because sensor networks depend on multiple nodes cooperating with each other, an effective communication paradigm is of prime importance and has been researched upon[10][6][3][9].

Noteworthy communication paradigms are: (i) Directed Diffusion[6], a general purpose, network-oriented approach to data-centric communication in sensor networks (ii) IDSQ[9], an information-oriented approach that combines data routing with information optimization objectives, and (iii) TAG[10], a database oriented approach to address numerous sensors in aggregate by means of SQL queries and gather the data back to a single, central server.

Today's Internet combines different devices such as routers, servers and hosts, even the routers can be classified into different categories (e.g., into core routers and edge routers). Large scale sensor networks may have thousands of nodes in the future. It is more realistic to have hierarchical models of network devices rather than flat ones.

Previously proposed data routing protocols for sensor networks have not been designed to leverage the capabilities of hybrid devices by exploiting resource-rich devices to reduce the communication burden on smaller, energy, bandwidth, memory and computation-constrained sensor devices.

Consequently, they may not be best suited for hybrid sensor network applications involving several mutually cooperative sinks. Our hypothesis is that an anycast communication paradigm can provide significantly improved performance for any application involving a hybrid of resource-rich specialized nodes with small sensor devices. The resource-rich nodes provide some service such as (i) long-range data communications, (ii) persistent data storage, or (iii) actuation. Examples of actuation would be re-charging or replacing small nodes whose energy has been depleted, imagers which can take photos or video when activated by sensors, sprinklers which can sprinkle water in badly parched areas etc. The resource-rich node acts as a sink, and we call it a micro-server. The service is more important than the server providing it.

In a battlefield, mobile soldiers may be equipped with additional powerful data transmitters (out of band higher-range radios) than ground-based sensors. It is effective to simply report information (e.g. enemy detection, land mind presence, convoy vehicles) to the *nearest* available soldier, who can forward it to the other soldiers.

We wish to design an anycast service that can extend system lifetime, reduce end-to-end latency and improve network scalability. Our design goals are:

- *Simple*: To accommodate small sensor nodes, anycast must be computation and memory efficient.
- *Energy-efficient*: Anycast must incur minimal energy overhead for control as well as data communications.
- *Self-organizing and Adaptive*: To be responsive to sinks joining and leaving dynamically; and robust to node failures, anycast must be self-organizing and adaptive.
- *Distributed*: To scale to arbitrarily large sensor networks, anycast must be completely distributed.

*Contributions*: It is challenging to design an anycast communication service that is simple to implement and incurs low overhead, while also being self-organizing and robust. Our approach is to construct an anycast tree rooted at each event source, which micro-servers can dynamically join (by flooding route discovery interests) and leave. Data is delivered to the nearest micro-server on the tree. We motivate and propose a tree-based anycast mechanism (Section III). Using extensive simulations, we demonstrate its benefits in conserving energy, improving latency (Section IV).

## II. RELATED WORK

In this section, we cover research in data dissemination most directly relevant to our approach.

*Directed Diffusion:* Directed Diffusion (see [6] for more details) is a data-centric, reverse-path based communication paradigm for sensor networks. Sinks flood their *interests* into the network when they join the network. An interest is a query specifying the attributes of the information a sink wants a sensor to collect and respond. Sources in turn flood the first few exploratory data packets into the network. Sinks select and reinforce the best paths and the sources use reverse best paths to deliver data back to the sinks.

*Two-tier Data Dissemination (TTDD):* Two-tier data dissemination mechanism [3] tries to set up a virtual grid by calculating the distance between sensors and relaying spots. The sensor with minimum distance becomes a relaying point. The sources broadcast their query/interest within the grid and the query/interests are forwarded by the relaying sensors to the sources. The sources transfer the data packets along the reverse path to the sinks. Compared to Directed Diffusion, it can better handle sink mobility because the query/interest is limited in one local grid. However, it may still introduce replicate data packets transmission to multiple sinks.

*Manycast:* Manycast[2] is a recently proposed group communication scheme for ad hoc networks. However manycast allows a source to communicate with many destinations *simultaneously*. We believe that our approach of getting data to the nearest sink, and then forwarding it to other sinks using out-of-band communication is more suited to large sensor networks.

*Internet anycast:* The Internet community has addressed anycast research extensively [4] [7]. However, the environment is radically more dynamic in sensor networks; and sensor nodes have significantly limited resources.

*Multi-robot coordination:* Within the field of distributed mobile robotics, Daniela Rus *et al* [8] have addressed the problem of maintaining continuous communication to route data amongst mobile robots. Their work is complementary to ours; once data reaches the nearest micro-server using our anycast mechanism; such techniques may be used to forward the data to other mobile nodes.

*Summary:* Previous sensor communications research has neither exploited hybrid device capabilities such as out-of-band communication nor explored anycast services for sensor networks. Previous anycast mechanisms proposed for the Internet and ad hoc networks do not work as effectively in the sensor networks domain where data sources are a function of system events. Our novel reverse tree-based anycast mechanism, described next, is tailored to deal with the unique constraints and event dynamics of sensor networks.

## III. TREE-BASED ANYCAST

In this section, we describe the design rationale and details of our anycast mechanism.

### A. Design Rationale

We assume a *hybrid* sensor network consisting of both resource-rich micro-server nodes and low-power sensor nodes. Further we assume that there are multiple micro-servers (sinks) interested in the same data. Sinks could be mobile. Data needs to only reach one sink, thus motivating an anycast service. We assume that sensor network applications can handle small amounts of data loss; and therefore anycast does not need to explicitly provide reliable data delivery.

We want to provide an anycast mechanism that is scalable, self-organizing, robust, simple and energy-efficient. A straight-forward approach to implement anycast is using an *expanding-ring search* with feedback from micro-servers. This is attractive because it is self-organizing and robust, requires minimal network state and can limit the flooding scope in diffusion. On the other hand it is not well suited to handle sink mobility. It incurs high latency and energy overhead if a sink leaves (moves away) because it must discover a route to an alternate nearby sink. Moreover, it may require sinks to synchronize with each other before sending feedback to the event source.

Instead we adopt a shared tree approach. Corresponding to each event source, a *shortest-path tree* rooted at the source is constructed. Sinks form the leaves of the tree. Sinks can dynamically join or leave the anycast tree. Although this approach requires more network state, it is a good approach to handling mobility, as it simultaneously maintains paths to all sinks. By eliminating the need to discover paths to alternate sinks each time a sink leaves, it can reduce worst-case latency and does not require synchronization among sinks.

### B. Algorithm Details

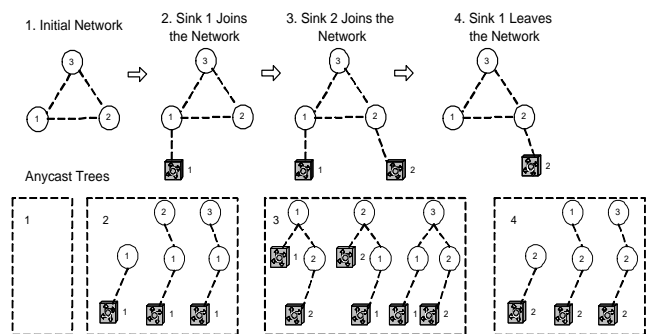


Fig. 1. Illustration of the anycast mechanism. The lower, boxed pictures show the structure of each anycast tree as two sinks join and leave a sensor network.

We now describe the details of our tree-based anycast mechanism. Fig. 1 shows how each anycast tree evolves when two sinks join and leave a sensor network.

*(Reverse) Tree Formation:* Every sensor node forms a potential event source. Therefore, corresponding to every sensor

node  $i$  in the network, there is an anycast tree  $T_i$  rooted at that sensor node. Each anycast tree is built from the leaves to the root. When sink  $S$  enters the network, a new branch leading to the sink must be added to each anycast tree. To minimize sensor energy consumption, information for building the branches is *piggybacked* with the route-discovery packet (e.g. interest packets of Directed Diffusion). To calculate the cost of the branch, sink initializes a *cost field*  $c$  (e.g. hop-count) in the route-discovery packet. Upon receiving this packet from sink  $S$ , each node  $i$  updates its anycast table by setting cost  $c(i, S)$  of the branch to  $S$  to be  $c$ . It increments the cost  $c$  before forwarding this packet. Eventually, a new branch (with cost) to sink  $S$  is added to each tree. To handle the memory constraint of sensors, an upper-bound can be added to limit anycast table size.

*Sink Leaves:* We adopt a soft state approach wherein we associate a timer with each anycast table entry - an entry is deleted when the timer expires.

*Data Delivery and Path Maintenance:* After initial set up, when a data packet arrives, a sensor looks up its anycast table for the sink with minimum cost before it forwards the packet. Therefore, the packet will be sent to only the nearest sink instead of to all sinks (as would be the case with Directed Diffusion). Sinks periodically send packets to refresh the anycast table entries. Stale entries are deleted when the related timer expires.

*Sink Mobility:* A sink may move out of range of its immediate upstream node. Some data packets may be delivered along this route until an (i) alternate route to this sink or (ii) an alternate route to an alternate nearby sink are found during next periodic route refresh. Our anycast algorithm does not implement delivery reliability explicitly because most sensor applications are inherently loss-tolerant to small amounts of loss. Reliability can always be implemented at application layer if necessary.

*Scalability:* The size of the anycast table in each sensor node is independent of the number of sources, but would increase linearly with the number of sinks in the network. We observe that since we are only interested in delivering packets to nearest sink, it is not necessary to maintain paths to all sinks in every anycast table. Therefore, the size of the table can be limited. We limit the size to 3 in our simulations. This enables our algorithm to be scalable in terms of the number of sources and the number of sinks. Moreover, it enables us to accommodate the memory constraints of small sensor devices. The sensor network has shorter paths and less number of packet-transmissions with anycast. This can reduce packet-collisions and delivery-latency.

*Differences from Internet Anycast:* The servers (leaves) join before the client/host (root) in Internet anycast. Therefore, the anycast tree is built when the leaf joins the tree. In sensor networks, it is more dynamic as event sources (root) could join the tree earlier or later than micro-servers. The tree is built via a reverse approach of route discovery (interest flooding) in our algorithm. Moreover, the hop-count informa-

tion is stored in the routing tables of Internet routers. A tree can be built by routing lookup. In sensor networks, hop-count information must be built from scratch. We piggyback this information with the route discovery packet (Interest) to save energy. Finally, in sensor networks, the size of the anycast routing table must be limited due to memory and computation constraints of small nodes; which is not a consideration in Internet anycast.

## IV. EVALUATION

### A. Goals, Metrics and Methodology

The goals of our evaluation are to study whether anycast can (i) lead to significant energy savings in comparison to traditional protocols such as Directed Diffusion (ii) improve the end-to-end latency in data transfer (iii) improve network scalability, and (iv) handle moderate sink mobility.

We use several metrics for evaluation.

- *Mean energy consumption:* We study this metric as a function of time. This metric characterizes the mean energy consumed per node at any given instant of time. Ideally it should be as low as possible.
- *Jitter in energy consumption:* The jitter percentage for each node is the percentage difference between the energy consumed by the particular node and the mean energy consumption. Ideally, the jitter should be close to zero so as to load balance energy consumption equally across all nodes. We study jitter across all nodes.
- *End-to-end latency:* We study this metric as a function of network size. This metric characterizes the cumulative latency for data to reach from its source to its destination. Ideally, this metric should be as small as possible to indicate timely data transfer.
- *Mean Path Length:* We study this metric as a function of network size. This metric denotes the mean number of hops traversed by a data packet. Ideally, this metric should be as small as possible for lower energy consumption across small nodes.
- *Data delivery rate:* We study this metric as a function of sink speed. This metric characterizes the percentage of event source data packets successfully delivered to at least one sink. Ideally, this should be 100%.

1) *ns-2 implementation:* We added an anycast filter to the Directed Diffusion implementation in ns-2 [1] which enables Directed Diffusion to build and maintain the anycast tree and deliver the data packet to exactly one of the sinks.

2) *Simulation parameters:* To understand how anycast can affect the above network metrics, we simulated random topologies with varying network sizes — 50, 100, 150, 200, 250 and 300 sensor nodes. Three sources are chosen randomly for each scenario. IEEE 802.11 is the MAC protocol. The transmission range is 100m; the initial energy in the sensors is 1000 Joules. Other simulation parameters are summarized in Table 1.

Table 1  
Simulation Parameters. The area size is set such that there are two sensors in each communication unit.

Nodes	Area	Sinks
50	350m × 350m	2
100	500m × 500m	2
150	600m × 600m	3
200	700m × 700m	4
250	790m × 790m	5
300	860m × 860m	6

To study the impact of sink mobility, 100 sensors (with 3 sources) and three sinks are randomly deployed in an area of 500m \* 500m. Sink speeds are varied from 1m/s to 100m/s.

Simulation time is 1002 seconds (which is sufficient to characterize protocol trends). Within the first 20 seconds, sources and sinks publish or subscribe the same interest in a random sequence. After publication, a source generates a data packet every three seconds. Sensor energy levels are logged every 10 seconds.

### B. Results

Fig. 2, 3 plot the mean energy consumption as a function of time. Energy consumption with anycast increases at a much slower rate; and is at least 60% lower with anycast (for 300 nodes) when simulations end. Anycast savings are more significant for larger network sizes (upto 60% for 300 nodes scenario) than for smaller network sizes (up to 40% for 150 nodes scenario). The results show that anycast mechanism enables energy savings for two reasons: (i) data is delivered to only one of the sinks instead of all the sinks (ii) the data delivery path is shorter so that fewer data transmissions per path are required.

Fig. 4, 5 plot the jitter with Directed Diffusion and anycast respectively across all nodes. Jitter is significantly lower with anycast (mainly between -15% and 10%) than with Directed Diffusion (mainly between -40% and 20%) because anycast forwards the data traffic locally to the nearest micro-server for each source; thereby distributing the load more evenly across the network when compared to Directed Diffusion where data traffic is global and burdens nodes in the middle of the network more heavily.

Fig. 6, 7 plot the end-to-end latency and the mean path length as a function of network size. As expected, the end-to-end latency and the mean path length for both Directed Diffusion and anycast increases with the network size. However, the increase is significantly less with anycast because the mean path length is smaller as data is only forwarded to the nearest sink. For 300 nodes, the end-to-end latency with anycast is nearly 60% lower than with Directed Diffusion.

These results show that the major contribution to reducing mean end-to-end latency are the reduced mean path lengths. Fig. 8 plots the data delivery rate as a function of sink speed for a network topology of size 100 nodes. We expect human

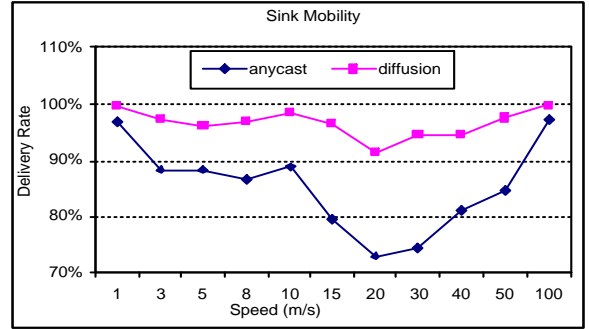


Fig. 8. Delivery rate vs mobile sinks' speed with 100 nodes in an area of 500 \* 500

and robotic mobility for many sensor network applications to be approximately 1 m/s (3.6 kmph). Our current anycast algorithm achieves 99% data delivery rates for this mobility regime, and can therefore accommodate it. However, the data delivery rate for anycast drops off to 73% as the sink speed increases to 20m/s. We conjecture that the subsequent increase in data delivery rate for higher sink speeds is an artifact of limited-size terrain. A complete study of mobility effects on anycast is beyond the scope of this paper. However, we are exploring how to extend our scheme to accommodate high-mobility micro-servers.

### C. Summary of Results

To summarize, our simulations show that our anycast service when added to Directed Diffusion can: (i) significantly reduce end-to-end latency (ii) significantly reduce energy consumption (iv) balance network load (energy consumption) more evenly by forwarding data traffic locally rather than globally, and (iv) handle low to moderate sink mobility with minimal extensions (as evidenced by the high data delivery rates achieved) but may require further modifications to handle higher mobility rates. While radio links work very well in simulation; they can be notoriously lossy in practice. These results need to be further validated experimentally.

## V. CONCLUSIONS

In this paper, we proposed and evaluated a tree-based anycast mechanism for *hybrid* sensor networks that is self-organizing, distributed, robust, scalable, and incurs low overhead. The key idea was to construct an anycast tree rooted at each event source, which micro-servers can dynamically join and leave. Data is delivered to the nearest micro-server on the tree. We exploit the out-of-band communication of micro-server nodes to forward data to other micro-servers, if necessary. Our evaluations demonstrate the benefits of anycast. In particular, we noticed a significant reduction (over 50% for the simulated scenarios) in end-to-end latency, mean energy consumption, and number of data transmissions. Moreover,

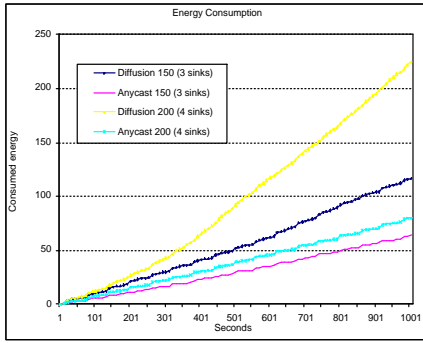


Fig. 2. Energy Consumption of 150/200 nodes in  $600 * 600/700 * 700$  areas (2 percent sinks and 3 sources).

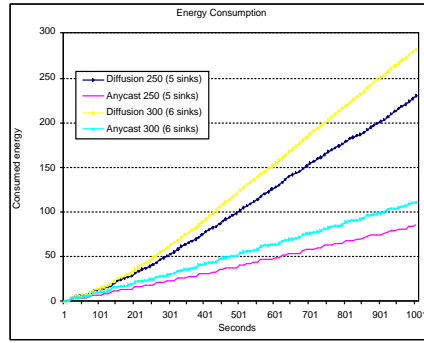


Fig. 3. Energy Consumption of 250/300 nodes in  $790 * 790/860 * 860$  areas (2 percent sinks and 3 sources).

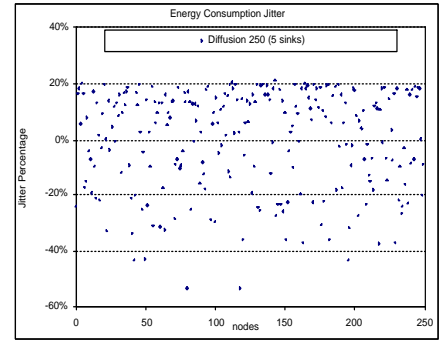


Fig. 4. Energy Consumption Jitter of 250 nodes with Directed Diffusion in a  $790 * 790$  area after 1002 seconds (5 sinks and 3 sources).

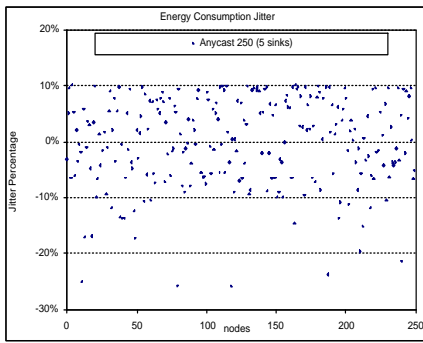


Fig. 5. Energy Consumption Jitter of 250 nodes with anycast in a  $790 * 790$  area after 1002 seconds (5 sinks and 3 sources).

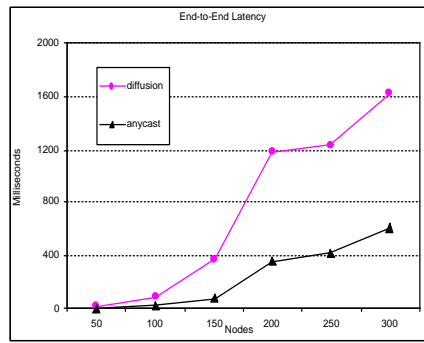


Fig. 6. Mean end-to-end transmission delays.

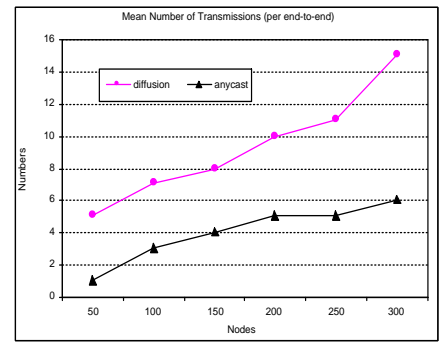


Fig. 7. Mean number of transmissions per end-to-end path (Mean path length).

anycast maintains relatively high data delivery rates for low and moderate sink mobility speeds.

## VI. ACKNOWLEDGEMENTS

We wish to acknowledge Professor Deborah Estrin (UCLA) for suggesting the term micro-server. Nirupama Bulusu is supported by the Australian government through the Center of Excellence program.

## REFERENCES

- [1] The network simulator - ns-2. <http://www.isi.edu/nsnam>.
- [2] C. Carter, S. Yi, P. Ratanachandani, and R. Kravets. Manycast: exploring the space between anycast and multicast in ad hoc networks. In *Proceedings of the 9th MobiCom*, pages 273–285. ACM Press, 2003.
- [3] F. H., J. Cheng, S. Lu, and L. Zhang. A two-tier data dissemination model for large-scale wireless sensor networks. In *Proceedings of the eighth annual international conference on Mobile computing and networking*, pages 148–159. ACM Press, 2002.
- [4] Network Working Group. Host anycasting service. *IETF RFC 1546*, 1993.
- [5] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. In *Proceedings of the ninth ASPLOS-X*, pages 93–104. ACM Press, 2000.
- [6] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Transactions on Networking (TON)*, 11(1):2–16, 2003.
- [7] D. Katabi and J. Wroclawski. A framework for scalable global ip-anycast (gia). In *Proceedings of SIGCOMM*. ACM Press, 2000.
- [8] Q. Li, M. De Rosa, and D. Rus. Distributed algorithms for guiding navigation across a sensor network. In *Proceedings of the 9th Mobicom*, pages 313–325. ACM Press, 2003.
- [9] J. Liu, F. Zhao, and D. Petrovic. Information-directed routing in ad hoc sensor networks. In *Proceedings of the 2nd WSNA*. ACM Press, 2003.
- [10] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: a tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review*, 36(SI):131–146, 2002.