International Journal of Science and Engineering Applications
Volume 2 Issue 5, 2013, ISSN-2319-7560 (Online)

Failure analysis of Upright & Base for Shelving Rack

Pratik Vitonde
Department of Mechanical Engineering,
YCCE, Nagpur,
India.

G.H. Waghmare
Department of Mechanical Engineering,
YCCE, Nagpur,
India.

Abstract: Shelving system is used for the display & storage of Books, Pharmacy products, FMCG goods, electronic goods, canned goods, hardware etc. CAEM is shelving Engineering for the Retail Sector which manufactures modular metal shelving systems for the shop fitting industry. The main structural members of steel storage racks are uprights & Bases. CAEM India imports upright & bases from CAEM Italy for shelving racks. In this work, the commercial software, ANSYS, is used for material and geometric nonlinear analysis of the upright & base and the results are compared with analytical data obtained & providing a suitable design for the same.

Keywords: Analysis; Base; CAEM; Shelving; Upright

1. INTRODUCTION

CAEM is Shelving Engineering industry for the Retail Sector having more than fifty years of experience in the manufacturing of modular metal shelving systems. CAEM offers today the widest and most complete and quality oriented shelving system and display solutions for the shop fitting industry. These units are designed to have modularity according to the need of customer, also designed to have interchange-ability as per the requirements. The headquarters is in Italy and the company's direct presence is guaranteed in the UK, in Australia and in India.

CAEM offers display solutions for all retailers, namely: Hyper Markets, Supermarkets, Departmental Stores, Fashion Stores, Consumer Durables, IT and Home Entertainment stores, Specialty stores. They have over the period of last few decades, provided the shelving solutions to many reputed clients located in Europe, Australia, Middle-East, etc The high quality of products along with the continuous development of new & innovative design solutions gives a strong competitive advantage.[4]

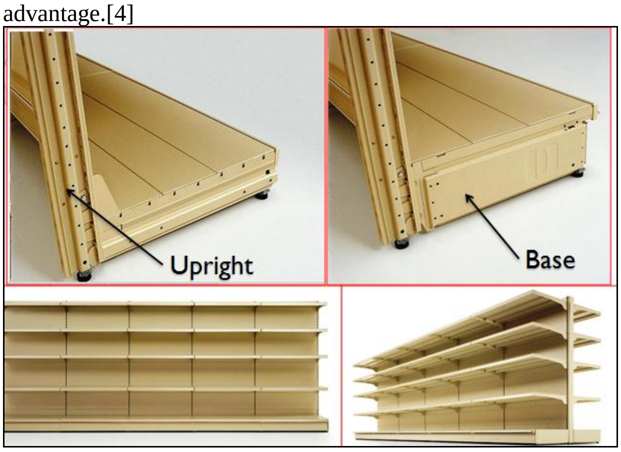

Fig.1 Shelving Rack TN9

CAEM India imports upright & bases from CAEM Italy for shelving rack system. As the items are imported they are very costly & also they are difficult to manufacture them in India because of their unique design, number of bends and the special certified high strength steel.

2. RELATED WORK

The various points like the dimension, codes and loads that should be specified in order to have a storage rack or shelving system that is economical and safe. The important dimensions to specify are length and width of shelf, and the height between shelves. From the size of the product and pallet, one can determine the depth of the upright or truss. The overhang is required when the shelf beams are not covered with say a sheet of particleboard and the pallet and load could fall between the beams. The upright depth of back to back of column would then be 48 inches minus a total overhang of 4 inches or 44 inches back-to-back of column or rack post. If back-to-back rows are being considered, then one must consider the clearance between loads at the back in order to position the uprights for back-to back rows are length and width of shelf, and the height between shelves. [1]

The product load or shelf load should be always specified. The shelf load will then be used by the shelving supplier to determine the reinforcement, if reinforcing is necessary. Most manufacturers have members such as bars and angles that reinforce the front, back, and side edges; as well as hat shaped sections that reinforce the centre of the shelf by spanning the length the shelf at the centre of the shelf. Failure analysis is an engineering approach to determining how and why equipment or a component has failed. Some general causes for failure are structural loading, wear, corrosion, and latent defects. The goal of a failure analysis is to understand the root cause of the failure so as to prevent similar failures in the future. In addition to verifying the failure mode it is important to determine the factors that explain the "how and why" of the failure event [2].

The main structural members of steel storage rack frames are uprights and pallet beams. While pallet beams brace uprights and provide stiffness against down-aisle buckling through semi-rigid connections to uprights, the most critical members of a rack structure are the uprights, which are usually made from cold-formed open sections. They are the members most affected by instability, including local buckling, distortional buckling. When braced, the spine bracing is accompanied with plan bracing, to also provide down-aisle bracing of the front row of uprights. The down-aisle buckling capacity is affected by the stiffness of the semi rigid connectors (joints) between uprights and pallet beams, and the semi-rigid stiffness of the base plate connection between the upright and floor, of which the latter is dependent on the axial force in the upright[3]. Performance of pallet racking systems depends upon the efficiency of beam-end connectors, which provide, together with column bases, sources of stiffness for down-aisle stability. Knowledge of the actual joint behaviour under static and seismic loading is of fundamental importance for a suitable definition of simplified moment rotation joint relationships to use into design of semi-continuous frames [4]. Rack systems are very similar to the framed steelworks traditionally used for civil and commercial buildings, but great differences in member geometry and in connection systems. In rack system, the beams are generally boxed cross-section, and columns are open thin walled perforated section to accept the tabs of beam end-connectors [5].

3. EXISTING UPRIGHT BASE ASSEMBLY

Initially a design is made as a solution for the TN9 Italian upright & base as shown in fig-1. In this design 80x40x2mm thick rectangular pipe section is used for both upright & base as shown in fig-2. The pipe is cut as per the required dimension for both of the upright & base. The slotting is done on the upright pipe in such a way that, the pitches of the slots made are compatible with the other assembly items such as backplanes, brackets, and loading bar etc.

A flange of size 75x75x3mm thick is welded on the top of the base as shown in the fig-2. The upright is inserted vertically into the flange. The outer dimensions of the flange are smaller than that of the inner dimension of the upright. Once the upright is inserted into the flange the assembly is tightened by the use of fasteners.

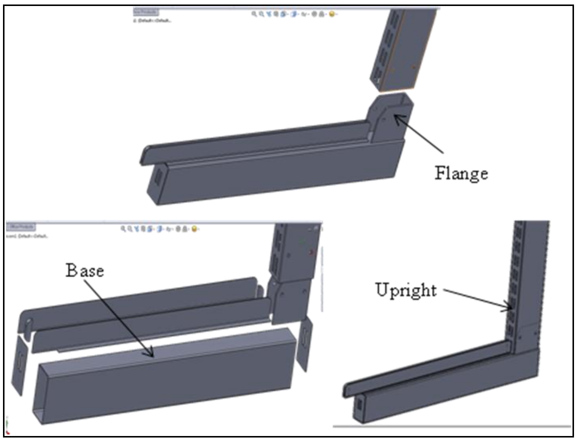

Fig.2 Present Design

Fig.3 Failure of upright & base.

The system is put under a specific loading condition for certain duration of time. A test is carried out on it after which it is found that the suggested design for upright & base is failed as shown in fig-3 on the specific loading condition. Initial reasons of failures for the suggested design is observed are as follows:

- The load acting on the upright is not going directly towards earth.
- The load acting on the upright is carried by the base.
- The height of base is used as 8cm instead of 16cm for more than 200cm ht. of upright.
- The thickness of pipe section for base & upright is used as 2mm instead of 2.5 to 3mm which normally used in other shelving system.

4. MODIFIED DESIGN

After identifying the mode of failure in the present design, a modified design is made for the same. In this design, the upright is of full length i.e. up to the ground and the base is inserted to the slot made on the upright. The design has an advantage that the load on the upright is not acting on the base as compare to the present design. The load is directly going to the earth and helps in balancing the overall load acting on the system. The idea is same that of the TN9 system which has similar type of arrangement.

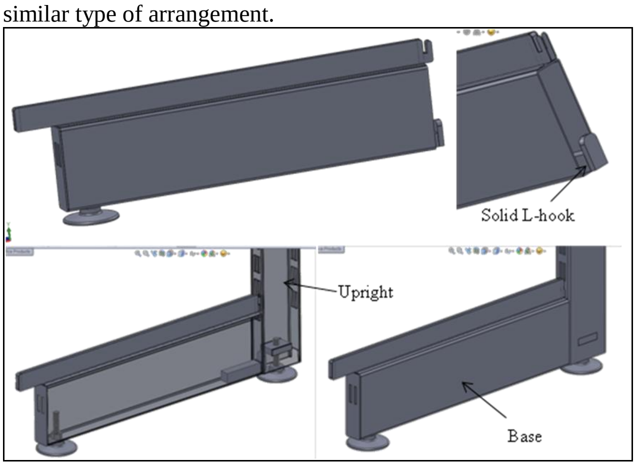

Fig.4 Modified Design

In this, a solid L-hook is welded on the back side bottom of the base pipe as shown in fig-4. The slots are made on the upright pipe in which the L-hook & top plate brackets are hook tightly. The system is completely modular and there is no need of fasteners to fix the upright base assembly. The rest of the assembly components are made similar to that of present design.

Following are the several advantages of the system & also the best suited design for the project because:

- The load acting on base & upright is carried by themselves separately.
- The load acting on the upright is going directly towards earth.
- In the welded option one has to maintain different combinations of uprights & basses. There are different heights of uprights available i.e. 88cm, 112cm, 120cm, 134cm etc around 25nos. of sizes. & different depths of

bases available i.e. 20cm, 30cm up to 80cm. around 8nos. of sizes.

- So if the welding option is used their will be 'n' number of combinations of upright & bases. In this case it is very difficult to maintain the separate stock of upright & bases combinations in warehouse which will directly leads to fatigue & increase is in inventory cost.
- In case of option one it is difficult to weld the flange perpendicular on the top of the upright. If the welding is not done properly the misalignment will be their & will be difficult for the assembly.
- Completely modular system.
- No need of fasteners.
- Simple assembly & saves time for the installation.

5. Material & Design for maximum load

Material: the Material for upright & base I.S. C-30 Soft Steel (SAE1030) material used is.

Table 1. Material and Their Properties

Material	S_{ut}	S_{yt}	S_{ye}	S_{eb}	E	G	BHN
I.S. C-25 Soft Steel (SAE 1025)	455	232	136	198	204	80	120
I.S. C-30 Soft Steel (SAE 1030)	527	296	183	225	204	79	150
I.S. C-30 Soft Steel (SAE 1035)	580	367	-	-	204	79	126

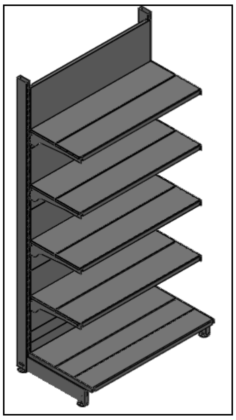

Fig.5 Assembly of designed unit

Loading Condition: - For the project work the upright height is selected as 208cm & base depth as 50cm.This combination is widely used in the supermarkets, big bazaar, hypermarkets etc. Fig-6 shows the detail structure for the system. The unit has 4 shelf of size L-40cm. A uniformly distributed load of 55kg is used for per shelves. 80x30x2.5mm thick rectangular pipe section is used for upright & 100x80x2.5mm thick section for base.

After doing little iteration on loading, the load 540N is found safer for the system as described below:

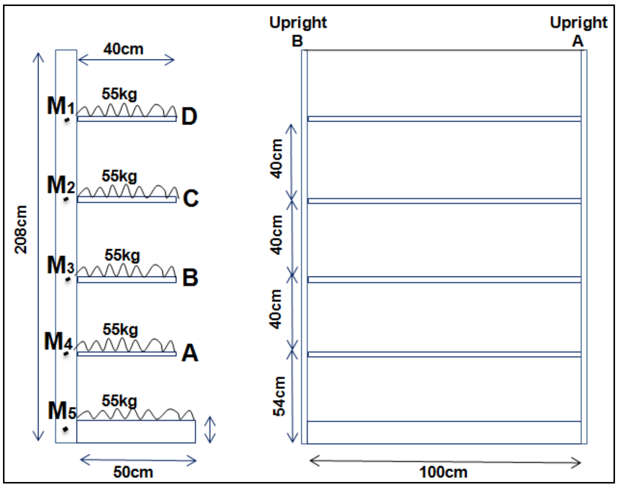

Fig.6 Loading condition

<u>Reaction at A,</u>

Ra= 540 x 0.4 Ra=216 N

<u>Shear Force</u>

A= Ra = 216 N

C= 216 N

B= 216 – (540x0.4) = 0

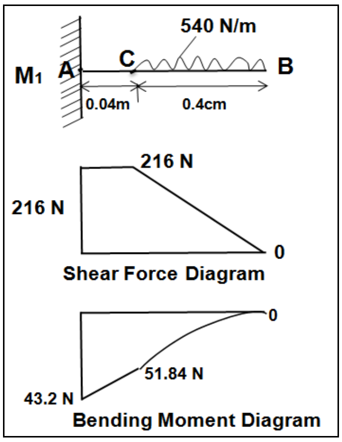

Fig.7 Shear force & Bending Moment Diagram

Bending Moment,

$\Sigma M_x = 0$

$M_b=0$

$M_c= -43.2$ N-m

$M_a= -51.84$ N-m

Now, $M_2= M_1+M_2 = -72.576$ N-m

$M_3= M_{1+}M_2+M_3 = -122.34$ N-m

$M_4= M_1+M_2+M_3 = -221.04$ N-m

$M_5 = M_1+M_2+M_3+M_4 = -420.36$ N-m

$$\frac{M}{I} = \frac{\sigma b}{Y} = \frac{E}{R}$$

$$\sigma b = \frac{M}{Z}$$

$\frac{I}{Y} = Z$ i.e. Section Modulus

For rectangular cross-section,

$$I = \frac{BD^3}{12} - \frac{bd^3}{12}$$

$$I = \frac{1}{12}(BD^3 - bd^3)$$

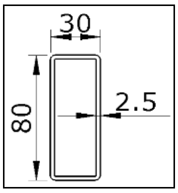

Fig.8 Cross-section of upright

$I = 82.34 \times 10^3$ mm^4

$$Y = \frac{D}{2}$$

Y=40mm

σb= 205 Mpa

Allowable Bending stress S_{eb}= 225 Mpa

σb= 205 Mpa < Allowable Bending stress Design is Safe

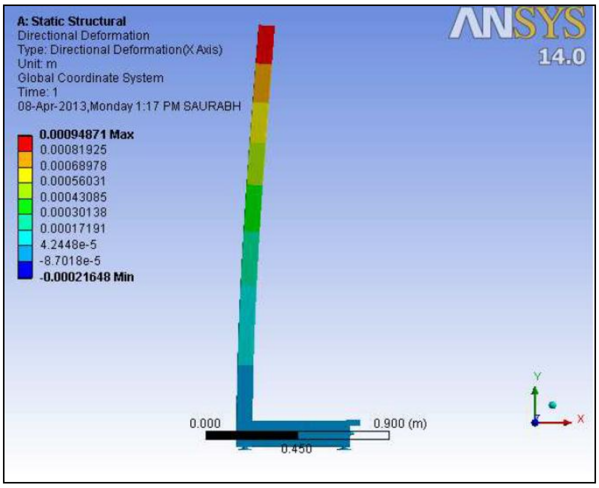

Fig.9 Structural analysis of upright base assembly at 490N

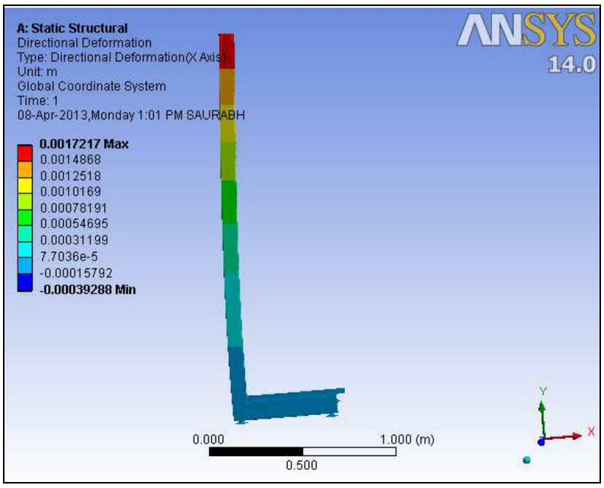

Fig.10 Structural analysis of upright base assembly at 270N

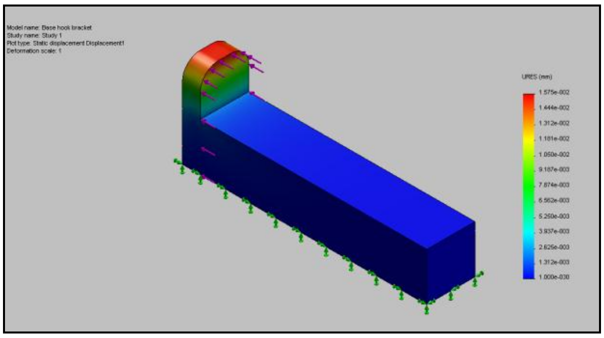

FIG.11 STRUCTURAL ANALYSIS OF BASE PLATE

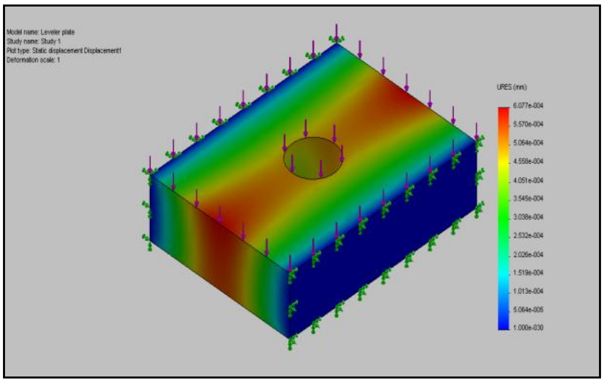

Fig.12 Structural analysis of Leveler plate

6. NOMENCLATURE

σb - Bending stress Mpa

I - Moment of inertia mm4

Z - Section modules mm3

M - Bending Moment N-mm

Y - Distance from neutral axis mm

R - Radius of curvature

S_{ut} - Ultimate tensile strength Mpa

S_{yt} - Yield strength in tensile Mpa

S_{ys} - Yield strength in shear Mpa

S_{eb} - Endurance limit in reversed bending Mpa

7. CONCLUSION

The main aim of the project is to carry the specified load by the upright & base assembly as long as possible. The existing arrangement is quite unable to carry the specified loading condition thus we modified the assembly. The new modified assembly for upright & base has given the better result than the existing one. The new suggested material & extra thickness for upright & base will improve its hardness and working efficiency at 540N and minimizes the bending & failure of the upright & base.

8. ACKNOWLEDGMENTS

I would like to give my sincere thanks to the whole CAEM group, who provided tremendous support, knowledge and encouragement and had the trust in my capability to the conduct the research of this nature.

9. REFERENCES

[1] W.R. Midgley, "Planning & Engineering Guidelines for Specifying Storage Racks and Shelving Systems".Principal Midgley-Clauer Associates, INC.4438 Lleffingwell Road Canfield, Ohio 44406.NA 92 Forum.

[2] TCR Engineering Services Technical Team, "Investigating Material & Component Failure", published July- 2004. K.J.R. Rasmussen & B.P. Gilbert, "Analysis-based design provisions for steel storage racks", School of Civil Engineering, University of Sydney, Sydney NSCC2009.

[3] K.J.R. Rasmussen & B.P. Gilbert, "Analysis-based design provisions for steel storage racks", School of Civil Engineering, University of Sydney, Sydney NSCC2009.

[4] Nadia Baldassino,Claudio Bernuzzi, Riccardo Zandonini, "PERFORMANCE OF JOINTS IN STEEL STORAGE PALLET RACKS", Department of Mechanical and Structural Engineering, University of Trento

[5] K.M. Bajoria, K.K. Sangle, "DYNAMIC ANALYSIS OF 2-D AND 3-D COLD FORMED STORAGE RACK STRUCTURES WITH RIGID AND SEMI RIGID CONNECTIONS" Associate Professor Department of Civil Engineering, Indian Institute of Technology Bombay, Powai, Mumbai– 400 076 (India). The 14th World Conference on Earthquake Engineering October 12-17- 2008, Beijing, China.

[6] B.D. Shiwalker, "Desigk data book for machine elements" (S.I. Units) 22nd Jun 2008.

[7] www.caem.net,www.youtube.com/wwwcaemnet,www.facebook.com/caem.net

International Journal of Science and Engineering Applications
Volume 2 Issue 5, 2013, ISSN-2319-7560 (Online)

Agent-Driven Distributed Data Mining

Rohini . P
Department of CSE
Malla Reddy College of Engineering
Andhra Pradesh, India.

Sree Lakshmi.P
Department of CSE
Malla Reddy College of Engineering
Andhra Pradesh, India.

Abstract: Multi-Agent systems (Autonomous agents or agents) and knowledge discovery (or data mining) are two active areas in information technology. A profound insight of bringing these two communities together has unveiled a tremendous potential for new opportunities and wider applications through the synergy of agents and data mining. Multi-agent systems (MAS) often deal with complex applications that require distributed problem solving. In many applications the individual and collective behavior of the agents depends on the observed data from distributed data sources. Data mining technology has emerged, for identifying patterns and trends from large quantities of data. The increasing demand to scale up to massive data sets inherently distributed over a network with limited band width and computational resources available motivated the development of distributed data mining (DDM).Distributed data mining is originated from the need of mining over decentralized data sources. DDM is expected to perform partial analysis of data at individual sites and then to send the outcome as partial result to other sites where it sometimes required to be aggregated to the global result.

Keywords: Distributed Data Mining, Multi-Agent Systems, Multi Agent Data Mining, Multi-Agent Based Distributed Data Mining.

1. INTRODUCTION

Data Mining (DM), originated from knowledge discovery from databases (KDD), the large variety of DM techniques which have been developed over the past decade includes methods for pattern-based similarity search, cluster analysis, decision-tree based classification, generalization taking the data cube or attribute-oriented induction approach, and mining of association rules [7]. DDM is a branch of the field of data mining that offers a framework to distributed data paying careful attention to the distributed data and computing resources. Distributed data mining (DDM) mines data from data sources regardless of their physical locations. The need for such characteristic arises from the fact that data produced locally at each site may not often be transferred across the network due to the Excessive amount of data and security issues. Recently, DDM has become a critical component of knowledge based systems because its decentralized architecture reaches every network such as weather databases, financial data portals, or emerging disease information systems has been recognized by industrial companies as an opportunity of major revenues from applications such as warehousing, process control, and customer services, where large amounts of data are stored.

In the DDM literature, one of two assumptions is commonly adopted as to how data is distributed across sites: homogeneously (horizontally partitioned) and heterogeneously (vertically partitioned). Both viewpoints adopt the conceptual viewpoint that the data tables at each site are partitions of a single global table.

Data Mining still poses many challenges to the research community. The main challenges in data mining are: 1) Data mining has to deal with huge amounts of data located at different physical locations. 2) Data mining is computationally intensive process involving very large data i.e. more than terabytes. So, it is necessary to partition and distribute the data for parallel processing to achieve acceptable time and space performance. 3) The data stored for particular domain the Input data changes rapidly. In these cases, knowledge has to be mined fast and efficiently in order to be usable and updated.

2. NEED OF MULTI-AGENTS

Autonomous agents are computational systems that inhibit some complex dynamic environment, sense and act autonomously in this environment, and by doing so realize a set of goals or tasks for which they are designed. Agents are reactive i.e., they perceive their environment and respond in a timely fashion to changes that occur.

Multi-Agent systems are used for all types of system composed of multiple autonomous components showing the following characteristics:

- each agent has incomplete capabilities to solve a problem
- there is no global system control
- data is decentralized
- computation is asynchronous

Multi-Agent has the following features

- Dividing functionality among many agents provides modularity, flexibility, modifiability, and extensibility.
- Knowledge that is spread over various sources

(agents) can be integrated for a more complete view when needed.

- Applications requiring distributed computing are better supported by MAS.
- Agent technology supports distributed component technology.

In a typical distributed environment analyzing distributed data is a non-trivial problem because of many constraints such as limited bandwidth (e.g. wireless networks), privacy sensitive data, distributed compute nodes, only to mention a few. The field of Distributed Data Mining (DDM) deals with these challenges in analyzing distributed data and offers many algorithmic solutions to perform different data analysis and mining operations in a fundamentally distributed manner that pays careful attention to the resource constraints.

Since MAS are also distributed systems, combining DDM with MAS for data intensive applications is appealing. DDM is expected to perform partial analysis of data at individual sites and then to send the outcome as partial result to other sites where it is sometimes required to be aggregated to the global result. Quite a number of DDM solutions are available using various techniques such as distributed association rules, distributed clustering, Bayesian learning, classification (regression), and compression, but only a few of them make use of intelligent agents at all. The main problems any approach to DDM is challenged issues of autonomy and privacy. For example, when data can be viewed at the data warehouse from many different perspectives and at different levels of abstraction, it may threaten the goal of protecting individual data and guarding against invasion of privacy. These issues of privacy and autonomy become particularly important in business application scenarios where, for example, different (often competing) companies may want to collaborate for fraud detection but without sharing their individual customers' data or disclosing it to third parties. DDM is a complex system focusing on the distribution of data resources over the network as well as extraction of data from those resources. The very core of DDM systems is the scalability as the system configuration may be altered time to time, therefore designing DDM systems deals with great details of software engineer issues, such reusability, extensibility, compatibility, flexibility and robustness. For these reasons, agents' characteristics are desirable for DDM systems.

Autonomy of the system: A DM agent here is considered as a modular extension of a data management system to deliberatively handle the access to the data source in agreement with constraints on the required autonomy of the system, data and model. This is in full compliance with the paradigm of cooperative information systems [6].

Multi-strategy DDM: For some complex application settings an appropriate combination of multiple data mining techniques may be more beneficial than applying just one particular one. DM agents may choose depending on the type of data retrieved from different sites and mining tasks to be done. The learning of multi-strategy selection of DM methods is similar to the adaptive selection of coordination strategies in a multi-agent system as proposed.

Collaborative DM: DM agents may operate independently on data they have gathered at local repositories, and then combine their respective patterns or they may agree to share potential knowledge as it is discovered.

Scalability of DM to massive distributed data: To reduce network and DM application server load, DM agents migrate to each of the local data sites in a DDM system on which they may perform mining tasks locally, and then either return with or send relevant pre-selected patterns to their originating server for further processing. Experiments in using mobile information filtering agents in distributed data environments are encouraging [9].

Security: Any agent-based DDM system has to cope with the problem of ensuring data security and privacy. Any mining operation performed by agents of a DDM system lacking sound security architecture could be subject to eavesdropping, data tampering, or denial of service attacks. Agent code and data integrity is a crucial issue in secure DDM: Subverting or hijacking a DM agent places a trusted piece of (mobile) software. In addition, data integration or aggregation in a DDM process introduces concern regarding inference attacks as a potential security threat. However, any failure to implement least privilege at a data source, that means endowing subjects with only enough permissions to discharge their duties, could give any mining agent unsolicited access to sensitive data. Finally, selective agent replications may help to prevent malicious hosts from simply blocking or destroying the temporarily residing DM agents.

Trustworthiness: Data mining agents may infer sensitive information even from partial integration to a certain extent and with some probability. This problem, known as the so called inference problem, occurs especially in settings where agents may access data sources across trust boundaries which enable them to integrate implicit knowledge from different sources using commonly held rules of thumb.

Furthermore, the decentralization property seems to fit best with the DDM requirement in order to avoid security treats. At each data repository, mining strategy is deployed specifically for the certain domain of data.

3. OPEN PROBLEMS STRATEGY

Several systems have been developed for distributed data mining. These systems can be classified according to their strategy to three types; central learning, meta-learning, and hybrid learning.

3.1 Central learning strategy is when all the data can be gathered at a central site and a single model can be build. The only requirement is to be able to move the data to a central location in order to merge them and then apply sequential DM algorithms. This strategy is used when the geographically distributed data is small. The strategy is generally very expensive but also more accurate [10]. The process of gathering data in general is not simply a merging step; it depends on the original distribution. Agent technology is not very preferred in such strategy.

3.2 Meta-learning strategy offers a way to mine classifiers from homogeneously distributed data. Meta-learning follows three main steps. 1) To generate base classifiers at each site using a classifier learning algorithms. 2) To collect the base classifiers at a central site, and produce meta-level data from a separate validation set and predictions generated by the base classifier on it. 3) To generate the final classifier (meta-classifier) from meta-level data via a combiner or an arbiter. Copies of classifier agent will exist or deployed on nodes in the network being used. Perhaps the most mature systems of agent-based meta-learning systems are: JAM system [11], and BODHI [11].

3.3 Hybrid learning strategy is a technique that combines local and centralized learning for model building [15]; for example, Papyrus [12] is designed to support both learning strategies. In contrast to JAM and BODHI, Papyrus can not only move models from site to site, but can also move data when that strategy is desired. Papyrus is a specialized system which is designed for clusters while JAM and BODHI are designed for data classification. The major criticism of such systems is that it is not always possible to obtain an exact final result, i.e. the global knowledge model obtained may be different from the one obtained by applying the one model approach (if possible) to the same data. Approximated results are not always a major concern, but it is important to be aware of that. Moreover, in these systems hardware resource usage is not optimized. If the heavy computational part is always executed locally to data, when the same data is accessed concurrently, the benefits coming from the distributed environment might vanish due to the possible strong performance degradation. Another drawback is that occasionally, these models are induced from databases that have different schemas and hence are incompatible.

Autonomous agent can be treated as a computing unit that performs multiple tasks based on a dynamic configuration. The agent interprets the configuration and generates an execution plan to complete multiple tasks. [7], [14], [8], [6], and [9] discuss the benefits of deploying agents in DDM systems. Nature of MAS is decentralization and therefore each agent has only limited view to the system. The limitation somehow allows better security as agents do not need to observe other irrelevant surroundings. Agents, in this way, can be programmed as compact as possible, in which light-weight agents can be transmitted across the network rather than the data which can be more bulky. Being able to transmit agents from one to another host allows dynamic organization of the system. For example, mining agent *ma*, located at repository r1, posses algorithm *alg1*. Data mining task t1 at repository *r*2 is instructed to mine the data using a*lg*1. In this setting, transmitting *alg*1 to *r*2 is a probable way rather than transfer all data from *r*2 to r1 where *alg*1 is available.

4. AGENT-BASED DISTRIBUTED DATA MINING (ADDM)

ADDM is a novel data mining technique that inherits all powerful properties of agents and, as a result, yields desirable characteristics. In general, constructing an ADDM system concerns with three key characteristics: interoperability, dynamic system configuration, and performance. Interoperability concerns with collaboration of agents in the system, and external interaction which allow new agents to enter the system seamlessly. The architecture of the system must be open and flexible so that it can support the interaction including communication protocol, integration policy, and service directory. Communication protocol covers message encoding, encryption, and transportation between agents. Integration policy specifies how a system behaves when an external component, such as an agent or a data site, requests to enter or leave. Dynamic system configuration, handles the dynamic configuration of the system, and is a challenge issue due to the complexity of the planning and mining algorithms. A mining task may involve several agents and data sources, in which agents are configured to equip with an algorithm and deal with given data sets. In distributed environment, tasks can be executed in parallel, in exchange, concurrency issues arise. Quality of service control in performance of data mining and system perspectives is desired; however it can be derived from both data mining and agent's fields.

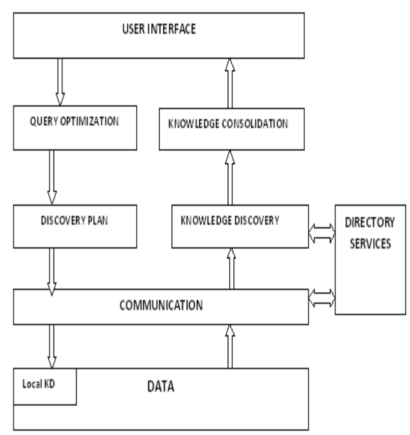

Fig.4.1: ADDM System Architecture

An ADDM system can be generalized into a set of components and viewed as depicted in figure 4.1.We may generalize activities of the system into request and response, each of which involves a different set of components. Basic components of an ADDM system are as follows.

Data: Data is the foundation layer of the architecture. In distributed environment, data can be hosted in various forms, such as online relational databases, data stream, web pages, etc., in which purpose of the data might be varied.
Communication: The system chooses the related resources from the directory service, which maintains a list of data sources, mining algorithms, data schemas, data types, etc. The communication protocols may vary depending on implementation of the system, such as client-server, peer-

to-peer etc.

User Interface: The user interface (UI) interacts with the user as to receive and respond to the user. The interface simplifies complex distributed systems into user-friendly message such as network diagrams, visual reporting tools, etc. On the other hand, when a user requests for data mining through the UI, the following components are involved.

Query optimization: A query optimizer analyses the request as to determine type of mining tasks and chooses proper resources for the request. It also determines whether it is possible to parallelize the tasks, since the data is distributed and can be mined in parallel.

Discovery Plan: A planner allocates sub-tasks with related resources. At this stage, mediating agents play important roles as to coordinate multiple computing units since mining sub-tasks performed asynchronously as well as results from those tasks. On the other hand, when a mining task is done, the following components are taken place,

Local Knowledge Discovery (KD): In order to transform data into patterns which adequately represent the data and reasonable to be transferred over the network, at each data site, mining process may take place locally depending on the individual implementation.

Knowledge Discovery: Also known as mining, it executes the algorithm as required by the task to obtain knowledge from the specified data source.

Knowledge Consolidation: In order to present to the user with a compact and Meaningful mining result, it is necessary to normalize the knowledge obtained from various sources. The component involves complex methodologies to combine knowledge/ patterns from distributed sites. Consolidating homogeneous knowledge/patterns is promising and yet difficult for heterogeneous case.

5. PROPOSED SCHEMA

Building and managing of large-scale distributed systems is becoming an increasingly challenging task. Continuous intervention by user administrators is generally limited in large-scale distributed environments. System support is also needed for configuration and reorganization when systems evolve with the addition of new resources. The primary goal of the management of distributed systems is to ensure efficient use of resources and provide timely service to users. Most of the distributed system management techniques still follow the centralized model that is based on the client-server model. Centralization have presented some problems, such as: 1) it could cause a traffic overload and processing at the manager node may affect its performance;2) it does not present scalability in the increase of the complexity of the network; 3) the fault in the central manager node can leave the system without a manager.

One model is the distributed management where management tasks are spread across the managed infrastructure and are carried out at managed resources. The goal is to minimize the network traffic related to management and to speed up management tasks by distributing operations across resources. The new trend in distributed system management involves using multi-agents to manage the resources of distributed systems. Agents have the capability to autonomously travel (execution state and code) among different data repositories to complete their task. The route may be predetermined or chosen dynamically depending on the results at each local data repository.

The concept of multi-agents promises new ways of designing applications that better use the resources and services of computer systems and networks. For example, moving a program (e.g., search engine) to a resource (e.g., database) can save a lot of bandwidth and can be an enabling factor for applications that otherwise would not be practical due to network latency.

Conceptually, a multi-agent can migrate its whole virtual machine from host to host; it owns the code, not the resources. Multi-agents are the basis of an emerging technology that promises to make it very much easier to design, implement, and maintain distributed systems. We have found that multi-agents reduce network traffic, provide an effective means of overcoming network latency, and, perhaps most importantly, through their ability to operate asynchronously and autonomously of the process that created them help us to construct more robust and fault tolerant systems. The purpose of the proposed multi-agent system is to locate, monitor, and manage resources in distributed systems. The system consists of a set of static and mobile agents. Some of them reside in each node or element in the distributed system. There are two multi-agents named delegated and collector agents that can move through the distributed system. The role of each agent in the multi-agent system, the interaction between agents, and the operation of the system

5.1 Structure of Multi-Agent Systems

The multi-agent system structure assumes that each node in the system will have a set of agents residing and running on that node [1]. These agent types are the following:

Client agent (CA) percepts service requests, initiated by the user, from the system. The CA may receive the request from the local user directly. In the other case, it will receive the request from the exporter agent coming from another node.

Service list agent (SLA) has a list of the resource agents in the system. This agent will receive the request from the CA and send it to the resource availability agent. If the reply indicates that the requested resource is local then the service list agent will deliver the request to the categorizer agent. Otherwise, it will return the request to the CA.

Resource availability agent (RAA) indicates whether the requested resource is free and available for use or not. It also indicates whether the requested resource is local or remote. It receives the request from the service list agent and checks the status of the requested resource through the access of the MIB. The agent then constructs the reply depending on the retrieved information from the database.

Resource agent (RSA) is responsible for the operation and control of the resource. This agent executes the on the resource. Each node may have zero or more RSAs.

Router agent (RA) provides the path of the requested resource on the network in case of accessing remote resources. Before being dispatched, the exporter agent will ask the router agent for the path of the requested resource. This in turn delivers it to the exporter agent.

Categorizer agent (CZA) allocates a suitable resource agent to perform the user request. This agent percepts inputs coming from the service list agent. It then tries to find a suitable free resource agent to perform the requested service.

Exporter agent (EA) is a mobile agent that can carry the user request through the path identified by the RA to reach the node that has the required resource. It passes the requested resource *id* to the RA and then receives the reply. If the router agent has no information about the requested resource, the EA will try to locate the resource in the system. There are also two additional mobile agent types exist in the system.

Delegated agent (DA) is a mobile agent that is launched in each sub network. It is responsible for traversing sub network nodes instead of theexporter agent to do the required task and carry results back to the exporter agent.

Collector agent (CTA) is a mobile agent that is launched from the last sub network visited by the exporter agent. It is launched when results from that sub network become available. This agent goes through the reversed itinerary of the exporter agent trip. The CTA collects results from the delegated agents and carries it to the source node. All mobile agents used here are of interrupt driven type.

5.2 Functionality of the System

The activity cycle of the multi-agent system resides in a local data repository. The client agent receives the service requests either from the user or from an exporter agent. The client agent then asks the service list agent for the existence of a resource agent that can perform the request. The service list agent checks the availability of the required resource agent by consulting a resource availability agent to perform the requested service. The reply of the resource availability agent describes whether or not the resource is locally available and whether or not there is a resource agent that can perform the requested service. If the resource availability agent accepts the request, the service list agent will ask the categorizer agent to allocate a suitable resource agent to the requested service and the resource agent will perform the requested service. Otherwise, the service list agent informs the client agent with the rejection and is passed to the exporter agent. The exporter agent asks the router agent for the path of the required resource agent. Once the path is determined, the exporter agent will be dispatched through the network channel to the destination node identified by that path. If the router agent has no information about the location of the required resource agent, the exporter agent will search the distributed system to find the location of the required resource agent and assign the required task to it.

As shown in Fig. 5.2.2, the exporter agent traverses the sub networks of the distributed system through its trip. At each sub network, a delegated agent is launched to traverse the local nodes of that sub network doing the required task and carrying results of that task.

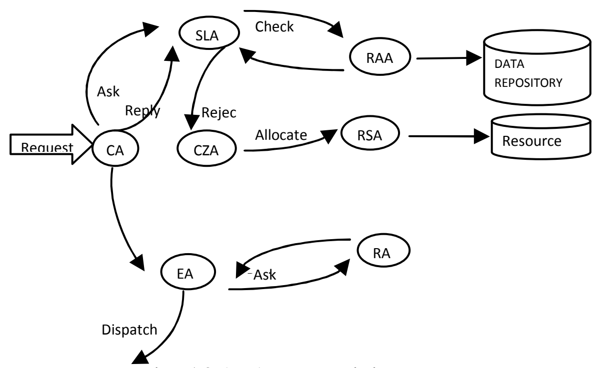

Fig. 5.2.1: Agents activity.

The agents of the social interface described in Fig. 5.2.1 are implemented at each node in the system. There are two approaches to collect results of the required task and send these results back to the source.

In traditional agent-based management systems that use mobile agent, the exporter agent will wait at each visited sub network until the delegated agent finishes its work and obtains results. Then, the exporter agent will take these results and go to the next sub network in its itinerary. The exporter agent will return to its home sub network after visiting all the sub networks determined in the itinerary. The home sub network of the exporter agent is the sub network from which it was initially dispatched. The waiting of the exporter agent prevents execution of tasks to be started in the other sub networks. This approach is used in most of the previously developed management systems in which operation is based on mobile agents. In the proposed multi-agent management system, the exporter agent does not wait for results from each sub network. It resumes its trip visiting other sub networks, and at each sub network, another delegated mobile agent is launched to carry out management tasks instead of the exporter agent. The exporter agent will be killed at the last visited sub network in its itinerary. When results from the last visited sub network become available, another mobile agent called collector agent is launched or dispatched from this sub network to collect results from it and other sub networks. The collector agent goes through the reversed itinerary of the exporter agent trip carrying results to the home sub network. In this manner, operations can be done in a parallel fashion at different sub networks because there is no delay of the task submission to local data repositories of these sub networks.

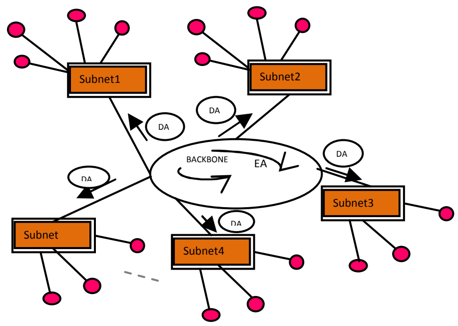

Fig. 5.2.2: Network architecture of ADDM.

6. CONCLUSION

Distributed management for distributed systems is becoming a reality due to the rapid growing trend in internetworking and the rapid expanding connectivity. This paper describes a new multi-agent system for the management of distributed systems. The system is proposed to optimize the execution of management functions in distributed systems. The proposed system can locate, monitor, and manage resources in the system. The new technique in that system allows management tasks to be submitted to sub networks of the distributed system and executed in a parallel fashion. The proposed system uses two multi- agents. The first is used to submit tasks to the sub networks of the distributed system and the other collects results from these sub networks. The proposed system is compared against traditional management techniques in terms of response time, speedup, and efficiency. A prototype has been implemented using performance management as the case study. The performance results indicate a significant improvement in response time, speedup, efficiency, and scalability compared to traditional techniques. The use of JVM in the implementation of the proposed system gives the system a certain type of portability. Therefore, it is desirable to use the proposed system in the management of distributed systems. The proposed system is limited to be applied to high-speed networks that have bandwidth 100 Mb/s or more. Also, the system cannot work when a failure occurs.

Future research will be related to the security of mobile agents and of hosts that receive them in the context of public networks. Mobile agents should be protected against potentially malicious hosts. The hosts should also be protected against malicious actions that may be performed by the mobile code they receive and execute. So, a detailed design and implementation of the whole secure system should be considered as a future work. Also, the high complexity of distributed systems could increase the potential for system faults. Most of the existing management systems assume that there is no fault in the system. It would be interesting to develop a fault tolerant management system that introduces safety in the system and attempts to maximize the system reliability without extra hardware cost.

7. REFERENCES:

[1] T.C. Du, E.Y. Li, and A. Chang,—Mobile agents in distributed network management, Commun. ACM, vol. 46, no. 7, pp. 127–132, July 2003.

[2] H. Ku, G.W.R. Ludere, and B.Subbiah, —An intelligent mobile agent framework for distributed network management,in Proc. Globecom'97Phoenix, pp. 160–164.

[3] N. R. Jennings and S. Bussmann, —Agent-based control systems—Why are they suited to engineering complex systems? IEEE Control Syst.Mag., vol. 23, no. 3, pp. 61–73, Jun. 2003.

[4] R. B. Patel, Neeraj Goel, —Mobile Agents in Heterogeneous Networks: A Look on Performance, Journal of Computer Science,2(11): 824-834, 2006.

[5] O'Hare G.M.P., Marsh D., Ruzzelli A., R. Tynan,—Agents for Wireless Sensor Network Power Management, in Proceedings of International Workshop on Wireless and Sensor Networks (WSNET-05), Oslo, Norway IEEE Press, 2005.

[6] S. Bailey, R. Grossman, H. Sivakumar, and A. Turinsky. Papyrus: a system for data mining over local and wide area clusters and super-clusters. In Supercomputing '99: Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM), page 63, New York, NY, USA, 1999. ACM

[7] R. J. Bayardo, W. Bohrer, R. Brice, A.Cichocki, J. Fowler, A. Helal, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, and Others.InfoSleuth: agent-based semantic integration of information in open and dynamic environments. ACM SIGMOD Record, 26(2):195–206, 1997.

[8] F. Bergenti, M.P. Gleizes, and F.Zambonelli. Methodologies And Software Engineering For Agent Systems: The Agent oriented Software Engineering Handbook. Kluwer Academic Publishers, 2004.

[9] R. Bose and V. Sugumaran. IDM: an intelligent software agent based data mining environment. 1998 IEEE International Conference on Systems, Man, and Cybernetics, 3, 1998.

[10] J. Dasilva, C. Giannella, R. Bhargava, H.Kargupta, and M. Klusch. Distributed datamining and agents. Engineering Applicationsof Artificial Intelligence, 18(7):791–807,

October 2005.

[11] S. Datta, K. Bhaduri, C. Giannella, R. Wolff,and H. Kargupta. Distributed data mining in peer-to-peer networks. Internet Computing,IEEE, 10(4):18–26, 2006.

[12] U. Fayyad, R. Uthurusamy, and Others. Data mining and knowledge discovery in databases.
Communications of the ACM,39(11):24–26, 1996.

[13] Vladimir Gorodetsky, Oleg Karsaev, and Vladimir Samoilov. Multi-agent technology for distributed data mining and classification.In IAT, pages 438–441. IEEE Computer
Society, 2003.

[14] Sven A. Brueckner H. Van Dyke Parunak. Engineering swarming systems.Methodologies and Software Engineering for Agent Systems, pages 341–376, 2004.

[15] W. Davies and P. Edwards. Distributed Learning: An Agent-Based Approach to Data-Mining. In *Proceedings of MachineLearning 95 Workshop on Agents that Learn from Other Agents*, 1995.

International Journal of Science and Engineering Applications
Volume 2 Issue 5, 2013, ISSN-2319-7560 (Online)

Improvement In LEACH Protocol By Electing Master Cluster Heads To Enhance The Network Lifetime In WSN

Kanwalnainpreet Kaur
Lovely Professional University
Punjab, India

Deepika Sharma
Lovely Professional University
Punjab, India

Abstract: In wireless sensor networks, sensor nodes play the most prominent role. These sensor nodes are mainly un-chargeable, so it raises an issue regarding lifetime of the network. Mainly sensor nodes collect data and transmit it to the Base Station. So, most of the energy is consumed in the communication process between sensor nodes and the Base Station. In this paper, we present an improvement on LEACH protocol to enhance the network lifetime. Our goal is to reduce the transmissions between cluster heads and the sink node. We will choose optimum number of Master Cluster Heads from variation cluster heads present in the network. The simulation results show that our proposed algorithm enhances the network lifetime as compare to the LEACH protocol.

Keywords: Cluster, Cluster-head, Network Lifetime, sink node.

1. INTRODUCTION

In recent years, wireless sensor network (WSN) has achieved a great attention of the researchers. The network comprises of number of sensor nodes which are deployed according to required application. Sensor nodes mainly perform the three basic tasks namely, sensing, processing and transmitting. There are numerous applications of WSNs exists in number of directions which includes environmental applications, medical monitoring, security for homes, surveillance, inventory management industrial and manufacturing automation, process control, , distributed robotics, etc. Mainly the wireless sensor network is situated at any unreachable track. Once the sensor network has been established at particular location it may not be changed easily. The energy of sensor nodes is going to absorb when it collects and send the information. So the energy consumption by the sensor network has become the most attractive issue for the researchers [1].

All the sensor nodes are allowed to communicate through a wireless medium. Figure 1. describes the architecture of a senor node. The wireless medium may either of radio frequencies, infrared or any other medium, of course, having no wired connection.

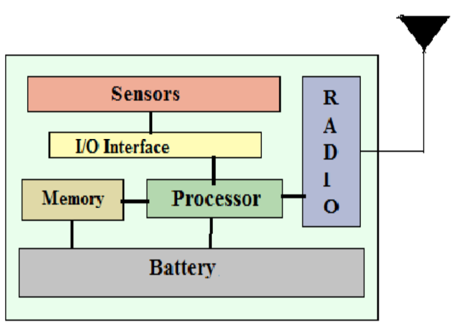

Figure 1. Architecture of sensor node

These nodes are deployed in a random fashion and they can communicate among themselves to make an ad-hoc network. WSN gives flexibility of adding nodes and removing the nodes as required. But this gives rise to many drastic changes to deal with in the network topology such as updating the path, or the network tree, etc. In a WSN the node that gathers the information refers to sink. The sink may be connected to the outside world through internet where the information can be utilized within time constraints [2].

The numbers of clustering algorithms have been proposed to improve the lifetime of the sensor network. In clustering, the sensor network is divided into clusters and then the one node from each cluster is selected as the cluster head. All the data aggregation activity has been done within the cluster and then cluster head use to send the information of a particular cluster to the BS which is also known as sink node. Clustering provides a reduction of redundancy and improvement over the lifetime of the wireless sensor network. LEACH is considered to be the most general clustering algorithm. In our proposed algorithm, an improvement over LEACH has been introduced. The simulation results show the improvement in the lifetime of the sensor network.

This paper is organized in the following manner. Section 2 is about literature survey regarding the variation clustering protocols of wireless sensor network. Section 3 will describes the block diagram representation of previous and new technique. Section 4 contains simulation results and the comparison of the new technique with the LEACH protocol. Finally, section 5 illustrates the conclusion for the work done for improving the lifetime of the network.

2. LITERATURE SURVEY

In this section, we present a brief study of some common clustering routing protocols for WSNs.

2.1 Low Energy Adaptive Clustering Hierarchy Protocol for WSNs (LEACH)

LEACH [3] is known as a distributed hierarchical protocol. It provides the aggregation for data in wireless sensor networks by selection of Cluster heads in random manner. This protocol first judges the strength of the received message or signal and then formation of cluster takes place. In this Cluster Head nodes are taken as routers to reach the sink node. Every non-Cluster Head node sends its data to their CHs. Before sensing received information to sink, CHs aggregate the information.

A stochastic algorithm is referred in each round by every single sensor node to determine whether it can be a cluster head for that particular round or may not act as a cluster head for that round. All normal nodes of the cluster communicate with CH in TDMA fashion which is scheduled by CH. LEACH clustering is shown in Figure 2.

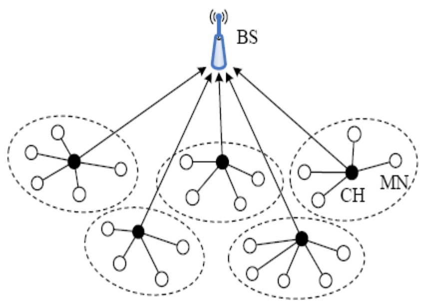

Figure 2. LEACH clustering

The operation of LEACH is conducted in numerous rounds, and each round is separated into two phases known as the set-up phase and the steady-state phase. In the set-up phase the various clusters of sensor network are organized, while in the steady-state phase information is delivered to the sink node. During the set-up phase, each sensor node decides whether or not to act as a cluster head for that particular round. This decision is made by the sensor node by randomly selecting a number between 0 and 1. A particular node becomes a cluster head for the current round if the number is less than the given following threshold value:

$$T(n) = \begin{cases} \overline{\quad\quad\quad} & \text{if } n \in G \\ 0 & \text{otherwise} \end{cases}$$

where P denotes desired percentage of cluster heads, r for current round, and G represents set of nodes that have not been elected CHs in the last 1/P rounds. When a sensor node is chosen as CH successfully, then it broadcasts an advertisement text to the other nodes of the network. By measuring the strength of received signal of the advertisement text, other sensor nodes decide to which cluster it will connect for this particular round and send a membership message to its CH.

Another phase is steady-state phase. During this phase the nodes sense and transmit data to the cluster heads. The CHs aggregate the information/data arriving from sensor nodes that belong to the respective cluster, and send aggregated information to the sink node directly.

2.2 Hybrid, Energy-Efficient, Distributed Clustering Approach (HEED)

HEED [4] is also a distributed clustering algorithm used for Wireless Sensor Networks. Every sensor node has some amount of energy associated with it. This energy of nodes reduces during reception and transmission of data. It also access query requests coming from the Base Station. HEED protocol follows to circulate the role of server among all nodes of the cluster so that a balance will be maintained between residual energy of all nodes of the cluster. Hence, remaining energy of cluster head would not drop to minimum leading to less node failures due to energy depletion in the network.

2.3 Power-Efficient Gathering in Sensor Information Systems (PEGASIS)

In this protocol [5], some chains consisting of different sensor nodes have been formed. Every node sends its data to the neighbor sensor node and most appropriate node is selected to transmit the data to the sink node. PEGASIS does not follow the concept of cluster formation and it prefer to decide or choose only one node from the chain to transmit to the sink node instead of using multiple sensor nodes present in the network. When a sensor node fails due to low battery backup, again the chain is made using the same previous greedy approach.

2.4 LEACH-C protocol

LEACH-C [3] is a centralized clustering algorithm in which sink node has the power to select the clusters based upon the annealing algorithm to find k optimal number of clusters. Here the Base Station selects the cluster heads for a particular round. The protocol guarantees an optimum number of clusters but it has a drawback that each sensor node provides information about its current position and remaining energy to the sink node during the set up phase which results in an extra wastage or overhead in the network.

2.5 LEACH-B

LEACH-B [6] is an improvement in Leach protocol for wireless senor network. In order to minimize the energy consumption and to prolong the life span of the sensor network, the protocol needs to ensure that the partition of cluster should be balance and uniform. To achieve this goal, the number of CHs needs to be dominated, and the network needs an optimal CHs amount. At each round, after first selection of cluster head according to LEACH protocol, a second selection is introduced to modify the number of cluster head in consideration of node's residual energy.

3. BLOCK DIAGRAM OF PREVIOUS AND PROPOSED TECHNIQUE

3.1 Previous technique- LEACH

Figure 3. illustrates the basic block diagram of LEACH algorithm. It gives a representation of step by step operation of LEACH. The working initiates by deployment of senor nodes in a random manner in network. A traditional clustering scheme is applied to divide the sensor network into clusters. Then cluster heads are chosen randomly and transmit the data to Base station. This process continuous in every round until the whole sensor nodes die or Base station commands for stopping the process.

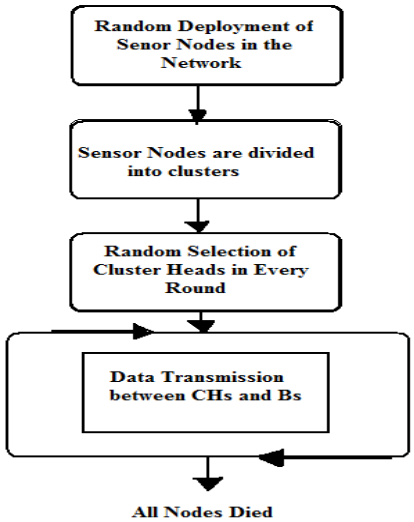

Figure 3. Basic Block Diagram for LEACH

3.2 New Proposed Technique

Figure 4. shows the basic block diagram for proposed technique. Every block marks the difference between the previous and new technique. Here, instead of transmission between every CH and sink node, it follows the technique of electing the Master Cluster Head for transmission of data from different cluster heads to the Base Station.

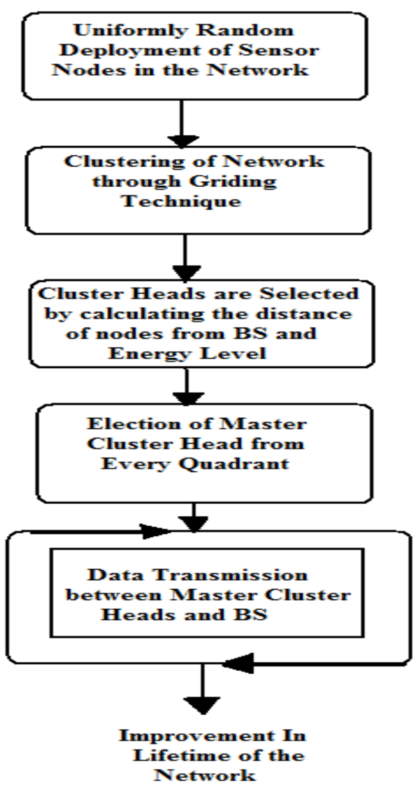

Figure 4.Basic Block Diagram for Proposed Technique

Thus our technique consists of following main stages:

- Random deployment of the nodes
- Formation of clusters using Gridding
- Selection of cluster heads and the intra-cluster communication has been made.
- Selection of Master Cluster Head from different cluster heads to minimize transmission process between CHs and sink node.

4. Simulation And Analysis

In this section, we perform simulations to analyze and evaluate the performance of the proposed technique. Here simulation is done on MATLAB. The simulation results depict that our proposed technique has better results in terms of the network lifetime. We get improvement in the lifetime of the network and with this more amount of information is gathered at the Base Station. To verify the improved algorithm proposed, we will compare the results with LEACH.

4.1 Simulation Setup

The different parameters which are used in our research has been mentioned in the Table I. We carried out our experiment using a simulation area of 100m * 100m. The initial energy provided to every node is 0.5J. Every node use to send a data packet of 4000 bits.

TABLE I. Different values used during simulation

Description	*Symbol*	*Value*
The Sensing area	$M \times M$	100m×100m
Number of nodes	N	100
The initial node energy	$E_{initial}$	0.5J
Energy consumed by the amplifier to transmit at a short distance	E_{fs}	10pJ/bit/m^2
Energy consumed by the amplifier to transmit at a longer distance	E_{amp}	0.0013pJ/bit/m^4
Energy consumed in the electronics circuit to transmit or receive the signal	E_{elec}	50pJ/bit
Data packet	k	4000 bits
Control packet	L_{ctrl}	100 bits
Data aggregationn energy	E_{da}	5pJ/bit/report

4.2 Deployment of sensor nodes and sink

Here 100 sensor nodes are randomly deployed in the sensor field. The location of sink is (50, 140).

4.3 Performance metrics

There are number of parameters which affect the performance of the WSNs. The initial energy provided to each node is

0.5J.This paper selects the two metrics that first node dies (FND), half of nodes die (HND) to evaluate the network performances. The network lifetime is depending on the number of alive nodes present. As the less number of nodes die, there will higher efficiency of using energy.

4.4 Election of Master Cluster heads

In this paper communication is take place between master cluster heads and sink node instead of cluster heads and sink node. Figure 5. shows simulation result of communication according to our technique.

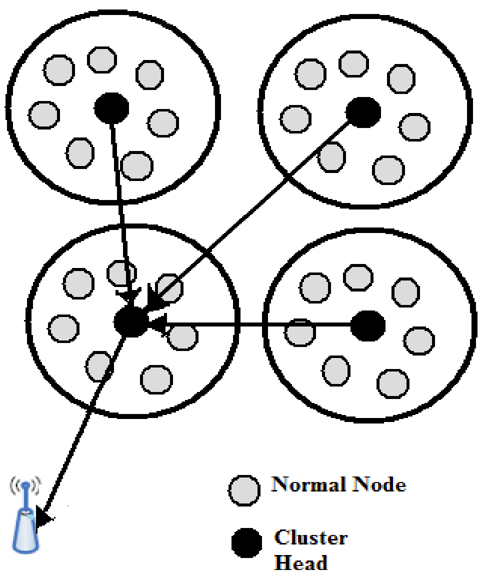

Figure 5. Master Cluster Head Technique

Every cluster head first selects its master cluster head and send its data to that particular node. These main nodes accept the data packets from cluster heads and then send aggregated data packets to the sink node. We can demonstrate from the figure that only one transmission is taking place. As we know that in sensor network most of the energy is consumed during the long way communication. So our main motive is to reduce the long way communication between cluster heads and the sink node. Our technique gives great improvement in the lifetime of the network.

4.5 Result for lifetime of the network

In Figure 6, the number of rounds is given on x-coordinate and y-coordinate represents for the number of dead nodes present per round. We calculated the FND and HND rounds for our technique. It can be demonstrate from the average results that first node dies around 1659 rounds and half nodes die at around 2226 rounds. Here we haven't taken external factors into consideration. The original LEACH algorithm began to its first node death at 1029th round and half node died in 1575th round. So above results mark a great difference in our proposed technique as compare to LEACH.

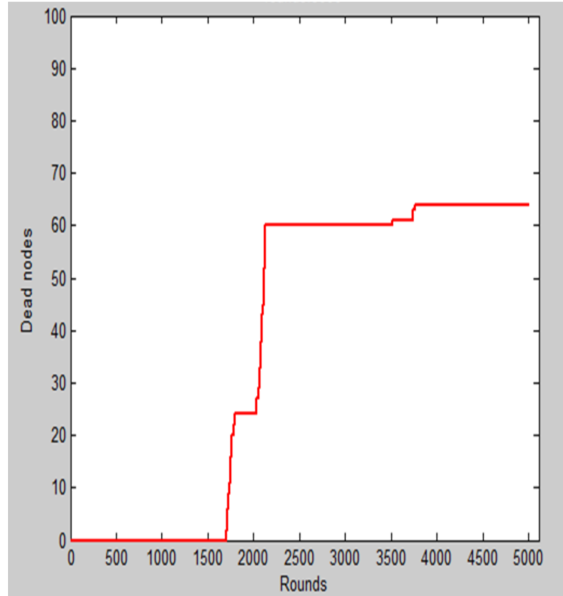

Figure 6. Network Lifetime of Proposed Technique

TABLE II. FND and HND for LEACH and Proposed Technique

Techniques	FND	HND
LEACH	1029	1575
Proposed Technique	1659	2226

TABLE II shows the comparison of FND and HND between LEACH and new technique. If we define network lifetime as per FND and HND rounds, then new technique gives improved results as compare to LEACH.

5. CONCLUSION

The energy absorbed during transmission of information from senor nodes to sink node is becomes a most critical issue in WSN. We have to modify some present algorithms which will boosts the energy of the sensor nodes and so that the lifetime of the network will improved. In this paper, we modify a clustering algorithm known as LEACH and provide new clustering technique. It will improve the some disadvantages of LEACH. To minimize the load of the network, minimum number of cluster heads has been elected in each transmission round. The simulation results will show that our proposed protocol increases network lifetime

6. ACKNOWLEDGMENTS

First of all, I would like to thank **Lovely Professional University** for giving me a chance to work under this topic.

I would like to express my sincere gratitude to my project supervisor Ms. Deepika Sharma for her continuous support and encouragement.

I would also like to express my deep gratitude to my father and mother for their care and support during my project. Without them the completion of this project would have been impossible.

7. REFERENCES

[1] Estrin D, Pisterk C D. "Connecting the physical word with pervasive networks". IEEE Pervasive Computing, 2002, pp.59-69.

[2] D.D.Chaudhary, S.P.Nayse, L.M.Waghmare (2011) "Application Of Wireless Sensor Networks For Greenhouse Parameter Control In Precision Agriculture", International Journal of Wireless & Mobile Networks (IJWMN) Vol. 3, No. 1, 2011,pp. 140-149.

[3] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan (2000), "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", Proceedings of the 33rd Hawaii International Conference on System Sciences– 2000.

[4] Peyman Neamatollahi, Hoda Taheri, Mahmoud Naghibzadeh, Mohammad-Hossein Yaghmaee (2011) "A Hybrid Clustering Approach for Prolonging Lifetime in Wireless Sensor Networks" International symposium on computer networks and distributed systems,2011, pp.170-174.

[5] S. Lindsey and C.S. Raghavendra, .PEGASIS:Power efficient Gathering in Sensor Information System. Proceedings IEEE Aerospace,Conference, vol. 3, Big Sky, MT, Mar. 2002, pp. 1125-1130.

[6] Mu Tong and Minghao Tang, "LEACH-B: An Improved LEACH Protocol for Wireless Sensor Network" 2010 IEEE .