ISSN 2348-375X

Unique Journal of Engineering and Advanced Sciences

Available online: www.ujconline.net

Research Article

THE UNCEASING DETECTION OF NODES AND THE FRAME WORK COORDINATION BETWEEN SENSOR NODES AND ACTOR NODES IN WIRELESS SENSOR NETWORKS

Pratheep S[1*], Kalyana Sundaram P[2], Gnanasekaran T[3]

[1]II-M.E (A.E), Nandha Engineering College-Erode
[2]Professor, Department of ECE, Nandha Engineering College-Erode
[3]Professor, Department of IT, RMK Engineering College-Chennai

Received: 25-02-2014; Revised: 24-03-2014; Accepted: 23-04-2014

*Corresponding Author: **Pratheep S,**
Nandha Engineering College-Erode Email: deepu28@ymail.com

ABSTRACT

A large class of Wireless Sensor Networks (WSANs) applications involves a set of isolated urban areas covered by sensor nodes monitoring environmental parameters. Mobile sinks mounted upon urban vehicles with fixed trajectories (e.g., buses)provide the ideal infrastructure to effectively retrieve sensory data from such isolated Wireless Sensor Networks fields. Existing approaches involve either single-hop transfer of data from sensor node that lie within the mobile sink range or heavy involvement of network periphery nodes in data retrieval, processing, buffering, and delivering tasks. These nodes run the risk of rapid energy exhaustion resulting in loss of network connectivity and decreased network lifetime. The proposed System is minimizing the overall network overhead and energy expenditure associated with the multi hop data retrieval process while ensuring balanced energy consumption among sensor node and prolonged network lifetime. This is achieved through building cluster structures consisted of member nodes that route their measured data to their assigned cluster head. Cluster head perform data filtering upon raw data exploiting potential spatial-temporal data redundancy and forward the filtered information to appropriate end nodes with sufficient residual, located in proximity to the mobile sink trajectory.

Keywords: Eecm Protocols , Adhoc Network Protocol, Routing Protoicols.

INTRODUCTION

A main reason of energy spending in WSNs relates with communicating the sensor readings from the sensor nodes (SNs) to remote sinks. These readings are typically relayed using ad hoc multihop routes in the WSN. A side effect of this approach is that the SNs located close to the sink are heavily used to relay data from all network nodes; hence, their energy is consumed faster, leading to a non uniform depletion of energy in the WSN. This results in network disconnections and limited network lifetime. Network lifetime can be extended if the energy spent in relaying data can be saved. Recent research work has proved the applicability of mobile elements (submarines, cars, mobile robots, etc.) for the retrieval of sensory data from smart dust motes in comparison with multi hop transfers to a centralized element[1].

A mobile sink (MS) moving through the network deployment region can collect data from the static SNs over a single hop radio link when approaching within the radio range of the SNs or with limited hop transfers if the SNs are located further. This avoids long-hop relaying and reduces the energy consumption at SNs near the base station, prolonging the network lifetime. A large class of monitoring applications involves a set of urban areas (e.g., urban parks or building blocks) that need to be monitored with respect to environmental parameters (e.g., temperature, moisture, pollution, light intensity, surveillance, fire detection, etc.)

In these environments, individual monitored areas are typically covered by isolated “sensor islands,” which makes data retrieval rather challenging since mobile nodes cannot move through but only approach the periphery of the network deployment region. In such cases, a number of representative nodes located in the periphery of the sensor field can be used as “rendezvous” points wherein sensory data from neighbor. Nodes may be collected and finally delivered to an mobile sinks MS when the latter approaches with in radio range[2].

In this context, the specification of the appropriate number and locations of rendezvous nodes (RNs) is crucial. The number of RNs should be equivalent (neither small nor very large) to the deployment density of SNs. Herein; we investigate the use of

MSs for efficient data collection from "sensor islands" spread throughout urban environments. We argue that the ideal carriers of such MSs are public surface transportation vehicles (e.g., buses) that repeatedly follow a predefined trajectory with a periodic schedule that may pass along the perimeter of the isolated sensor fields[3].

This is achieved through building cluster structures consisted of member nodes that route their measured data to their assigned cluster head (CH). The CHs perform data filtering upon the raw data exploiting potential spatial-temporal data redundancy and forward the filtered information to their assigned RNs, typically located in proximity to the MS's trajectory. We introduce a sophisticated method for enrolling appropriate nodes as RNs taking into account the deployment pattern and density of sensor nodes. Last, we propose methods for building adaptable inter cluster overlay graphs and techniques for fairly distributing sensory data among RNs and delivering data to MSs in nonintersecting time windows.In the existing system Maximum Amount Shortest Path Tree (MASPT) method is proposed to choose subsinks and relay data from members shown in Figure 1.1. Each member chooses the closest sub sink in terms of hop distance as its destination and then sends its own data or forwards data from downstream nodes to upstream nodes along shortest path trees. However, the number of members associated with each sub sink is independent of its communication time. It is possible that some sub sinks with longer communication time own fewer members, implying that the mobile sink may collect less data than expected. On the other hand, some sub sinks with very short communication time may own too many members. Consequently, the excess data traffic may result in oversaturated sub sinks which are not able to transmit all data to the mobile sink in the limited communication duration. In other words, the MASPT method has low energy efficiency for data collection.We Propose a rendezvous-based data collection approach that explores the controlled mobility of Mobile Sink (MS) and the capability of in-network data caching. Specifically, a subset of static nodes in the network will serve as the rendezvous points (RPs) and aggregate data originated from sources. The MS periodically visits each RP and picks up the cached data[4].

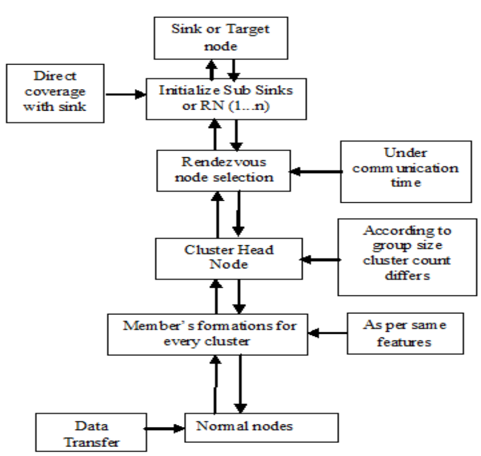

Figure 1: Energy-Efficient Cluster Management Protocol

A. Table-Driven (or Proactive) protocols

The nodes maintain a table of routes to every destination in the network, for this reason they periodically exchange messages. At all times the routes to all destinations are ready to use and as a consequence initial delays before sending data are small. Keeping routes to all destinations up-to-late, even if they are not used, is a disadvantage with regard to the usage of bandwidth and of network resources.

B. On-Demand (or Reactive)

These protocols were designed to overcome the wasted effort in maintaining unused routes. Routing information is acquired only when there is a need for it. The needed routes are calculated on demand. This saves the overhead of maintaining unused routes at each node, but on the other hand the latency for sending data packets will considerably increase[5].

C. Modules

1. Topology formation
2. Cluster head Election
3.Actor node election and Data Process

1. Topology Formation

Deployment of sensor nodes and neighbor node and region estimation in WSN.

2. Cluster head Election

Based on node deployment each node sends the CH_ELECTION Packets to its neighbor for electing the Cluster Head.

3. Actor node Election and Data Process

Actor node election process should be takes place based on nodes which are nearer to the mobile sink. It will act as a intermediary between cluster head and mobile sink

ROUTING PROTOCOLS IN AD HOC NETWORK

Table-Driven (or Proactive)

The nodes maintain a table of routes to every destination in the network, for this reason they periodically exchange messages. At all times the routes to all destinations are ready to use and as a consequence initial delays before sending data are small. Keeping routes to all destinations up-to-date, even if they are not used, is a disadvantage with regard to the usage of bandwidth and of network resources[6].

Proactive

DSDV (Destination-Sequence Distance Vector)

DSDV has one routing table, each entry in the table contains: destination address, number of hops toward destination, next hop address. Routing table contains all the destinations that one node can communicate. When a source A communicates with a destination B, it looks up routing table for the entry which contains *destination* address as B. Next hop address C was taken from that entry. A then sends its packets to C and asks C to forward to B. C and other intermediate nodes will work in a similar way until the packets reach B. DSDV marks each entry by sequence number to distinguish between old and new route for preventing loop[7].

DSDV use two types of packet to transfer routing information: full dump and incremental packet. The first time two DSDV nodes meet, they exchange all of their available routing information in full dump packet. From that time, they only use incremental packets to notice about change in the routing table to reduce the packet size. Every node in DSDV has to send update routing information periodically. When two routes are discovered, route with larger sequence number will be chosen. If two routes have the same sequence number, route with smaller hop count to destination will be chosen.

DSDV has advantages of simple routing table format, simple routing operation and guarantee loop-freedom. The disadvantages are (i) a large overhead caused by periodical update (ii) waste resource for finding all possible routes between each pair.

Reactive

On-demand Routing Protocols

In on-demand trend, routing information is only created to requested destination. Link is also monitored by periodical Hello messages. If a link in the path is broken, the source needs to rediscovery the path. On-demand strategy causes less overhead and easier to scalability. However, there is more delay because the path is not always ready[8].

AODV Routing

Ad hoc on demand distance vector routing (AODV) is the combination of DSDV and DSR. In AODV, each node maintains one routing table.When a node wants to communicate with a destination, it looks up in the routing table. If the destination is found, node transmits data in the same way as in DSDV. If not, it start Route Discovery mechanism: Source node broadcast the Route Request packet to its neighbor nodes, which in turns rebroadcast this request to their neighbor nodes until finding possible way to the destination. When intermediate node receives a RREQ, it updates the route to previous node and checks whether it satisfies the two conditions: there is an available entry which has the same destination with RREQ (ii) its sequence number is greater or equal to sequence number of RREQ. If no, it rebroadcast RREQ .If yes, it generates a RREP message to the source node. When RREP is routed back, node in the reverse path updates their routing table with the added next hop information.If a node receives a RREQ that it has seen before (checked by the sequence number), it discards the RREQ for preventing loop. If source node receives more than one RREP, the one with greater sequence number will be chosen. For two RREPs with the same sequence number, the one will less number of hops to destination will be chosen. When a route is found, it is maintained by Route Maintenance mechanism: Each node periodically send Hello packet to its neighbors for proving its availability. When hello packet is not received from a node in a time, link to that node is considered to be broken. The node which does not receive Hello message will invalidate all of its related routes to the failed node and inform other neighbor using this node by Route Error packet. The source if still want to transmit data to the destination should restart Route Discovery to get a new path. AODV has advantages of decreasing the overhead control messages, low processing, quick adapt to net work topology change, more scalable up to 10000 mobile nodes[9].

DYNAMIC SOURCE ROUTING PROTOCOL

DSR is a reactive routing protocol which is able to manage a MANET without using periodic table-update messages like table-driven routing protocols do. DSR was specifically designed for use in multi-hop wireless ad hoc networks. Ad-hoc protocol allows the network to be completely self-organizing and self-configuring which means that there is no need for an existing network infrastructure or administration. For restricting the bandwidth, the process to find a path is only executed when a path is required by a node (On-Demand-Routing).

In DSR the sender (source, initiator) determines the whole path from the source to the destination node (Source-Routing) and deposits the addresses of the intermediate nodes of the route in the packets. Compared to other reactive routing protocols like ABR or SSA, DSR is beacon-less which means that there are no hello-messages used between the nodes to notify their neighbors about her presence. DSR was developed for MANETs with a small diameter between 5 and 10 hops and the nodes should only move around at a moderate speed.DSR is based on the Link-State-Algorithms which mean that each node is capable to save the best way to a destination. Also if a change appears in the network topology, then the whole network will get this information by flooding.

DSR contains 2 phases

- Route Discovery (find a path)
- Route Maintenance (maintien a path)

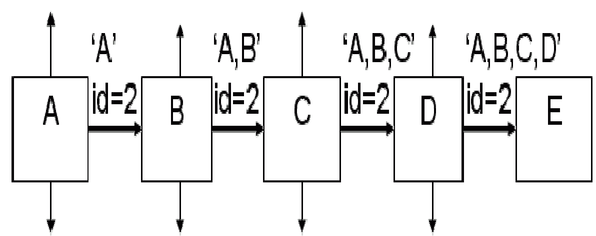

Figure 2: Route Discovery

If node A has in his Route Cache a route to the destination E, this route is immediately used. If not, the Route Discovery protocol is started:

1. Node A (initiator) sends a Route Request packet by flooding the network
2. If node B has recently seen another Route Request from the same target or if the address of node B is already listed in the Route Record, Then node B discards the request!
3. If node B is the target of the Route Discovery Figure3.2, it returns a Route Reply to the initiator. The Route Reply contains a list of the "best" path from the initiator to the target,
4. When the initiator receives this Route Reply, it caches this route in its Route Cache for use in sending subsequent packets to this destination.
5. Otherwise node B isn't the target and it forwards the Route Request to his neighbors (except to the initiator).

   Path-finding-process: Route Request & Route Reply

Route Maintenance

In DSR every node is responsible for confirming that the next hop in the Source Route receives the packet. Also each packet is only forwarded once by a node (hop-by-hop routing)IS is shown in Figure3.3,If a packet can't be received by a node, it is retransmitted up to some maximum number of times until a confirmation is received from the next hop.

Route Maintenance:
Once a broken link is discovered, nodes make a new reference height and broadcast to their neighbors. All nodes in the link will change their reference height and Route Creation is done to reflect the change.

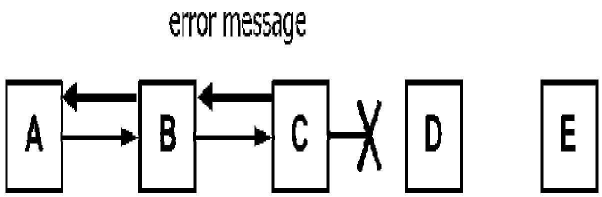

Figure 3: Route Maintenance

Only if retransmission results then in a failure, a Route Error message is sent to the initiator that can remove that Source Route from its Route Cache. So the initiator can check his Route Cache for another route to the target. If there is no route in the cache is a Route Request packet is broadcasted.

1. If node C does not receive an acknowledgement from node D after some number of requests, it returns a Route Error to the initiator A.
2. As soon as node receives the Route Error message, it deletes the broken-link-route from its cache. If A has another route to E, it sends the packet immediately using this new route.
3. Otherwise the initiator A is starting the Route Discovery process again.

TORA (Temporary Ordered Routing Algorithm)

TORA is based on link reversal algorithm. Each node in TORA maintains a table with the distance and status of all the available links. Detail information can be seen at Figure 3.3.
TORA has three mechanisms for routing:
Route Creation:
TORA uses the "height" concept for discovering multiple routes to a destination. Communication in TORA network is downstream, from higher to lower node. When source node does not have a route to destination, it starts Route Creation by broadcasting the Query messages (QRY). QRY is continuing broadcasted until reaching the destination or intermediate node that have the route to the destination. The reached node then broadcast Update (UPD) message which includes its height. This mechanism is called reversal algorithm and is claimed to create number of direct links from the originator to the destination
Route Erasure:
Erases the invalid routes by flooding the "clear packet" through the network The advantages of TORA are: having multiple paths to destination decreases the route creation in link broken case therefore decrease overhead and delay to the network. TORA is also claimed to be effective on large and mildly congested network [9]. The drawbacks are requiring node synchronization due to "height" metric and potential for oscillation. Besides that, TORA may not guarantee to find all the routes for reserving in some cases

A Rendezvous-Based Approach Enabling Energy-Efficient Sensory Data COLLECTION WITH MOBILE SINKS

MSs are mounted upon public buses circulating within urban environments on fixed trajectories and near-periodic schedule. Namely, sinks motion is not controllable and their routes do not adapt upon specific WSN deployments. Our only assumption is that sensors are deployed in urban areas in proximity to public transportation vehicle routes. Also, an adequate number of nodes are enrolled as RNs as a fair compromise between a small number Which results in their rapid energy depletion and a large number which results in reduced data throughput.
SNs are grouped in separate clusters. Raw sensory data are filtered within individual clusters exploiting their inherent spatial-temporal redundancy. Thus, the overhead of multihop data relaying (interclustering traffic) to the edge RNs is minimized. Given that the communication cost is several orders of magnitude higher than the computation cost, in-cluster data aggregation can achieve significant energy savings. A basic assumption in the design of Mobi Cluster protocol is that SNs are location unaware, i.e., not equipped with GPS- capable antennae. Also, we assume that each node has a fixed number of transmission power levels. we assume the unit disk model, which is the most common assumption in sensor network literature. The underlying assumption in this model is that nodes which are closer than a certain distance (transmission range R) can always communicate. However, in practice, a message sent by a node is received by the receiver with only certain probability even if the distance of the two nodes is smaller than the transmission range.
The five phases of MobiCluster are described. The first three phases comprise the setup phase while the last two comprise the steady phase. The setup phase completes in a single MS trip and during this trip, the MS periodically broadcasts BEACON messages which are used by SNs for determining a number of parameters important for the protocol operation. In the steady phase, data from SNs are routinely gathered to Rns and then sent to MS. During the steady phase, reselection of RNs and/or local reclustering is performed in case of energy exhaustion of some critical nodes.
Phase 1: Clustering
The large-scale deployment of WSNs and the need for data aggregation necessitate efficient organization of the network topology for the purpose of balancing the load and prolonging the network lifetime. Clustering has proven to be an effective approach for organizing the network in the above context. Besides achieving energy efficiency, clustering also reduces channel contention and packet collisions, resulting in improved network throughput under high load. We slightly modify the approach of to build clusters of two different sizes depending on the distance of the CHs from the MS's trajectory. Specifically, SNs located near the MS trajectory are grouped in small- sized clusters while SNs located farther away are grouped in clusters of larger size. By maintaining the clusters of these CHs small, CHs near the MS trajectory are relatively relieved from intracluster processing and communication tasks and thus they can afford to spend more energy for relaying intercluster traffic to RNs.

Phase 2: RNs Selection

RNs guarantee connectivity of sensor islands with MSs hence, their selection largely determines network life time .RNs lie within the range of traveling sinks and their location depends on the position of the CH and the sensor field with respect to the sinks trajectory. Suitable RNs are those that remain within the MS's range for relatively long time, in relatively short distance from the sink's trajectory and have sufficient energy supplies. In practical deployments, the number of designated RNs introduces an interesting trade-off:

Phase 3: CHs Attachment to RNs

CHs located far from the MS trajectories do not have any RNs within transmission range. An important condition for building intercluster overlay graphs is that Chs with no attached RNs, attach themselves to a CH u with nonempty Ru set so as to address their clusters' data to u.The description of the intercluster overlay graph building procedure can be found in Appendix C, available in the online supplemental material. It is noted that our approach typically requires a single MS trip to collect (through the receipt of BEACON messages) the information needed to execute the setup phase. Clustering starts upon the completion of the first MS trip. All these phases complete in reasonably short period of time, typically within the time interval between two successive bus trips. As soon as the setup phase finalizes, sensory data collected at CHs from their attached cluster members are forwarded toward the RNs following an intercluster overlay graph. The selected transmission range among CHs may vary to ensure a certain degree of connectivity and to control interference[10].

Phase 4: Data Aggregation and Forwarding to the RNs

The steady phase of Mobi Cluster protocol starts with the periodic recording of environmental data from sensor nodes with a Tr period. The data accumulated at individual source nodes are sent to local CHs (intracluster communication) with a Tc period (typically, Tc is a multiple of Tr). CHs perform data processing to remove spatial-temporal data redundancy, which is likely to exist since cluster members are located maximum two hops away. CHs then forward filtered data toward remote CH they are attached to. Alongside the intercluster path, a second-level of data filtering may apply.

Phase 5: Communication between RNs and Mobile Sinks

The delivery of data buffered to RNs to MSs. Data delivery occurs along an intermittently available link; hence, a key requirement is to determine when the connectivity between an RN and the MS is available. Communication should start when the connection is available and stop when the connection no longer exists, so that the RN does not continue to transmit data when the MS is no longer receiving it. To address this issue, we use an acknowledgment-based protocol between RNs and MSs. The MS, in all subsequent path traversals after the setup phase, periodically broadcasts a POLL packet, announcing its presence and soliciting data as it proceeds along the path. The POLL is transmitted at fixed intervals Tpoll (typically equal to Tbeacon). This POLL packet is used by RNs to detect when the MS is within connectivity range. The RN receiving the POLL will start transmitting data packets to the MS. The MS acknowledges each received data packet to the RN so that the RN realizes that the connection is active and the data were reliably delivered. The acknowledged data packet can then be cleared from the RN's cache.

TOPOLOGY FORMATION

Deployment of sensor nodes and neighbor node and region estimation in WSN. Sensor nodes are grouped in separate clusters A basic assumption in the design of MobiCluster protocol is that SNs are location unaware as shown in the Figure (a).

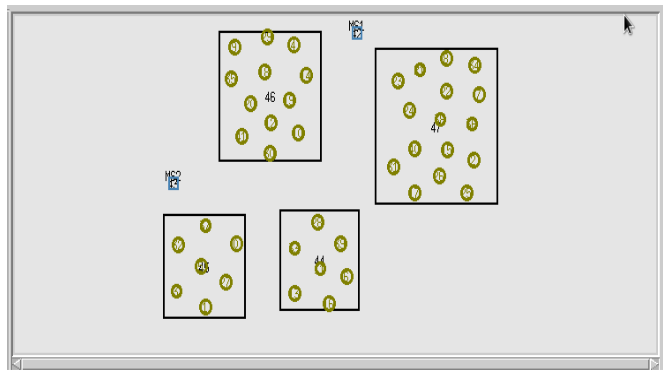

Figure (a)

Based on node deployment each node sends the CH_ELECTION Packets to it's neighbor for electing the Cluster Head, Actor node election process should be takes place based on nodes which are nearer to the mobile sink. It will act as a intermediary between cluster head and mobile sink as shown in the Figure (b).

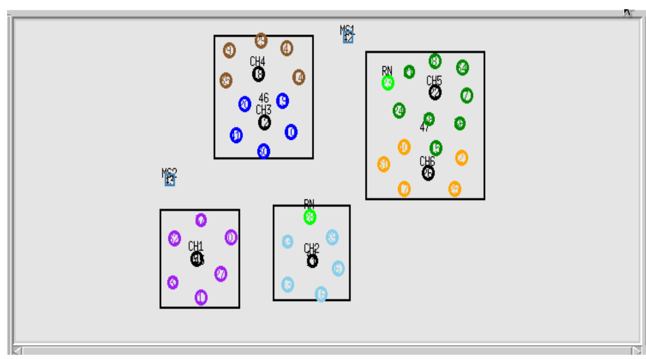

Figure (b)

It act as automated mode according to environmental condition and situation as shown in Figure (c).

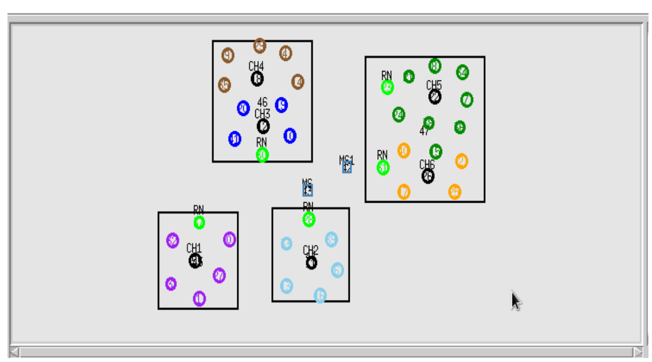

Figure (c)

COMPARISON OF POWER EFFICIENCY

The Figure.4.shows the power efficiency between existing method and proposed method

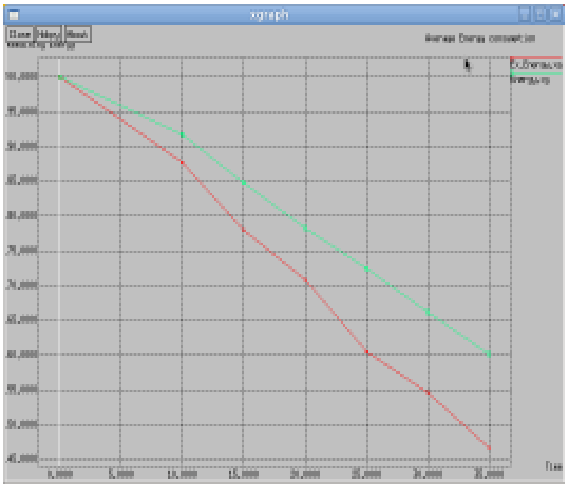

Figure .4

LESSONS LEARNED AND FUTURE WORKS

In this section, we summarize the lessons learned from the works we surveyed in this paper and provide a comparison on some important attributes of the reviewed protocols. We also suggest feasible future works and research directions The review of the impacts of other planes such as power management on coordination is recommended as a future work. Also some direct-contact data collection approaches can be modified to be considered as rendezvous based approaches

CONCLUSION

Given the importance of weak connectivity problem in wireless sensor actor networks (WSANs) to critical applications of WSANs, and also the lack of any classifications on the problem in WSANs literature, we proposed a categorization of connectivity in WSANs by focusing on weak connectivity and its impact on coordination. The categorization has been derived from existing researches in the area including our own previous researches on connectivity, coverage, and coordination in WSANs.The connectivity objective is addressed by employing MSs to collect data from isolated urban sensor islands and also through prolonging the lifetime of selected peripheral RNs which lie within the range of passing MSs and used to cache and deliver sensory data derived from remote source nodes. Increased data throughput is ensured by regulating the number of RNs for allowing sufficient time to deliver their buffered data and preventing data losses and enables balanced energy consumption
across the WSN through building cluster structures that exploit thigh redundancy of data collected from neighbor nodes and minimize intercluster data overhead.

REFERENCES

1. Akyildiz IF, Melodia T, Pompili D, Gungor. A distributed coordination framework for wireless sensor and actor networks in 6th ACM International Symposium on Mobile Ad hoc Networking andComputing. 2005.
2. Akyildiz IF, Kasimoglu IH. Wireless sensor and actor networks: research challenges, Ad Hoc Networks. 2004; 2(4): 351.
3. Ahmed N, Kanhere S, Jha S. The holes problem in wireless sensor networks: a survey, ACM SIGMOBILE Mobile Computing and Communications Review. 2005; 9(2): 18.
4. BarolliL, Durresi A, Paruchuri V. Delay-energy aware routing protocol for sensor and actor networks," in 11th International Conferenceon Parallel and Distributed Systems. 2005; 292.
5. Bri D, Garcia M, Coll H, Loret J. Using MANET protocols in wireless sensor and actor networks" in Second International Conferenceon Sensor Technologies and Applications. 2008.
6. Eltoweissy M, Rezgui A. Service-oriented sensor-actuator networks, IEEE Communication. Mag. 2007; 45: 92.
7. Ghosal D, Yick J, Mukherjee B. Wireless sensor network survey, Computer Networks. 2008; 52(12): 2292.
8. Nayak A, Stojmenovic I. Wireless Sensor and Actuator Networks, Algorithms and Protocols for Scalable Coordination and Data Communication. Wiley Inter science. 2010.
9. Ruiz IE, Villasenor GL. Cooperation mechanism taxonomy for wireless sensor and actor networks, Ad Hoc & Sensor Wireless Networks. 2009; 7: 91.
10. Sivakumar R, Vedantham R, Zhuang Z. Hazard avoidance in wireless sensor and actor networks, Computer Communications. 2006; 29: 2447.

Source of support: Nil, Conflict of interest: None Declared