Research Journal of Applied Sciences, Engineering and Technology 8(1): 24-34, 2014
DOI:10.19026/rjaset.8.936
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: November 11, 2013 Accepted: December 17, 2013 Published: July 05, 2014

Research Article

Sub-head Transmission of Heterogeneous Data by Cloned Agent to Android Mobile

[1]S. Karthikeyan, [1]S. Shanmuga Priya, [2]S. Jayashri and [3]V. Saravanan
[1]Sathyabama University, Chennai-600119, Tamil Nadu, India
[2]Adhiparasakthi Engineering College, Melmaruvathur 603319, Kanchipuram District, Tamil Nadu, India
[3]SRR Engineering College, Tamil Nadu, India

Abstract: The occurrence of an event can be detected using sensor nodes. The eminent constraint of these sensor nodes is the non-replenishing battery. Conservation of energy is an important aspect due to its limited battery power. Energy utilization using cloned mobile agents is the ultimate aim in this proposed work. The network is divided into clusters and 2 subheads are elected from each cluster. The event occurred is sensed by a sensor and is transmitted to the subhead. The cloned mobile agent collects data from the subhead and transmits it to the sink. The gathered data from sink is then transferred to android mobile. The advantages of this mechanism are less delay and reduced energy consumption. When the energy of existing subhead is reduced to half of its initial energy, new subhead is selected. Thus energy utilization is minimized and overall lifetime of network is increased.

Keywords: Cloning, energy utilization, heterogeneous, lifetime, mobile agent, sub-head

INTRODUCTION

Sensors are battery-driven devices that operate on a limited energy. A wireless sensor network is a collection of nodes organized into a network and a distributed system consisting of a base station and a number of wireless sensor nodes equipped with wireless radio communication capabilities. Due to the compactness and low cost of wireless sensors, they can be embedded and distributed at a fraction of the cost of conventional wired sensors. A wide range of tasks can be performed by these tiny devices, such as remote object monitoring and tracking; detection of the presence or absence of certain elements; and condition-based maintenance among other special applications.

Sensors should have a reasonable lifespan to be deployed. In a network with thousands of sensors, battery replacement is not an option, thus requiring efficient management of the energy resources. Sensor networks have been receiving attention not only from academia, but also from the Government and Industries.

Energy consumption is a major factor in determining the lifespan of a sensor node. It is well known that a sensor utilizes a significant amount of energy while sending or receiving packets. Power dissipation is also expressive even when a sensor receives a packet not intended for it. State-of-the-art systems support a number of power states of the sensor nodes. To simplify matters, the assumption here is that each sensor node in a WSN has two.
Power states:

- Awake
- Asleep

Although energy dissipation is negligible in the asleep mode, a significant amount of energy is utilized in the awake state.

The efficiency of a protocol will be assessed by two metrics:

- The overall amount of time required by the protocol to terminate
- For each individual sensor node, the total amount of time must be awake to transmit/receive packets

The goals of optimizing these parameters are conflicting. Sometimes, one can easily minimize the overall completion time at the expense of energy consumption and vice versa. The challenge is to strike a sensible balance between the two by designing the protocols that are time and energy efficient (Jacir, Year).

In many applications, the sensor nodes must aggregate the sensed/monitored data. Because the sensor nodes are empowered with the ability to share their observations and coordinate among themselves to gather and process information, meaningful information can be transferred to the base station. Such information can then be retrieved and used to control the environment from remote locations.

The WSNs constitutes many tiny sensor nodes, which is capable of communicating by means of transceivers. These sensor nodes work with low power

Corresponding Author: S. Karthikeyan, Sathyabama University, Chennai-600119, Tamil Nadu, India

batteries which are not rechargeable. The base station is equipped with a powerful antenna that enables it to monitor all the sensor nodes under consideration. Taken into assumption, the energy for data transfer between sensor nodes will be lesser that the energy for transmission from nodes to it base station. This assumption is based on the fact that one can increase the base station's receiver front-end sensitivity and make it reasonable.

In WSN, there are two possible communication modes; single-hop and multi-hop. The direct data transmission from a sensor node to another is said to be a single hop communication (Jacir, Year). In a Multi-hop WSN, the communication between two sensor nodes may involve a sequence of hops through a chain of pair wise adjacent sensor nodes. The sensor nodes transmit data to the base station via single hop communication and data transmission between nodes can follow either single hop or multi-hop communication (Mohammad and Imad, 2005). Multiple agents are used in parallel for gathering data from Subhead. Multi-hop techniques have more latency when compared to single-hop methods.

Mobile agent existence in WSN: A mobile agent is a software program, capable of moving across sensor nodes. The Lifecycle of mobile agent involves seven stages. They are Agent creation, transmit to host, cloning of agent, deactivation, activation, retraction and discarding. The agent is initially created in the home machine and then dispatched to host machine. The mobile agent contains the program code and execution state, which holds its state information such as the current state of execution, next instruction etc., (Francesco *et al*., 2010; Karthikeyan and Jayashri, 2012a). The agent will be dispatched along with its code and execution state. The host where the mobile agent is dispatched is called as the mobile agent server or mobile agent platform.

The mobile agent executes in the host machine (Abdelkader, 2009), whose environment is suitable for its execution. The host machine provides all its resources for the mobile agent to perform its operation. Total execution time for the entire task is given by the size of the results that the agent is required to carry after remote processing, program code, program execution state, data code. The mobile agent traverses from one host machine to another after performing its task in the first host machine. The agent is capable of executing its task from where it left in the previous host machine with the aid of the execution state information, which is also transferred along with the agent when it migrates (Karthikeyan and Jayashri, 2012b). Thus the execution state information helps the agent to resume its operation rather starting from the beginning. Similarly the agent migrates all the machines and after completing its operation in the last host machine, it returns back to the home machine from where it is dispatched (Sajid *et al*., 2006; Osborne and Shah, 2003). This completes a single itinerary of the mobile agent (Parineeth, 2002).

Life Cycle of Mobile Agents:

- Initially the mobile agent is created in a machine, which is its home machine.
- The mobile agent is dispatched to the host machine A for execution.
- The agent gets executed on host machine A.
- After its execution in host machine A, the agent is cloned to create 2 copies. One copy is dispatched to host machine B and the other is dispatched to host machine C.
- The cloned agents get executed on their respective host machines.
- After execution of the mobile agent in the host machines B and C, they will move back to home machine, where it is created.
- The home machine receives the data from the all the agents it has taken back.
- The home machine analyses the data received and finally dispose the agents (Parineeth, 2002; Sundari and Sankarnarayanan, 2011).

LITERATURE REVIEW

Chandreyee and Sarmistha (2010) proposes the dependable aspect, especially reliability issues that need to be addressed before the mobile agents can be used for a broad range of commercial applications in Mobile Adhoc Network (MANET). Also he proposed an algorithm for estimating the task route reliability of the Mobile Agent System (MAS), which are based on the conditions of the underlying wireless network and estimates the reliability of mobile agent system considering different failures of the underlying network. The results achieved demonstrate the robustness of the proposed algorithm.

Tian (2010) presented a new method of real-time downloading and distributed computing model is studied based on the mobile Geographic Information System (GIS) environment. In order to deal with the narrow bandwidth and the instability of the wireless internet, distributed computing of tremendous spatial information, real-time request for spatial information based on limited processing speed and low memory of mobile devices, a new mobile agent based Real-time downloading and Distributed Mobile GIS model (RDMG) is further proposed that has a high processing efficiency, less network communication, good load balance and thus suitable for mobile real-time spatial information downloading and distributed computing. Field tests say that the proposed model and methods are feasible and adaptable for real-time downloading and distributed computing of spatial information based on mobile GIS environment.

Monaco *et al*. (2006) has addressed the problem of optimal data gathering in Wireless Sensor Networks (WSNs) is addressed by means of optimization techniques. The basic objective of his work is to lay to develop algorithms and techniques that minimize the data gathering latency and also balance the energy consumption among the nodes, in order to maximize the network lifetime. He formulated his static routing problem for large and dense WSN's. An accurate network model is proposed that captures the trade-off between the data gathering latency and the energy consumption, by modelling the interactions among the routing, medium access control and physical layers.

Mário *et al*. (2009) presented a new TDMA based MAC protocol for Wireless Sensor Networks (WSNs), especially suited to extend the lifetime of networks supporting alarm-driven, delay-sensitive applications characterized by converge cast traffic patterns and sporadic traffic generation named as Latency-Energy Minimization Medium Access (LEMMA). The LEMMA's timeslot allocation protocol makes decisions based on the interference actually experienced by the nodes, instead of following the simple but potentially ineffective n-hop approach. The simulation results of this work demonstrate the ineffectiveness of the n-hop time-slot allocation in comparison with LEMMA, as well as to evaluate the performance of LEMMA against the L-MAC, T-MAC and Low Power Listening.

Rachuri and Siva Ram Murthy (2011) put forth an idea to modify the random walk to take some level biased steps to improve its energy efficiency and latency which is important design parameters of protocols for WSNs. The level of a node is defined as the minimum number of hops in which it can reach the sink node. He proposed three protocols viz., Several Short Random Walks (SSRW) search, Random Walk with Level Biased Jumps (RWLBJ) search and Level Biased Random Walk (LBRW) search. The proposed protocols use a combination of random and level biased steps to search for the target information. The simulations and test bed experiments show that SSRW, RWLBJ and LBRW are better choices compared to that of a pure Random walk in terms of the energy consumption and latency of search.

Kun-Hsien *et al*. (2012) presents a Hierarchical Ring-based Data Gathering (HRDG) scheme for dense wireless sensor networks. A hierarchical grid structure is constructed and only some sensor nodes are elected as grid heads for gathering data, subsequently reducing the total energy consumption per round. The simulation results of the proposed work indicate that the proposed HRDG scheme outperforms other data gathering schemes in terms of the number of rounds, the energy × delay cost and coverage ratio.

Waleed and Mohamed (2010) formulate an optimization model for the asset planning problem and present effective algorithms for solving it. The proposed solution scheme employs contemporary search heuristics such as k-means and genetic algorithms. Validation results confirm the effectiveness of our approach in achieving the desired design goals.

Kan *et al*. (2011) discussed the use of Forward Error Correction (FEC) codes in WSN in order not only to improve the link reliability but also to reduce the number of retransmissions in harsh industrial environments and proposed an FEC scheme suitable for MAC level protection where the packet is divided into groups and encoded using systematic FEC codes. They have implemented different FEC codes in a typical WSN chip to evaluate memory consumption and to ensure that we are not violating the strict timing rules for acknowledgment. So it is obviously understood from the output that only some FEC codes are suitable to be implemented in a typical WSN node while several others fails because of memory and long encoding and decoding issues.

METHODOLOGY

The energy conservation in sensor nodes is a significant factor that is considered in network design. The proposed work also targets at conserving energy by utilizing various techniques for data gathering and subhead selection. Here hundreds of sensor nodes are deployed and distributed uniformly across the heterogeneous environment. These nodes being large in numbers are divided into 3 clusters. Each cluster holds more than 70 nodes and it is further divided as two sections. Each section has 35 nodes a subhead.

Pseudo code:

Overall idea of the algorithm:

- Add nodes to subhead 1, 2 in each cluster
- Identify the trigger nodes in each cluster and store SH
- Create a sink node which use a mobile agent together from SH of each cluster
- Display the data obtained by mobile agent ' Mobile Agent Data'

Declaration part:

Cluster: C_i Containing 70 nodes
$SH_{i;j}$: j^{th} Subhead of i^{th} Cluster
$Etn_{i,j,k}$ = k^{th} event triggered node of j^{th} subhead in i^{th} cluster

$Etn_{i,j,k}$ specification:

```
Etn i, j, k
{
var node id;
var cluster id;
var time t;
}
```

Initialization:

Cluster I = 3
Cluster J = 2
Node K = 35
Subhead $SH_{i,j}$ = ø
Mobile Agent Data MA data = ø

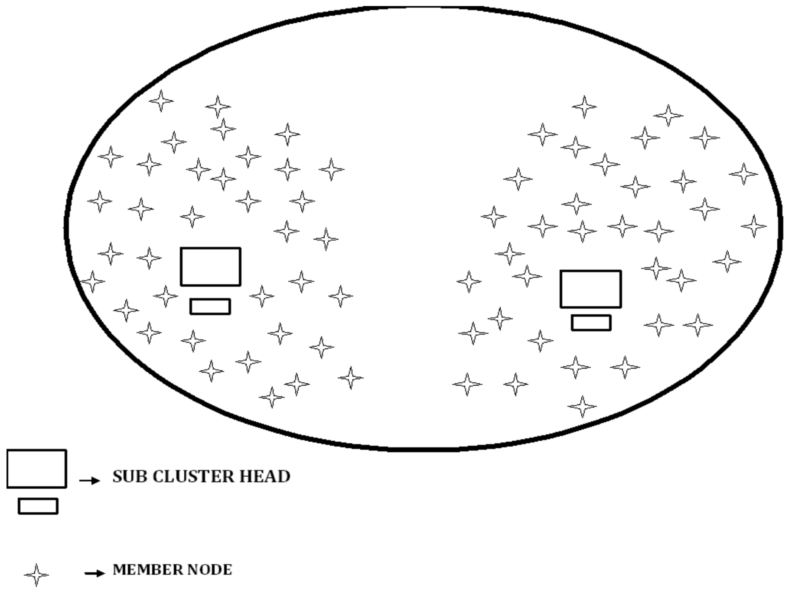

Fig. 1: Cluster with partition of sub cluster head

Cluster formation:

For each cluster e_i
where I = 1 to 3
For each cluster SH_j
where J = 1 to 2
For each node K
where k = 1 to 35
$SH_j = SH_j$ U node $_k$
end K
end J
end I

Event occurrence:

For each C_i
For each SH_J
For each node k
If event triggered
then
$SH_{I,J} = SH_{I,J}$ U Etn $_{i,j,k}$
end if
End k
End J
End i

Data gathered by cloned MA to sink node:

Create a Mobile Agent MA
N Represents 6 SH
For M = 1 to N-1
Clone MA = $MA_{(M)}$
MA_N = MA U MA_M
For I = 1 to n (For each cluster i)
MA data = MA_N data U $SH_{I,J}$
MA data = ø
End for
Write MA data
End procedure

Since there are larger number of nodes, more data will be sensed by these nodes and transmitted to the corresponding subhead. A single subhead cannot be adequate to collect of sensed data from all the nodes and hence two subhead per clusters is used. As shown in the Fig. 1, SCH0 and SCH1 will be the Subheads for Cluster1.

The sub cluster head is elected based on the residual energy of sensor nodes. The new subhead selected only when the energy of the existing subhead is reduced to half of its initial energy. By which the same node acts as the subhead. This will minimize the energy for subhead election process.

Since it is a heterogeneous network, the nodes in each cluster will sense different environment data. The nodes will communicate the sensed information to its respective subhead. The sensed data transmitted from the nodes are gathered in the cluster subheads. Here the data gathering follows the event driven mechanism, in which only the event occurred environment data is gathered by the subhead. Each heterogeneous data have a threshold value defined in its environment. The data which exceeds this threshold value is identified as an event and that data is sensed and communicated by the nodes to its subhead. Such sensed data is further collected by the mobile agent from the subhead. The agent communicates with the cluster subheads instead of the sensor nodes. Thus the mobile agent collects and computes only the event occurred data.

The mobile agent based data communication is opted to reduce the energy utilization for data communication from cluster subhead to the sink. Otherwise the data transmission to sink may happen through single hop or multi-hop routing, which

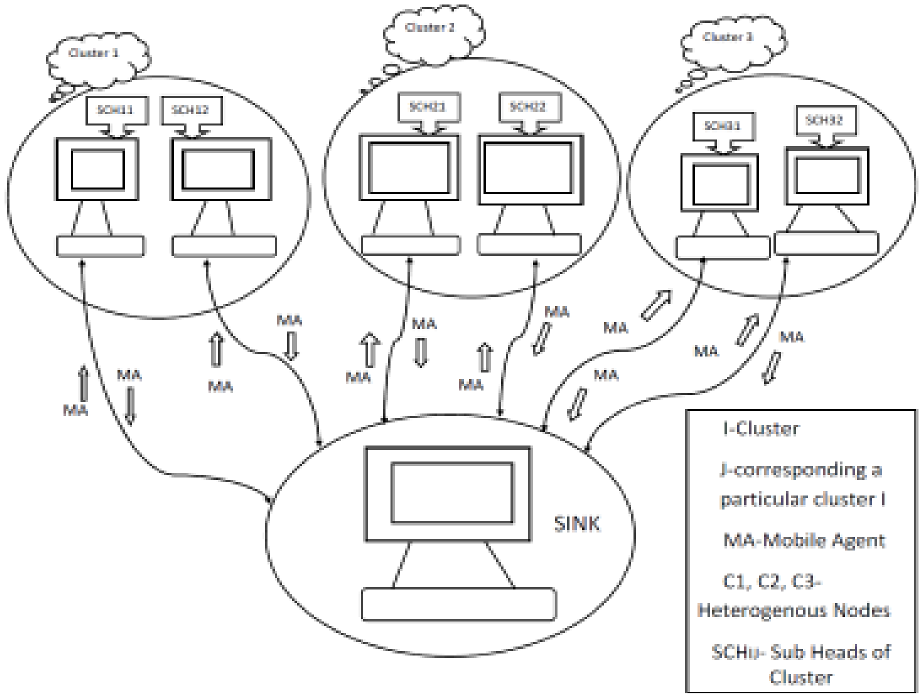

Fig. 2: Multiple mobile agent

Fig. 3: Data transfer to the android mobile

consumes more energy. In addition to that, cloning mobile agent renders parallel communication which is considerably faster. The multi mobile agent communication reduces delay and energy utilization.

The mobile agent is created in the sink and then cloned into multiple agents, thereby forming a multi-agent system is shown in Fig. 2. The MA is cloned N-1 times, where N is the number of subheads. In this study, N represents 6 sub-heads in 3 clusters. The cloned mobile agent migrates to all the subheads at the same time in parallel mode. Each cluster consists of heterogeneous sensor nodes. The MA first should recognize the type of sensor from each cluster. The each agent computes the heterogeneous event occurred data received from the respective subheads and stores the results in the form of a table. The table contains the cluster list of events occurred nodes and its cluster names along with node Id and time of event occurrences. The MA transmits the collected data to the sink and finally it gets disposed in the sink. Thereby the sink receives the computed result from the mobile agent. The parallel communication with the cloned mobile agents is considerably faster comparing to serial communication with a single mobile agent.

GSM mobile is interfaced to the laptop. Data is transmitted through that to the android mobile as shown in Fig. 3. Like an Android mobile the event occurred data is getting in the Android emulator in the laptop itself. Through the android mobile which controls and communicates for different section of the sub-head GSM modem is connected to the computer that acts as sink via a serial port. SMS itself will be sufficient for alerting the user about an event. Received SMS consist of data and the time of occurrence. This extra check is present to ensure that the user gets an intimation on the fault immediately so that the user can respond without delay. This proposed work is suitable for any heterogeneous network application.

SIMULATION RESULTS

In this simulation, there are three clusters as shown in Fig. 3. Each cluster is partitioned into 2 groups of sensor node namely Senor Group 1 and Sensor Group 2. Each group of sensor nodes has sub-head. The GUI in the Fig. 4 shows the data in sub head 1 in the first cluster. Similarly subhead 2 of first cluster will also contain data. Here the sensor nodes in each cluster senses different environment data i.e., temperature, pressure and motor speed. Figure 4 represents the temperature data congregated at the first cluster. Similarly the data for pressure and the motor speed is gathered by other clusters. Thus this simulation results show gathering event occurred heterogeneous data.

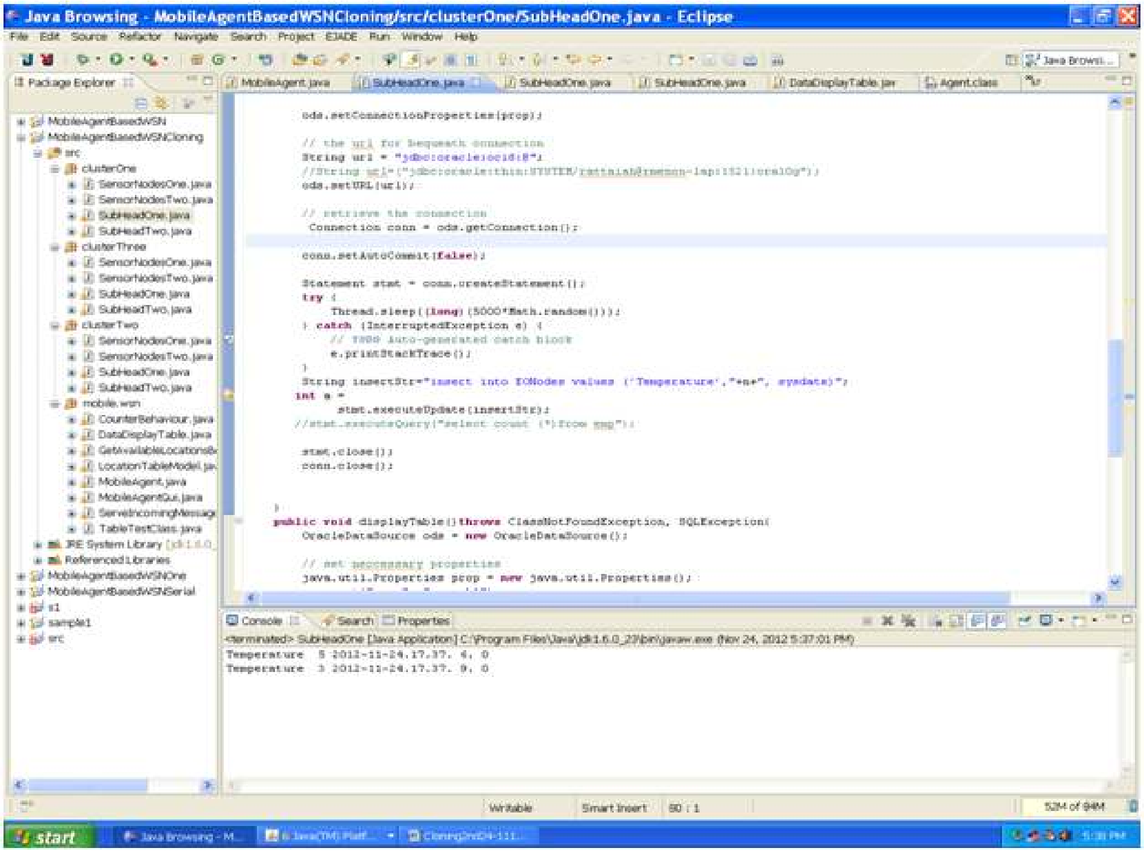

Fig. 4: Cluster partition and data from cluster 1

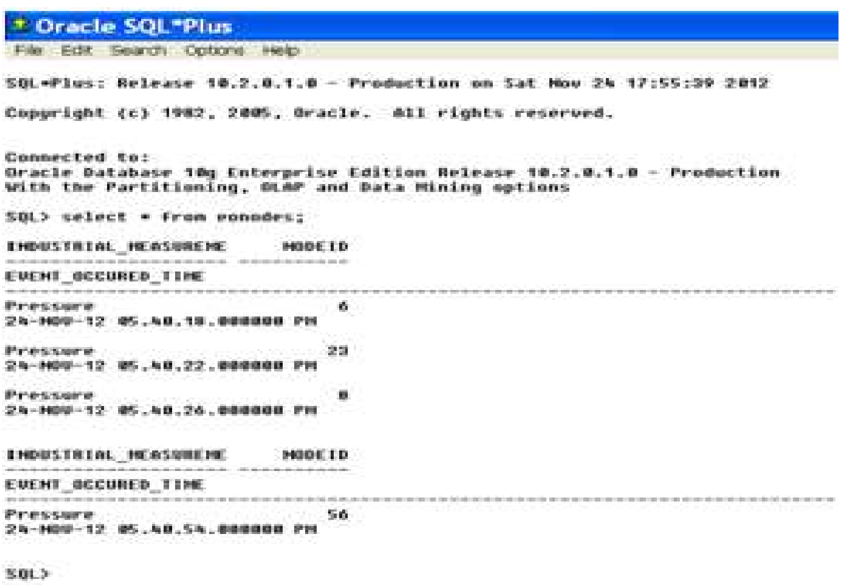

Fig. 5: Cluster partition and data from cluster 1

Data congregated in the second Cluster sub-head is shown in the Fig. 4. The event occurred node ID and the time of occurrence of the event are the information collected as displayed in Oracle SQL Plus in Fig. 5. Similarly, all three clusters with different sensor nodes are used to collect the heterogeneous data with the node ID and time of occurrences.

The MA is created in the sink and the Agent is cloned (N-1) times, which is shown in the Fig. 6. N is the total number of sub-heads, in this simulation it is

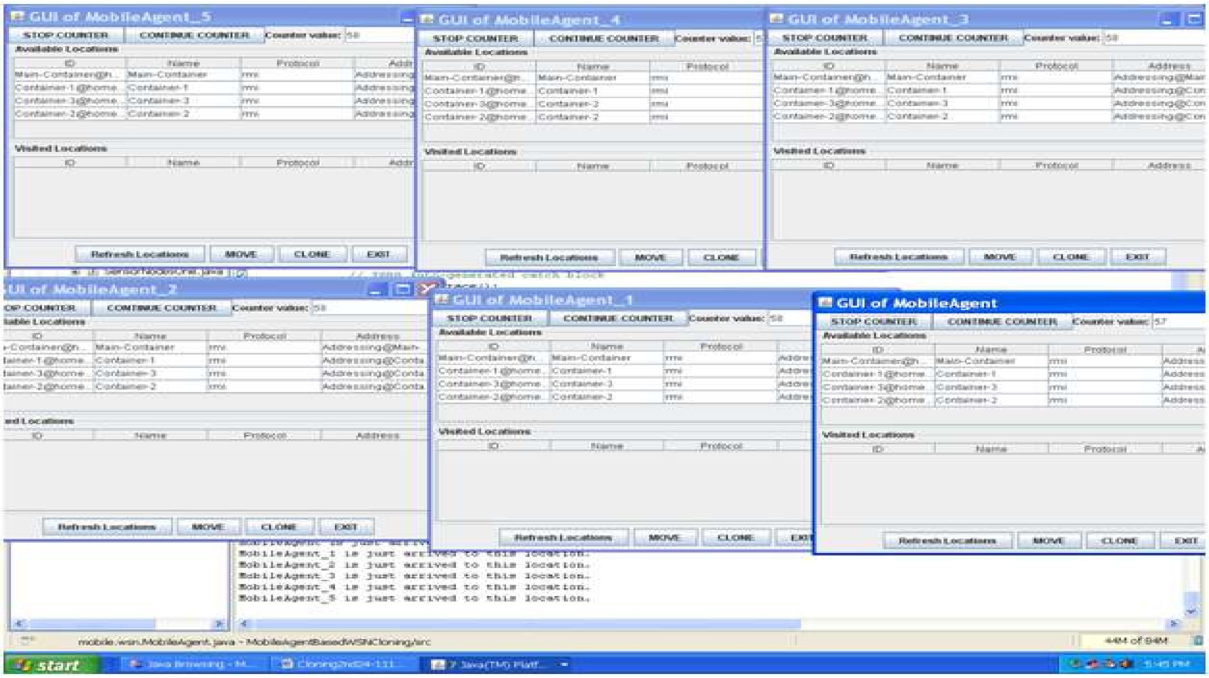

Fig. 6: GUI of multiple agents

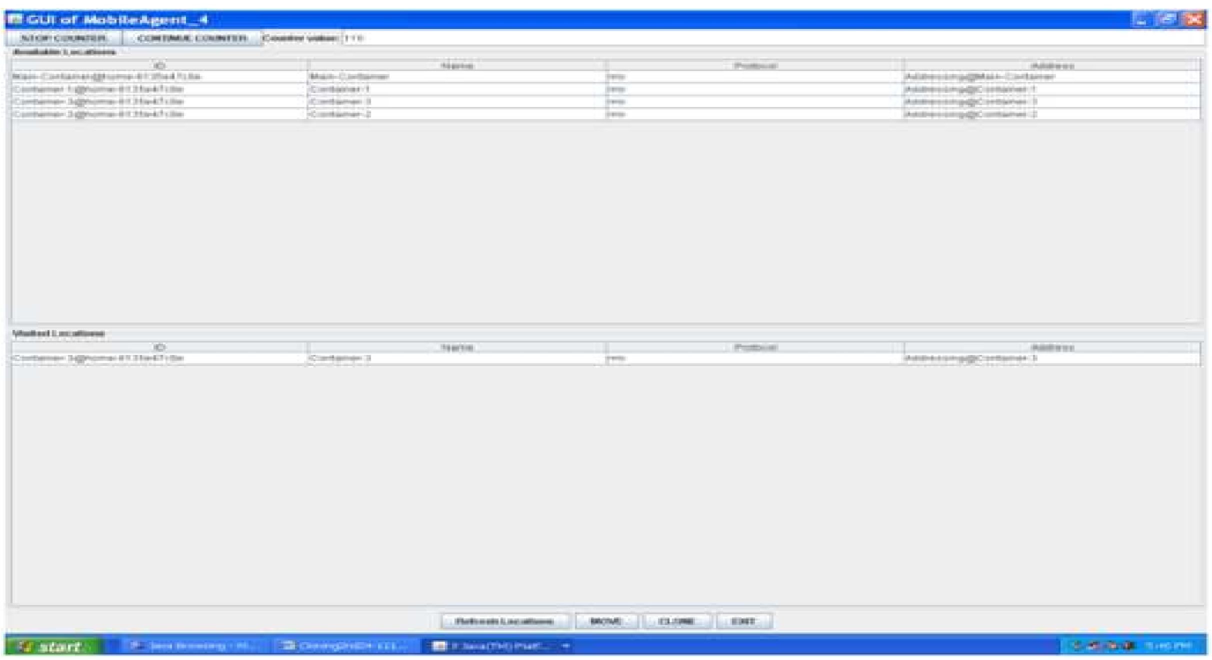

Fig. 7: GUI of MA to third cluster with sub-head 1

six. In this algorithm, all the cloned agents move to their respective sub-head simultaneously and computation is carried out and the resultant value is stored in the table by the agent.

All the agents execute its software code in their respective clusters at the same time and the results of the event occurred are displayed in the tables as shown in Fig. 6. The simulation results are stored in the Oracle database. The node ID and the corresponding cluster ID are stored along with the data as shown in Fig. 6.

Figure 7 and 8 shows the GUI of the MA, where the Agent moves to both the subheads of Cluster 3.

MA computes the data from the respective subheads and the final results of the event occurred node and its details are sent back to the main container.

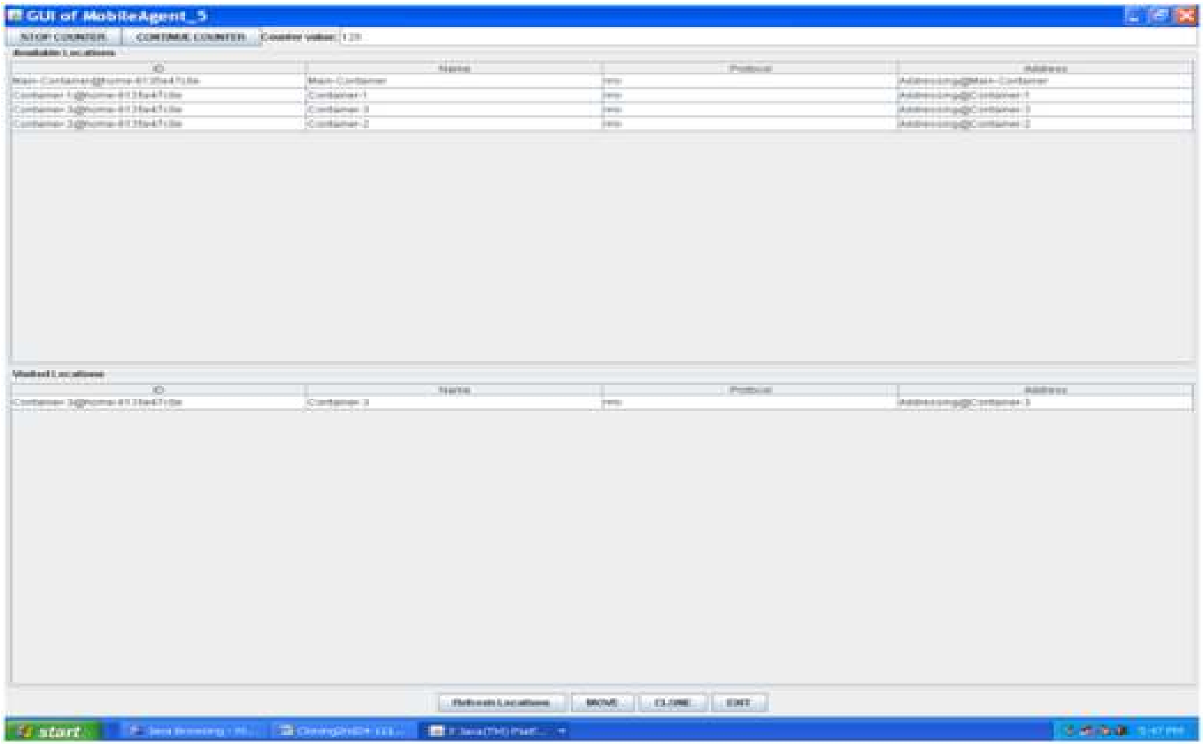

Fig. 8: GUI of MA move to third cluster with sub-head 2

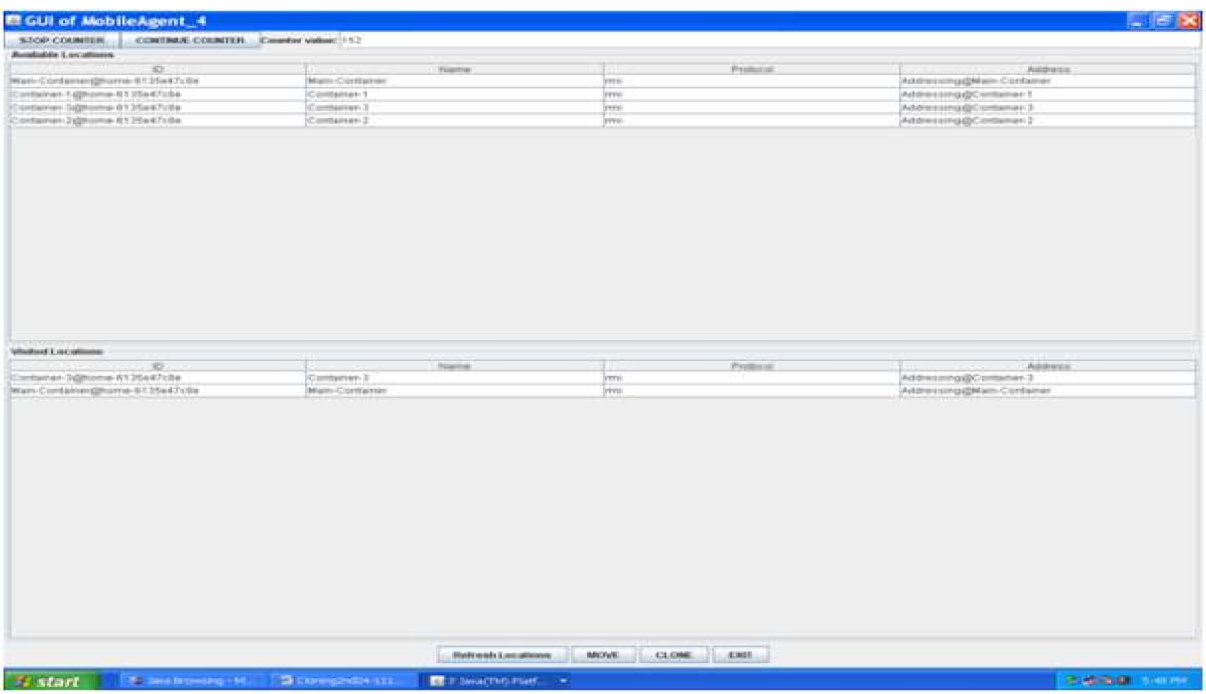

Fig. 9: MA moves to main container from sub-head 1

From the cluster 3 with two sub-head the data are computed sent to the main container is shown in the Fig. 9 and 10.

Time required for the MA movement from main container to sub-head is 1.0172. Scathes mention that time is only for the single hop movement of the MA and excluding computational time.

The resultant value, computed by the mobile agent after fetching data in parallel fashion from all the cluster sub-heads is returned to the main container. The

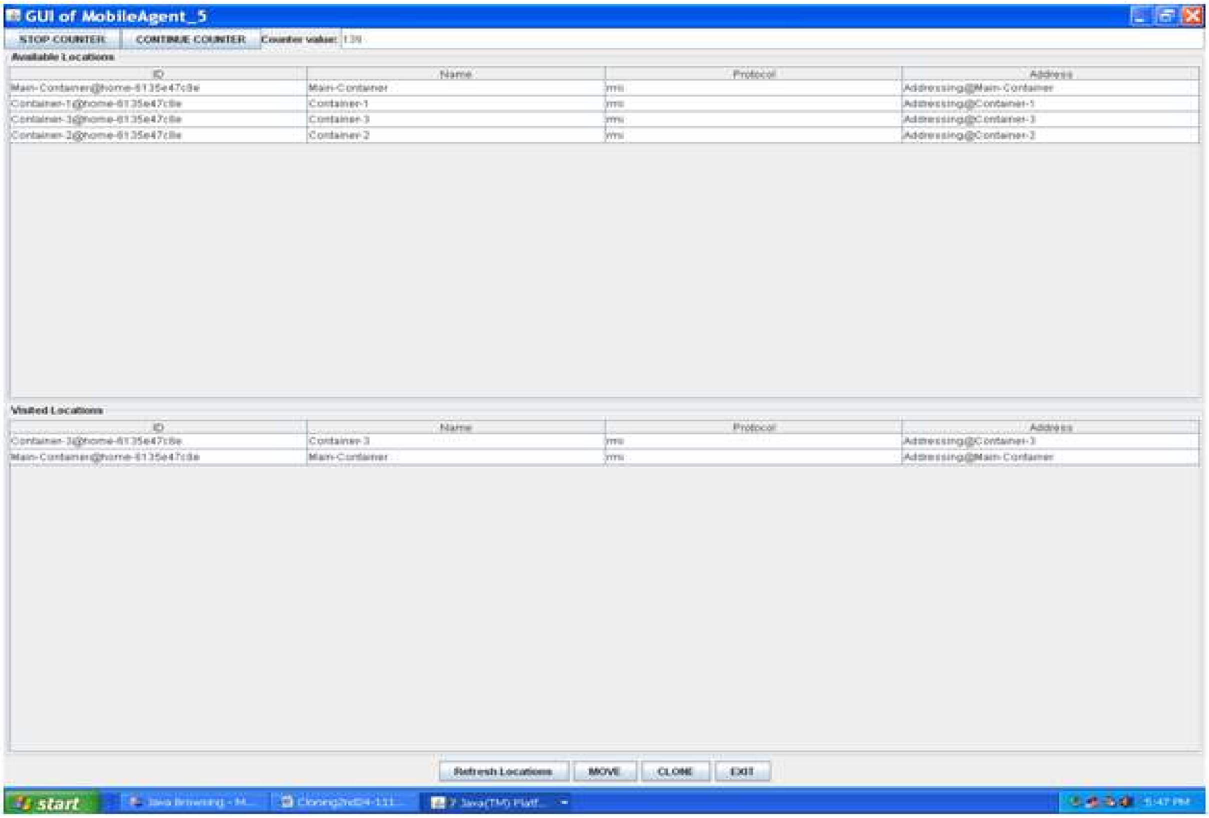

Fig. 10: MA moves to main container from sub-head 2

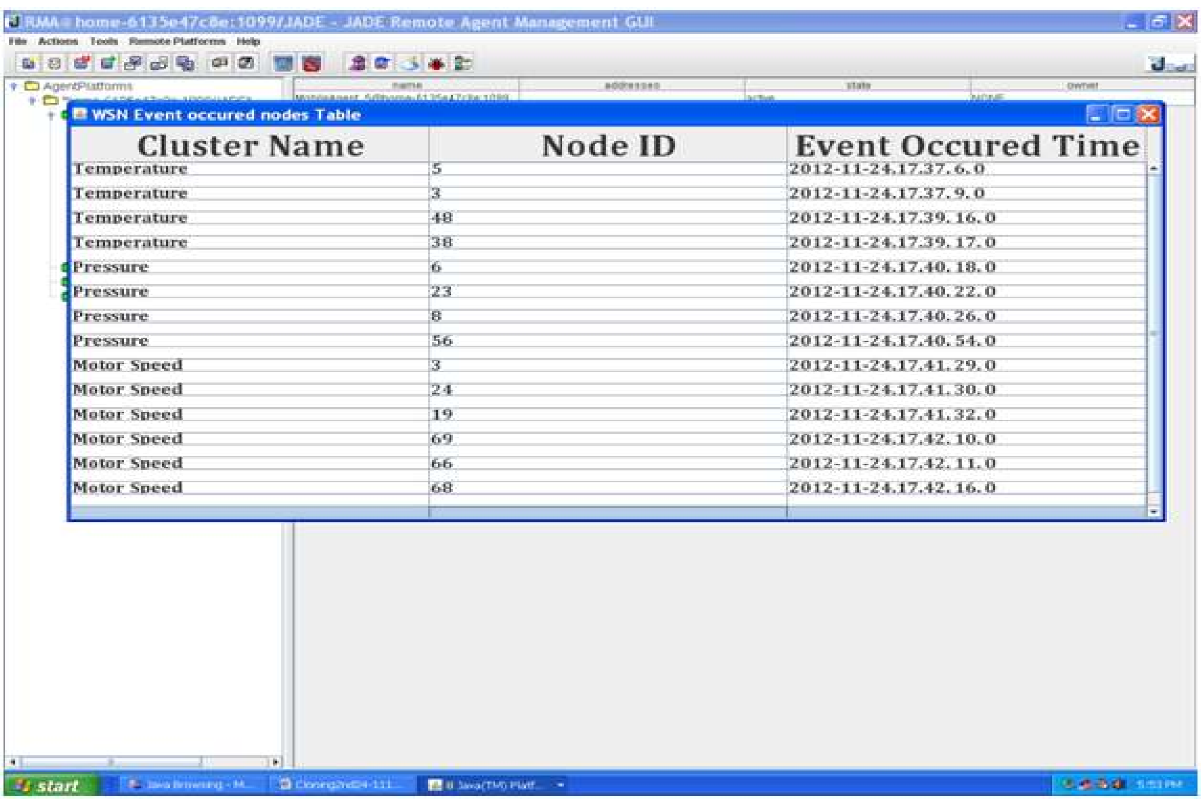

Fig. 11: Resultant value in the main container

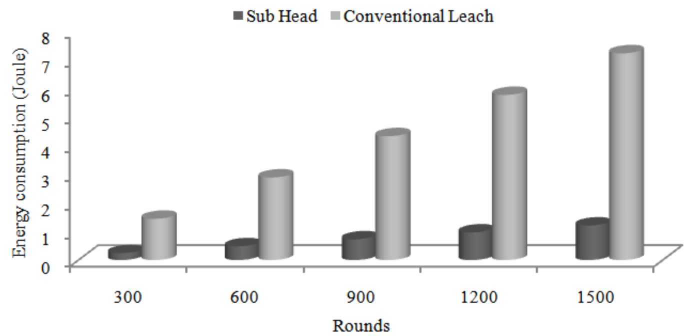

Fig. 12: Energy consumption for 1000 byte of data

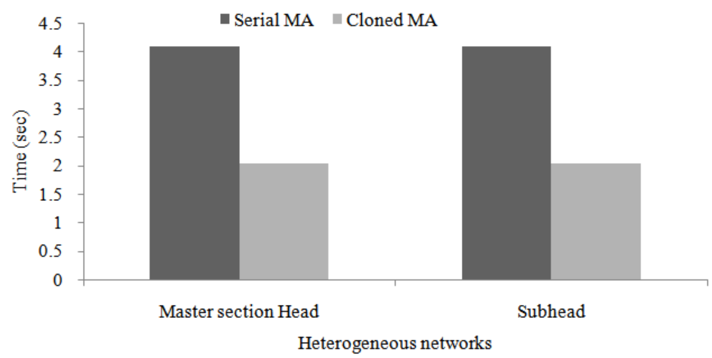

Fig. 13: Heterogeneous networks for mobile agents

Fig. 14: Temperature measurements in emulator

table shown in Fig. 11 displays the results of sensing parameter, the node I.D and event occurred time. This result is stored in the sink in table format as shown in Fig. 11.

Figure 12 explains the energy consumption for 1000 bytes of data transmitted for each round. The energy consumption of the Sub head is lesser by 83.286% respectively, when compared to conventional LEACH for 1500 rounds.

The average computation is analyzed for 3 clusters considering the mobile agent based communication. In case of serial communication the agent takes 4 hops to congregate the event occurred data to the sink. In case of multiple agent based data communication, the agent gathers the data to the sink in 2 hops. From Fig. 13 an analysis is made that the network latency is reduced by more than 50% in the case of multi agent based system when compared to single agent based network communication.

Hence it is proved from the results that the multi agent based parallel data gathering is faster when compared to the serial data gathering. Based on the results, the sink gives data to a mobile device via Android Emulator. The temperature above the threshold values is shown in Fig. 14.

CONCLUSION

In this proposed work, the entire cloned mobile agent migrates at the same time to respective clusters. The data computation takes place concurrently and only the result is transmitted to the sink. As a consequence the energy consumption of the node is reduced. From the simulation results, it is well concluded that the parallel computation by multiple cloned mobile agents consumes less energy. The mobile agents transmit only the event occurred data which further reduces energy consumption. Moreover, the tactic used in the Subhead election for a cluster has contributed to energy conservation which in turn delays the latency of the network. The subhead selection takes place only when the energy of the present subhead is reduced to half of its initial energy. Until then the same subhead is used. With this approach the energy utilization for the selection of subhead is minimized and which in turn delays the latency of the network. Therefore this algorithm diminishes the overall energy exploitation and improves the lifetime of the network.

REFERENCES

Abdelkader, O., 2009. Mobile agent based applications: A survey. Int. J. Comput. Sci. Netw. Secur., 9(11).

Chandreyee, C. and N. Sarmistha, 2010. Estimating reliability of mobile agent system for mobile ad hoc networks. Proceeding of the 2010 3rd International Conference on Dependability (DEPEND). Venice, pp: 45-50.

Francesco, A., F. Giancarlo, G. Raffaele and G. Antonio, 2010. A java-based agent platform for programming wireless sensor networks. Comput. J., 54(3): 439-454.

Jacir, L.B., Year. Handbook of Adaptive Communications Research Laboratories. Hiroshima University, Koji Nakano.

Kan, Y., G. Mikael, A. Johan and B. Mats, 2011. Reliable and low latency transmission in industrial wireless sensor networks. Procedia Comput. Sci., 5: 866-873.

Karthikeyan, S. and S. Jayashri, 2012a. Proficient data gathering by mobile agent using JADE. I-Manage. J. Wirel. Commun. Netw., 1(2).

Karthikeyan, S. and S. Jayashri, 2012b. Cloned agent based data computed in homogenous sensor networks. Int. J. Comput. Sci. Inform. Secur., 10(9).

Kun-Hsien, L., H. Shiow-Fen, S. Yi-Yu, C. Hsiao-Nung and D. Chyi-Ren, 2012. Hierarchical ring-based data gathering for dense wireless sensor networks. Wireless Pers. Commun., 64(2): 347-367.

Mário, M., G. António and N. Mário, 2009. Distributed latency-energy minimization and interference avoidance in TDMA wireless sensor networks. Comput. Netw., 53(5): 569-582.

Mohammad, I. and M. Imad, 2005. Handbook of Sensor Nodes: Compact Wireless and Wired Sensing System. CRC Press, LLC.

Monaco, U., F. Cuomo, T. Melodia, F. Ricciato and M. Borghini, 2006. Understanding optimal data gathering in the energy and latency domains of a wireless sensor network. Comput. Netw., 50(2006): 3564-3584.

Osborne, L. and K. Shah, 2003. Performance Analysis of Mobile Agents in Wireless Internet Applications using Simulation. A Scientific and Technical Publishing Co., OACTA Press.

Parineeth, M.R., 2002. Mobile agents: Intelligent assistants on the internet. Resonance, 7(7): 35-43.

Rachuri, K.K. and C. Siva Ram Murthy, 2011. Energy efficient and low latency biased walk techniques for search in wireless sensor networks. J. Parallel Distr. Com., 71(3): 512-522.

Sajid, H., E. Shakshuki and A.W. Matin, 2006. Agent-based system architecture for wireless sensor networks. Proceeding of the 20th International Conference on Advanced Information Networking and Applications, April 18-20.

Sundari, G. and P.E. Sankarnarayanan, 2011. Multi agent systems using JADE in industrial application. Int. J. Comput. Inform. Syst., 2(4): 21-24.

Tian, G., 2010. A new real-time downloading and distributed mobile GIS model based on mobile agent. Proceeding of the International Conference on Computer Application and System Modeling (ICCASM), 2010: V4-664-V4-668.

Waleed, Y. and Y. Mohamed, 2010. Optimized asset planning for minimizing latency in wireless sensor networks. Wirel. Netw., 16(1): 65-78.