International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 8, October - 2012

Context Monitoring Framework for Sensor Nodes

1Avinash Palave,2 Prof. Dr.Shiv K Sahu,3Prof.Amit Sinhal

1Student, M.Tech.I.T. Technocrats Institute Of Technology
Bhopal,M.P.India

2 Professor.HOD I.T., Technocrats Institute Of Technology
Bhopal,M.P.India

3Professor,CSE Dept, Technocrats Institute Of Technology
Bhopal,M.P.India

***Abstract*— This paper can perform efficient processing and sensor control mechanisms by using context monitoring approach. Here implemented ESS algorithm for calculating ESS. Also implemented Greed MCFSS algorithm to reduce the energy cost as much as possible while simplifying the computation.This paper describes energy utilization by mobile sensor nodes.**

By developing this paper can solve energy consumption problem. Here used performance metrics such as packet delivery ratio, end to end delay, energy consumption and additionally average throughput for comparison of seemon and existing system by using varying n/w scenarios and varying mobility.

***Index Terms*— PDR , Essential Sensor Set, T-QSet, F-QSet, U- QSet,SeeMon,Packet Delivery Ratio**

INTRODUCTION

In this paper, purpose SeeMon,a scalable and energy efficient context monitoring framework for personal context-aware applications. By using this context aware application, it continuously monitor user's context and capture their changes over time.SeeMon approaches the context monitoring problem by using two methods[1].

In first it uses unidirectional approach where the processing flow proceeds in one direction through a pipeline, which consists of several stages i.e preprocessing, feature extraction, context recognition and change detection.

In second it uses bidirectional approach, it forms a feedback path from the applications to sensors. This approach gives the opportunity to achieve a high degree of efficiency in computation and energy consumption.

I.SeeMon FRAMEWORK: - SeeMon is a middle-tier framework between personal context-aware applications and a personal sensor networks[1]. SeeMon provides programming APIs and a runtime environment for applications. , SeeMon receives and processes sensor data and controls the sensors in the personal sensor network. For the wireless communication between them, protocols such as Bluetooth and ZigBee can be used. In addition to wireless personal sensor network, device attached sensors such as accelerometer and gyroscope deployed on smart phones can be easily incorporated in the SeeMon framework without architectural change. SeeMon consists of four components: the CMQ

Processor, the Sensor Manager, the Application Broker, and the Sensor Broker. Based on these components, the operation of SeeMon is performed in three phases: query registration, query processing, and sensor control.

II.SeeMon API's:-The SeeMon has following API's for proper functioning[1]

registerCMQ(CMQ_statement)
deregisterCMQ(CMQ _Id)
createMAP(Parent_Map_Id)
deleteMAP(Map_Id)
InsertContextElement
DeleteContextElement

III.CONTEXT MONITORING QUERY

SeeMon provides CMQ, an intuitive monitoring query language that supports rich semantics for monitoring a wide range of contexts. It is important for applications to catch the changes in users' context proactively.

The CMQ template has the following format[1]:

Context <context element>
(AND <context element>)*
ALARM <type>
DURATION <duration>

In the SeeMon calculating the ESS is a complicated problem. CMQs can be evaluated using a small number of sensors. It is simple to calculate ESS for the true-state CMQs and undecided-state CMQs. However, it is complicated to compute the set of essential sensors with minimum cost for the false-state CMQs. We call this problem minimum cost false-query covering sensor selection (MCFSS).By using this context monitoring query approach we can evaluate ESS Problem[i].

Here also we have to use sensor control mechanism for controlling sensors based on the ESS calculation result. It sends a control message to the sensors that are not included in the calculated ESS. The control message configures the sensors to be put into the inactive state so that the sensors stop transmitting data. Afterward, the ESS Calculator updates the state of context elements related to the inactive sensors in the CMQ-Table.

Table 1.CMQ Table

Query ID	State	Context Element	Period	Time Stamp
Q1	True	[f1.(bo,b1),true) [f2,(b4,b7),false)	12:00	11:45:12
Q2	False	[f1.(bo,b1),false) [f2,(b4,b7),false)	Null	11:49:15
Q3	True	[f1.(bo,b1),false) [f2,(b4,b7),true)	11:00	12:56:18

CMQ Table can be represent as follows.

In CMQ table we will get query id, state of query either it is true or false, period required for accessing query and time when query is executed as shown in table 1.

This paper implementing ESS algorithm for calculating Essential Sensor Set. This paper will also employ Greed MCFSS algorithm to reduce the energy cost as much as possible while simplifying the computation in the broader sense. By using this above mentioned algorithm we can can easily solve ESS calculation problem and also reduce energy cost and time.

Abbreviations and Acronyms

ABBREVIATION	ILLUSTRATION
CMQ	Context Monitoring Query
MCFSS	Minimum Cost False Query Covering Sensor election
ESS	Essential Sensor Set
T-QSet	Set of All True State CMQs.
F-QSet	Set of All False State CMQs
U- QSet	Set of All Undecided State CMQs.
PDR	Packet Delivery Ratio

ALGORITHM

I.ESS Algorithm:-

S:a set of all true-state CMQs

T-QSet:a set of all true-state CMQs.

F-QSet:a set of all false state CMQs.

U- QSet:a set of all undecided –state CMQs.

q.sensor:a set of sensors which are associated with the context elements of a CMQ q

Steps:

```
1.  TQCover,UQCover,TUQcover,RF-QCover=ϕ
2.  For qi, where qi belongs T – Q set.
        TQCover = TQCover Union qi.sensor.
3.  For qi ,where qi belongs U -QSet.
        UQCover = UQCover Union qi.sensor.
4.  TUQCover =TQCover Union UQCover.
5.  RF-QSet =FQSet
6.  For si ,where si belongs TUQCoverF
        If qi evaluates to false by sensor si,then RF-QSet belongs RF-QSet – set of qi.
7.  For qi where qi belongs RF-QSet
        If qi evaluates to false by sensor si, then RF-QCover belongs RF-QCover Union set of si.
8.  MCFQCover belongs Greedy-MCFSS
9.  ESS=TUQCover union MCFQCover.
10. Output is ESS
```

II.Greed MCFSS Algorithm.

F-QSet: a set of false-state CMQs.

S: Set of sensors,each of which covers a subset of F-QSet.

```
1. M =ϕ  //Minimum cost of subset.
   S'=S.
2. While F-QSet'(M) subset FQSet do
     Find Sc
     Where $a(s) = \frac{COST(Si)}{|F\text{-}QSet'(Si)\ intersect\ F\text{-}QSet-F\text{-}QSet'(M)|}$
       S'=S'- set of Sc
```

RESULTS

In this paper based on our aims and objectives, we are discussing the results from simulation studies. Here used different conditions to check the performance of proposed algorithm against the existing routing algorithms. Here used performance metrics such as packet delivery ratio, end to end delay, energy consumption and additionally average throughput. Following points explaining the every case and presenting their results:

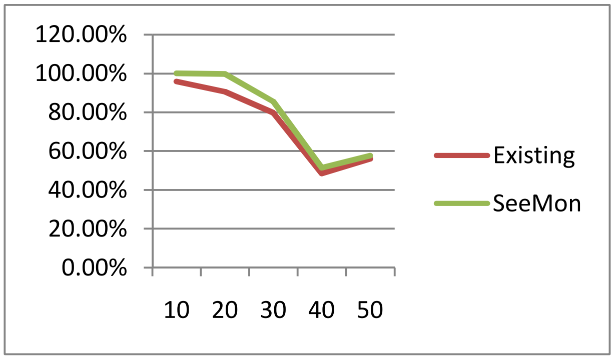
A. Results According to Varying Network Scenarios

In this first case, here used three network scenarios such as 10 nodes, 20 nodes, 30 nodes, 40 nodes and 50 nodes in order to evaluate the routing protocols performances. Network size is varying from 500x500 to 1000x1000 with constant simulation time 200-300 sec.

1. Packet Delivery Ratio

This is also one of the major performance metrics which evaluates the performance of routing protocols and TCP variants. packet delivery ratio is nothing but: PDR: total number of packets generated / total number of packets received. From the following graph 1A, the PDR of proposed robust routing protocol is better as compared to existing routing algorithms.

GRAP 1A: PDR (%) Vs Data Scale



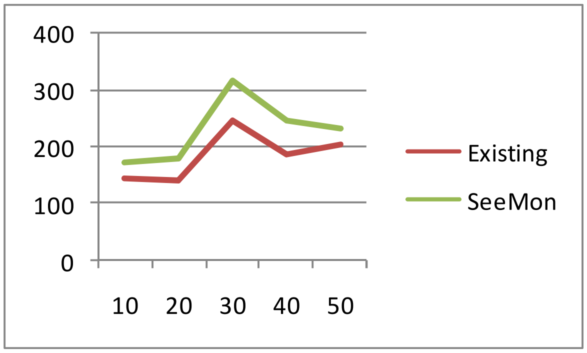
2. Average Throughput Performance

Throughput is the ratio of total amount packets the receiver will receive from the source of the data within the specified time frame. End to end delay for the packet transmission is most important metrics for the throughput performance of the routing protocols. Following graphs showing the SeeMon framework for WSN having better throughput as compared to the existing context recognition based monitoring method.

Table 2A. Average Throughput

Nodes	Existing	SeeMon
10	143.95	171.92
20	140.77	179.72
30	247.25	315.62
40	186.14	245.31
50	204.57	230.66

GRAP 2A: Throughput Vs Data Scale



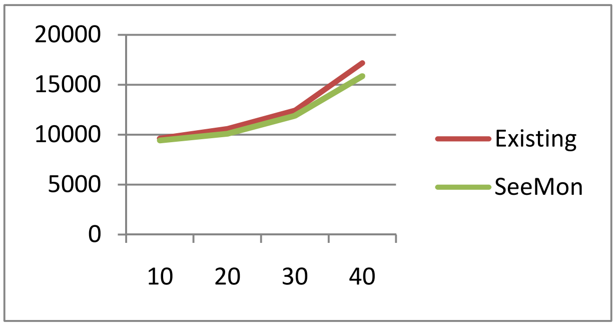
3. Total Energy Consumption

In the simple case, the energy consumed by the network interface when a host sends, receives or discards a packet can be described using a linear equation

Energy=m x size + b

Table 3A. Total Energy Consumption Readings

Nodes	Existing	SeeMon
10	9597	9398
20	10575	10058
30	12398	11861
40	17180	15863
50	18923	17605



GRAP 3A: Energy (J) Vs Data Scale

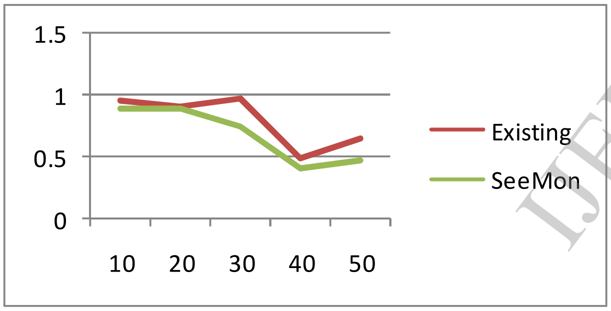
4. Average End to End Delay

This metrics calculates the time between the packet origination time at the source and the packet reaching time at the destination. Here if any data packet is lost or dropped during the transmission, then it will not consider for the same. Sometimes delay occurs because of discovery of route, queuing, intermediate link failure, packet retransmissions etc are considered while calculating the delay. Such kind of metrics we have to measure against the different number of nodes, different traffic patterns and data connections.

Table 4A.Average End to End Delay Readings

Nodes	Existing	SeeMon
10	0.957	0.8991
20	0.905	0.895
30	0.9759	0.75
40	0.4887	0.4135
50	0.6589	0.4756

GRAP 4A: Delay (Sec.) Vs Data Scale



B. Results According to Varying Mobility/CMQ

In this second case performing the simulations based on varying number of mobility used. This used mobility varying in between 5 M/S to 20 M/S. Following the results presented for the same: Numbers of nodes are 10 with 500x500 network area and 200 seconds simulation time.

1. Packet Delivery Ratio(PDR)

GRAP 1B: PDR (%) Vs Mobility (M/S)

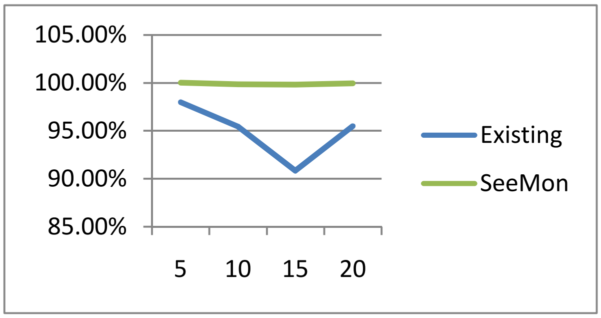


Table 1B.PDR Reading

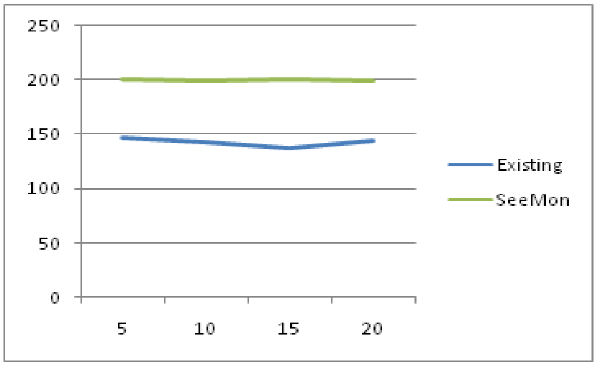
Mobility(M/S)	Existing	SeeMon
5	97.98%	100%
10	95.45%	99.83%
15	90.83%	99.80%
20	95.48%	99.92%

2. Average Throughput Performance

Table 2B.Average Throughput Performance

Mobility(M/S)	Existing	SeeMon
5	147.04	200.27
10	143	198.66
15	137.03	199.48
20	143.63	199.25

GRAP 2B: Average Throughput performance



3. Average Energy Consumption

Table3B.Average Energy Consumption

Mobility(M/S)	Existing	SeeMon
5	72.49	33
10	72.02	36.4
15	72.41	40.7
20	70.75	43.4

4. Total Energy Consumption

Table 4B.Total Energy Consumption

Mobility(M/S)	Existing	SeeMon
5	724	331
10	720	364
15	724	407
20	707	434

5. Average End to End Delay

Table 5B.Average end to end delay

Mobility(M/S)	Existing	SeeMon
5	0.9784	0.8993
10	0.953	0.8969
15	0.9069	0.8966
20	0.9534	0.8978

CONCLUSION

This paper implemented ESS and Greed MCFSS Algorithm.Here used performance metrics such as packet delivery ratio, end to end delay, energy consumption and additionally average throughput for comparison of seemon and existing system by using varying n/w scenarios and varying mobility.By using this paper can conclude that seemon framework is better than existing system in respected to performance metrics.

REFERENCE DETAILS:

[1] Seungwoo Kang,Jinwon Lee,Hyukjae Jang,Youngki Lee,Souneil Park,and Junehwa Song," A Scalable and Energy-Efficient Context Monitoring Framework for Mobile Personal Sensor Networks,"IEEE Transaction On Mobile Computing,VOL 9,No.5,May 2010

[2] T. Hofer et al., "Context-Awareness on Mobile Devices—The Hydrogen Approach," Proc. Hawaii Int'l Conf. System Siences, 2003.

[3]. A. Rahmati and L. Zhong, "Context-for-Wireless: Context-Sensitive Energy-Efficient Wireless Data Transfer," Proc. MobiSys, 2007.

[4] S. Chakraborty et al., "On the Effectiveness of Movement Prediction to Reduce Energy Consumption in Wireless Communication," IEEE Trans. Mobile Computing, vol. 5, no. 2, pp. 157-169, Feb. 2006.

[5] G. Anastasi et al., "Performance Measurements of Motes Sensor Networks," Proc. Int'l Symp. Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), 2004.

[6]. V. Shnayder et al., "Simulating the Power Consumption of Large-Scale Sensor Network Applications," Proc. Int'l Conf. Embedded Networked Sensor Systems (SenSys), 2004.

[7]. R.S. Sandhu and P. Samarati, "Access Control: Principles and Practice," IEEE Comm. Magazine, 1994.

First Author

	**Mr.Avinash Palave** completed B.E. (I.T.) From N.M.U, Jalgaon in 2006.Now presently doing M.Tech (I.T.) From Technocrats Institute Of Technology, Bhopal M.P.

Second Author

	**Prof.Dr.Shiv Kumar** B.Tech(I.T.) From Bhagwat Institute Of Technology,Muzzaffarnagar U.P in 2006.He have received M.Tech(Honors,I.T.) from Technocrats Institute Of Technology, Bhopal M.P. in 2010.He have completed Ph.D.(C.S.E) from Banasthali University,Tonk(Rajashtan,India) in 2012.His research area interest includes voice signal compression, Image Processing, Voice Recognition. He is having 30 papers/articles published in International Journals. Now currently He is working as Professor and HOD.IT in Technocrats Institute of Technology, Bhopal since November 2006.

Third Author

	**Prof. Amit Sinhal** completed his B.E. in Computer Engineering from NIT Surat in 1996, M.Tech in Computer Science & Engineering from SATI Vidisha in 2005 and pursuing Ph.D. from Rajiv Gandhi Technical University, Bhopal. He worked in various reputed software development companies as Project Lead and University Institute of Technology, Barkatullah University Bhopal as Assistant professor. He is having 15 papers/articles published in International Journals. Currently he is working as professor in Technocrats Institute of Technology, Bhopal in CSE Department