Data Collection Using Smart Nodes For Mobile Monitoring Applications.

Bhagyashri Patil
Computer Engineering Department
St. Vincent Pallotti College of Engineering and Technology
Nagpur
bhagyashriptl198@gmail.com

Abhishek Pathak
Computer Engineering Department
St. Vincent Pallotti College of Engineering and Technology
Nagpur
abhishekpathak81@rediffmail.com

Abstract: The use of mobile ferries from which we understand the nodes that transfer the data between other nodes in process of their movement, provides multiple benefits for Wireless Sensor Networks (WSNs). In the paper, we suggest and evaluate several different approaches for collecting the data using ferries from WSN networks for different scenarios. We present the results of the simulations that reveal the effects of the WSN density, ferry speed, node sleep policy and maximum hop limitation on the performance and required resources for the tested protocols.

In this paper we survey the recent progress of using mobile ferry nodes for data collection in WSNs addressing two main areas, determining the path of the ferry and the scheduling when to dispatch the ferry to collect data from sensors. We also highlight challenges facing the deployment of mobile ferries in wireless sensor networks.

Keywords: ***wireless sensor networks; mobility; WSN; mobile ferry; delay tolerance networks; routing protocols; energy efficiency protocols***

Introduction:

The wireless sensor networks (WSN) are a division of adhoc wireless networks predictable among a number of sensor nodes deployed over a monitored area.[16]h sensor node is a low-cost, energy-constrained device capable of sensing its environment and performs simple processing tasks and then transmits sensed data over the wireless medium towards neighbouring sensor nodes. To perform more complicated data processing, data gathering mechanisms are designed and deployed for efficient data collection at one or a small number of reliably powered sink nodes inside the WSN.

Sink nodes are dedicated nodes that are responsible for gathering composed data and serve as gateway between the sensor network and the wired or wireless network.[3] While the applications of sensors may be quite different, data packets want to be aggregated at data sink. In a homogeneous network where sensors are organized into a flat topology, since they need to relay many packets from sensors far away from the data collector. As a result, If any of these sensors fail, then other sensors also cannot reach the data collector and the whole network becomes disconnected, but most of the nodes can still survive for a long period. For a large-scale sensor network, using single static data sink to gather data from all sensors is not a good idea. For some applications, sensors are densely deployed and connected, but some of the sensors may be disconnected and cannot forward data to the data sink via wireless links.

In this paper Ferry nodes acts as smart nodes. Here, the ferries have the capacity to collect the data from the nodes and transfer that collected data to the base- station/server.

A mobile data collector is perfectly suitable for such applications. A mobile data collector serves as a mobile “data carrier” and links all are separated to sub networks. The moving path of the mobile data collector acts as virtual links between separated sub networks.

To provide a scalable data-gathering scheme for large-scale static sensor networks, we employ mobile data collectors to gather data from sensors. Mobile data collector could be a mobile robot or a vehicle equipped with a powerful transceiver, battery, and large memory. The mobile data collector starts a tour from the data sink, traverses the network, collects sensing data from nearby nodes while moving, and then returns and uploads data to the data sink. Since the data collector is mobile, it can move close to sensor nodes, such that if the moving path is well planned, the network lifetime can be greatly prolonged.

Here, network lifetime is defined as the duration from the time sensors start sending data to the data sink to the time when a certain percentage of sensors either run out of battery or cannot send data to the data sink due to the failure of relaying nodes. In the following, for convenience, we use mobile collector to denote the mobile data collector.

The mobile ferries (MFs) are the nodes that transfer the data between the isolated parts of the WSN in the process of their movement with temporal data storage on-ferry .[11] The most significant difference between the mobile sinks or mobile relays and the MFs is that the MF mobility does not need to be controlled, while the relays and sinks usually require full control over their mobility. The other advantage of MFs is that those can be used to interconnect even distant subnetworks – see Fig. 1. As has been shown e.g. in , the MFs can increase the coverage of the WSN by connecting the sink and isolated subnetworks and improve the energy efficiency.

The major disadvantage of the data transferring using MFs is the communication latency that depends on the network layout, speed and mobility pattern of the MFs.

In the paper, we focus on the MFs scenario and the data collecting from the isolated nodes and subnetworks using a MF. Below we suggest and evaluate several different data collection (DCol) protocols, under which we mean the combination of isolated cluster discovery, connection establishment, routing, and data communication mechanisms.

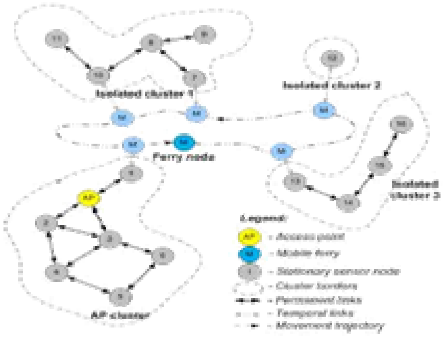

Fig 1: Example of the WSN with mobile ferries**.**

Literature Survey

In [1], the cluster heads will inevitably consume more energy than other sensor nodes. To avoid the Problem of cluster heads failing faster than other nodes, sensor nodes can become cluster heads rotationally. In [2], controlled movement was exploited to improve data delivery performance. Some mobile observers, called message ferries, were used to collect data from sensors. Two variants were studied based on whether ferries or nodes initiate proactive movement.

In [4], a number of mobile observers, called data mules, pick up data directly from the sensors when they are in close range, buffer the data, and drop off the data to wired access points. The movement of mules is modelled as 2-D random walk. In [5], mobile observers traverse the sensing field along parallel straight lines and gather data from sensors. To reduce latency, packets sent by some sensors are allowed to be relayed by other sensors to reach mobile observers. This scheme works well in a large-scale uniformly distributed sensor network. However, in practice, data mules may not always be able to move along straight lines, for example, obstacles or boundaries may block the moving paths of data mules.

When only a small number of data mules are available and not all sensors are connected, data mules may not cover all the sensors in the network if they only move along straight lines In [6], a data-gathering scheme was proposed to minimize the maximum average load of a sensor by jointly considering the problems of movement planning and routing. Based on the assumption that sensors are distributed according to a Poisson process, the average load of a sensor can be estimated as a function of the node density.

In [7], several advantages and design issues were discussed for incorporating controlled mobility into the networking infrastructure, and the main focus was on motion/speed control and communication protocol design. In [8], a heterogeneous and hierarchical architecture was proposed for the deployment of WSNs with mobile sinks for large-scale monitoring, where the sensors transmit their sensing data to the gateway nodes for temporary storage through multihop relays and the mobile sinks travel along predetermined trajectories to collect data from nearby gateway nodes. Under this data-gathering paradigm, the capacitated minimum forest problem was studied, and approximate algorithms were devised for instances where all gateways have uniform and arbitrary capacities, respectively.

The operation of the MFs typically includes three major phases:

• Isolated node or isolated clusters detection and connection establishment;

• Data download from the isolated nodes or clusters to the memory of the MF;

• Sink cluster detection, connection to it and data uploading from the MF memory to the sink.

In the case if the movement pattern of the MF is known in advance by all the nodes and the locations of all the nodes are known to the MF, the cluster detection and the connection establishment can be implemented effectively

with the minimum energy consumption (consider e.g., [3], [16]). Nonetheless, if the MF movement schedule is unknown or if the route is subject to change, the nodes are unable to predict the time when the MF will arrive. In such scenario two following techniques can be used. The first option is to use the mechanism that wakes up the sensor node once the MF has arrived, e.g., the radio triggered activation [3] or activation from the sensor signal (e.g., button press or sensor data analysis). This solution ensures minimum energy consumption, but increases the price of the WSN node due to the external components use.

The other option, which does not need any external components but causes higher energy consumption is the periodic listening by the WSN nodes for the advertisements issued by the MF during its movement [3], [12], [17].

Proposed methodologies:

For investigating the different WSN Dcol methodologies using the MF we have chosen the most general scenario. We assume that n static wireless sensor nodes are randomly placed in the area of (x, y) meters and their position is unknown. Each node has N data packets that should be forwarded to the sink node. At random moment of time, single MF approaches the test area and crosses it via some unpredictable route. The mobility of the MF cannot be controlled by the WSN in any way. For increasing the operation time of the WSN, the sensor nodes are allowed to use the low-power sleep mode. The sensors are not synchronized and their sleep schedules are independent.

Inputs:

Addr(host):- Address of host node.(i.e next node on the route to ferry. Addr(token holder):- Address of the node that holds the token.
ID(last beacon)- ID for the last received beacon packet.
ID(last rqst)-ID for the last received token request packet.
ID(last token)-ID for the last received token advertise packet.
N_{buf}- Number of packets in buffer of Sensors.
N(ferry)- Distance from ferry(Number of hops). NodeType-
Type of the node,can be either Ferry or Sensor.
T(beacon)- period for beacons(or token advetisement) transmitted by ferry.
T(dataTX)- Period of data transmission by
Sensors. T(rand)- Random Delay.
Types of Radio Packets:-
ACKP- Acknowledgement
Packet. BP- Beacon Packet.
DP- Data Packet.
Tadv- Token advertise packet.
Trqst-token request packet.
Tgrant- Token grant packet.
Treturn- Token return packet.
Radio Packet Arguments:-
Data- Actual data.
ID- unique identification of this packet.
ID(ack)- unique identification of the packet to be acknowledged. N(hops)- number of hops the packet had already passed. N(left)- number of hops left for this packet.

Figure 1:- Input data and radio packet format for the developed algorithms.

If nodeType is Ferry
broadcast BP(IDack) every T(beacon) seconds
if DP (ID Data) received
ID -> ID(ack)
end if
else if nodeType is Sensor
if BP(IDack) received
delete packet with ID=ID(ack) from buffer if it exists
send next data packet from the buffer after random(0,Trand) seconds end if
end if

Figure 2:- Algorithm for data collection from single hop(SH) neighbours.

If nodeType is Ferry
broadcast BP(IDack ID) every Tbeacon seconds if DP(ID Data) received
ID-> Idack
end if
else if nodeType is Sensor
if BP (Idack ID) received and ID != ID(last beacon) ID=ID(last beacon)
address of BP(IDack) transmitter -> Addr(host) if packet with ID=IDack exists in buffer
if packet with ID was generated not by me
send ACKP(IDack) to the node from which it has been received end if
delete packet with Idack from buffer Nbuf-1 -> Nbuf
end if
if Nbuf>0
send next data packet from the buffer to Addr(host) after random(0,Trand) seconds
else
rebroadcast BP(IDack,ID) after random(0,Trand) seconds
end if
else if DP(ID,Data) received
put DP(ID,Data) packet in my data buffer Nbuf+1 -> Nbuf
else if ACKP (Idack)received
if packet with ID=IDack exists in buffer if packet with ID was not generated by me
send ACKP(IDack) to the node from which it has been received end if
delete packet Idack from buffer Nbuf-1 -> Nbuf
end if
end if
end if

Figure 3: Algorithm for limited multihop single route(MSR) data collection.

If nodeType is Ferry
broadcast BP(ID,Nhops=0) every T(beacon) seconds
If DP(ID,Data received)
ID -> ID(ack)
send ACKP (Idack) to source of DP(ID,Data) end if
else if nodeType is Sensor
if BP(ID,Nhops) received and ID != ID(last beacon) ID -> ID(last beacon)
N(hops) -> N(ferry)
rebroadcast BP (ID,Nferry+1) after random (0,Trand) seconds
start broadcasting data packets DP(ID,Data,Nleft) from buffer every T(DataTX) seconds
else if DP (ID,Data,Nleft) received and Nferry < Nleft ID -> ID(ack)
put DP (ID,Data) packet in my data buffer
send ACKP (ID(ack)) to source of DP(ID,Data,Nleft) else if ACKP(IDack) received
delete packet Idack from buffer end if
end if

Figure 4: Algorithm for data collection using unlimited multihop multiroute flooding (MMF) towards Ferry.

For such scenario, the main task for the data collecting protocol is the reliable (i.e., without packet losses) data transfer from the sensor nodes to the MF. Unfortunately, the majority of the existing protocols cannot be used for this scenario due to a high level of input data uncertainty.

Therefore, below we suggest the five protocols for collecting the data from such environment. The algorithms describing the protocols are presented in Figs. 4-8 with Fig. 3 revealing the designation used by all algorithms. The first algorithm (see Fig. 4) implements the most basic data collecting protocol for the MF scenario and is similar to the one used e.g., in [17], [18]. This protocol implies the periodic broadcasting of the beacon packet by the MF to advertise its presence to the sensor nodes. The WSN nodes within the communication range of the MF that receive this advertisement, reply with the data packet. To improve the reliability of the communication, we have included in the beacon packet the IDack field that signalizes

if nodeType is Ferry
if Addr(token holder)= myself
broadcast Tadv(ID) every T(beacon) seconds
end if
if Trqst(ID) is received and Addr(token holder)=myself
initial source of Trqst(ID) -> Addr(token holder)
send Tgrant() to the source of Trqst (ID) and start “connection lost” timer
else if DP(ID,Data) received
rebroadcast the message (used instead of ACK) and restart “connection lost”
timer end if
if “connection lost” timer fired or Treturn() is received and stop “connection lost” timer
myself -> Addr(token holder)
end if
else if nodeType is Sensor
if Tadv(ID) is received and ID!= ID(last beacon) ferry -> Addr(token holder)
if Nbuf >0
address of Tadv(ID) transmitter -> Addr(host)
send Trqst (ID) to Addr(host) after random(0, Trand) seconds else
rebroadcast Tadv(ID) after random (0,Trand) seconds
end if
else if Trqst(ID) is received and ID!= ID(last request)
ID -> ID(last request)
forward Trqst(ID) to Addr(host)
else if Tgrant(Addr(token holder)) is received
address of Tgrant (Addr(token holder)) transmitter -> Addr(host) if Addr(token holder) = myself

start sending DP(ID, Data) to Addr(host) every T(dataTX) seconds else
rebroadcast Tgrant(Addr(token holder)) after random(0,Trand) seconds
end if
else if Treturn() is received
forward Treturn to Addr(host)
else if DP(ID,Data) received if
DP(ID,Data) is for me
retransmit DP(ID,Data) to Addr(host)
else if DP(ID, Data) is from Addr(host)
delete packet ID from buffer
Nbuf-1 ->
Nbuf if Nbuf=0
send Treturn to Addr(host)
end if
end
if end if
end if

Figure 5: Algorithm for limited token based data leaching.(TDL)

if nodeType is Ferry
broadcast BP(Idbeacon) every Tbeacon seconds
if DP (ID,Data) received
ID -> ID(ack)
send ACKP (Idack) to source of
DP(ID,Data) end if
else if nodeType is Sensor
If BP(ID) received and ID!= ID(last beacon)
address of BP(ID) transmitter -> Addr(host)
ID -> ID(last beacon)
move one data packet from buffer to buffer
2 if buffer 2 is not empty
send data packet from buffer 2 to Addr(host) after T(dataTX) seconds and start "packet lost" time
else if ACKP(ID(ack)) received
delete packet ID(ack) from buffer
2 if buffer 2 is not empty
send data packet from buffer 2 to Addr(host) after t(dataTX) seconds and start "packet lost" time
end if
else if DP(ID, Data)
received ID -> ID(ack)
put DP(ID, Data) packet in my buffer 2
send ACKP(ID(ack)) to source of DP(ID, Data)
send data packet from buffer 2 to Addr(host) after T(dataTX) seconds and start "packet lost" timer
end if
if "packet lost" time had fired
send data packet from buffer to Addr(host) after T(dataTX) seconds and restart "packet lost" timer
end if
end if

Figure 6: Algorithm for collecting a single data packet from all the nodes in the network (SPC)

Algorithm for collecting a single data packet from all the nodes in the network(SPC) [22]the sensor node that its previous data packet has been successfully received by MF.

Unlike the first algorithm, the second and the following enable the use of multi-hop forwarding. E.g., for the algorithm presented on Fig. 5 the WSN nodes once finished transmitting all data packets, start to rebroadcast the beacon advertisements and forward the data traffic from the other nodes to the MF. The data from the sensor node to the MF are transmitted via the route with the minimum transmission time for the beacon packet from the MF to this sensor. In case of data or acknowledgement packets collision – the data is rebroadcasted once again after the next beacon.

The algorithm presented in Fig. 6 uses the multi-hop multi-route flooding technique that has been suggested for partially-connected WSNs. Once receiving the beacon, the sensor nodes rebroadcast it and start sending data packets.

The neighbor nodes that are located closer to the MF (i.e., having less hops) acknowledge, save and forward the packets from the more distant nodes.

The algorithm presented in Fig. 7 utilizes the token mechanism. The MF advertises the token, for which the sensor nodes compete. The winner – transmits all its packets to the MF with short delay. In case if the node has no data to transmit, it retransmits the token advertisement further enabling the other nodes to enter the competition. Currently, only one node can hold the token at a time.

The "connection lost" timer on the MF ensures network recovery in the case if the token has been lost or if data connections have been broken.

Finally, the Single-Packet Collect (SPC) algorithm presented in Fig. 8 tries to collect only a single packet but from each node in the WSN. This protocol is especially useful for the applications that require the "snapshot" of the measurements in the whole WSN and do not care much about the measurements' history for specific nodes. To the best of our knowledge, this is the first protocol for WSNs that tries to equalize the DCol from different sensor nodes.

For distinguishing the traffic from the other nodes that should be transmitted as soon as possible from node's own packets, the second buffer (bufferII) has been introduced.

Conclusion

In this paper we show how using mobile ferries can be used for data gathering in WSNs addressing two areas, determining the path of the ferry and the scheduling when to dispatch the ferry to collect data from static sensors. We presented a classification of mobile ferries, based on the role they play in addition to carrying information. Furthermore, we will show on path planning and scheduling of ferry dispatching in the literature. In addition to that, common challenges in deploying mobile ferries in WSNs were discussed along with many of their possible applications.

REFERENCES:

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energyefficient communication protocols for wireless microsensor networks," in Proc. HICSS, Maui, HI, Jan. 2000, pp. 1–10.

[2] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in Proc. ACM MobiHoc, 2004, pp. 187–198.

[3] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: Modelling a three-tier architecture for sparse sensor networks," in Proc. IEEE WorkshopSens. Netw. Protocols Appl., 2003, pp. 30–41.[4] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting mobility for energy efficient data collection in wireless sensor networks". Norwell, MA: Kluwer, 2005.

[5] D. Jea, A. A. Somasundara, and M. B. Srivastava, "Multiple controlled mobile elements (data mules) for data collection in sensor networks," inProc. IEEE/ACM Int. Conf. DCOSS, Jun. 2005, pp. 244–257.

[6] J. Luo and J.-P. Hubaux, "Joint mobility and routing for lifetime elongation in wireless sensor networks," in Proc. IEEE INFOCOM, 2005, pp. 1735–1746.

[7] A. Kansal, A. Somasundara, D. Jea, M. Srivastava, and D. Estrin, "Intelligent fluid infrastructure for embedded networks," in Proc. ACM MobiSys, 2004, pp. 111–124.

[8] W. Liang, P. Schweitzer, and Z. Xu, "Approximation algorithms for capacitated minimum forest problems in wireless sensor networks with a mobile sink," IEEE Trans. Comput., to be published.

[9] F. Bai, K. S. Munasinghe, and A. Jamalipour, "A novel information acquisition technique for mobile-assisted wireless sensor networks," IEEE Trans. Veh. Technol., vol. 61, no. 4, pp. 1752–1761, May 2012.

[10] I. Vasilescu, K. Kotay, D. Rus, M. Dunbabin, and P. Corke, "Data collection, storage and retrieval with an underwater sensor network," in Proc.ACM SenSys, 2005, pp. 154– 165.

[11] A. A. Somasundara, A. Ramamoorthy, and M. B. Srivastava, "Mobile element scheduling for efficient data collection in wireless sensor networks with dynamic deadlines," in Proc. IEEE RTSS, Dec. 2004, pp. 296–305.

[12] Chongqing Zhang , Binguo wang , Shen Fang , Zhe Li , " Clustering Algorithms for wireless sensor networks using spatial data correlation ", International conference on information and Automation , pp-53-58 ,june 2008.

[13] Zhikui chen , Song Yang , Liang Li and Zhijiang Xie , " A clustering Approximation Mechinism based on Data Spatial Correlation in Wireless sensor Networks ", Proceedings of the 9th international conferenses on wireless telecommunication symposium -2010.

[14] S. Basagni, A. Carosi, and C. Petrioli, "Controlled vs. Uncontrolled mobility in wireless sensor networks: some performance insights," Proc. IEEE Vehicular Technology Conference (VETECF'07 Fall), Mar. 2007, pp. 269–273, doi:10.1109/VETECF.2007.70.

[15] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs: modeling a three-tier architecture for sparse sensor networks," Proc. IEEE Workshop on Sensor Network Protocols and Applications (SNPA'03), IEEE Press, May 2003, pp. 30–41, doi:10.1109/SNPA.2003.1203354.

[16] T. Chen, T. Chen, and P. Wu, "Data collection in wireless sensor networks assisted by mobile collector," Proc. IFIP Wireless Days (WD'08), IEEE Press, Nov. 2008, pp. 1–5, doi:10.1109/WD.2008.4812895.

[17] M. Zhao and Y. Yang, "Data gathering in wireless sensor networks with multiple mobile collectors and SDMA technique sensor networks," Proc. IEEE Wireless Communications and Networking Conference (WCNC'10), IEEE Press, Apr. 2010, pp. 1–6, doi:10.1109/WCNC.2010.5506199.

[18] H. Jun, W. Zhao, M. H. Ammar, E. W. Zegura, and C. Lee, "Trading latency for energy in densely deployed wireless ad hoc networks using message ferrying," Ad Hoc Netw., vol. 5, no. 4, May 2007, pp. 444–461, doi:10.1016/j.adhoc.2006.02.001.

[19] G. Anastasi, M. Conti, E. Gregori, C. Spagoni, and G. Valente, "Motes sensor networks in dynamic scenarios: an experimental study for pervasive applications in urban environments," International Journal of Ubiquitous Computing and Intelligence, vol. 1, no. 1, Apr. 2007, pp. 9–16, doi:10.1166/juci.2007.002.

[20] A. Kansal, A. A. Somasundara, D. D. Jea, M. B. Srivastava, and D. Estrin, "Intelligent fluid infrastructure for embedded networks," Proc. International Conference on Mobile Systems, Applications, and Services (MobiSys '04), ACM, 2004, pp. 111–124,doi:10.1145/990064.990080.

[21] S. Gao, H. Zhang, and S. K. Das, "Efficient data collection in wireless sensor networks with path-constrained mobile sinks," IEEE Transactions on Mobile Computing, vol. 10, no. 4, Apr. 2011, pp. 592 –608, doi:10.1109/TMC.2010.193.

[22] A. A. Somasundara, A. Kansal, D. D. Jea, D. Estrin, and M. B. Srivastava, "Controllably mobile infrastructure for low energy embedded networks," IEEE Trans. Mobile Computing, vol. 5, no. 8, Aug. 2006, pp. 958–973, doi:10.1109/TMC.2006.109.