IJCSC International Journal of Computer Science & Communication Vol. 1, No. 2, July-December 2010, pp. 321-327

Metric based Efficient Traffic Management in Intranets using Incremental Advanced DST Approach

J. Amudhavel[1], M.S. Saleem Basha[1], T. Vengattaraman[1], P. Dhavachelvan[1] & R. Baskaran[2]

[1]Department of Computer Science, Pondicherry University, Puducherry, India.
[2]Department of Computer Science and Engineering, Anna University, Chennai, India.
Email: {[1]amudhavel86, [1]smartsaleem1979, [1]vengat.mailbox, [1]dhavachelvan, [2]baskaran.ramachandran}@gmail.com

ABSTRACT

In the recent computing network environment based on reliability and data availability data availability plays a vital role in Network environment. Traffic Management(TM) is widely used in distributed network environment to reduce access cost and improve data availability [5],[6]. It also focuses in maintaining the effective availability of data and maintaining the reliability of data. This paper addresses in maintaining the effective local traffic management in network by using optimal algorithm of Distributed spanning tree and the results based on QOS parameters. With this proposed approach we maintain the effective local traffic management deals with the effective forwarding transmission of packets in the Networks. Therefore, this paper evaluates the efficiency of existing techniques, and proposes a more efficient local traffic management algorithm. The new algorithm leverages existing techniques which are shown to be efficient. It also contrasts the effectiveness of DST based approach in Transfer of packets Management in the distributed network environment.

Keywords: Traffic Management, Data Availability, Local Traffic Management, Distributed Spanning Tree, QOS Parameters and Performance Evaluation

1. INTRODUCTION

In the computing environment, accessing popular objects can bring the intended items close to the clients and thereby it is possible to reduce the access latency and Network traffic [17] in the distributed environments. A proper Traffic Management scheme can also reduce access time and improves fault tolerance and load balancing capabilities in order to improve the serviceability and availability of the intended environment. There are many approaches provided to solve the critical issues in the existing filed of Traffic Management and they differ in terms of their location, permanency, scope and applicability. Based on the scope and applicability, the traffic methods can be classified as local and global. The scope of the global Traffic Management schemes are so broad and applicable for inter group management, whereas the local Traffic Management schemes are so confined and they are applicable for intra group management. The Local Traffic Management (LTM) systems are associated with single group, where the numbers of nodes are fixed and limited. Transmission of a data item may be inconsistent in an environment where frequent data updates occur, particularly in case of LTM environments. Though the scope for a local group is limited, but due to the nature of the networks, where the nodes may be disparately placed in uncertain locations. In such environments, the enhanced degree of serviceability and availability are essential and they can be improved only by an effective the Local Traffic Management schemes. The Propagation models suggested a procedure like the updated version of document is to be delivered to all copies as soon as a change is made to the document at the origin server. Although the copies always keep the latest version of the originals, this method may generate significant levels of unnecessary traffic if documents are updated more frequently than accessed.

Y. Chang[1], addresses the interoperability issues in terms of network performance and fairness of bandwidth allocation when an ATM network consists of switches using different rate control mechanisms, namely, the Explicit Forward Congestion Indication (EFCI) mode and the Explicit Rate (ER) mechanism. Simulation results show that these algorithms will interoperate as long as the switch implementation conforms to the end system.

Andrea Borella[2], proposed dynamic management of DQDB Multiple Access Control. It is based on the opportune activation of the Increasing Counter Priority Controlled mechanism, driven by the traffic levels present at the user interfaces. The main goal of that paper is to make the operation of the Metropolitan Area Networks clear and effective for multipriority interworking between remote computers. They compared the results with the results provided by a DQDB network simulator.

J.W.K. Hong[3] proposed the design and implementation of a portable, Web-based network traffic monitoring and analysis system called WebTrafMon. It provides monitoring and analysis capabilities not only for traffic loads but also for traffic types, sources and destinations. The probe extracts raw traffic information from the network, and the viewer provides the user with analyzed traffic information via Web browsers.

Intae Ryoo[4] proposed a new RTM (real-time traffic management) scheme that can effectively manage VBR (variable bit rate) traffic having unpredictable characteristics in ATM (asynchronous transfer mode) networks. In addition to its precise cell control capability, the proposed scheme intends to efficiently use the resources of high-speed, high-performance broadband networks without any deterioration in QoS (quality of service) of the accepted connections.

Vincenzo Catania [12] proposed the correction required by the CAC algorithm to avoid this risk. They compare two different policing mechanisms, one based on conventional logic and another on fuzzy logic, assessing the influence of their degree of selectivity on the additional bandwidth the CAC algorithm needs to reserve in order to guarantee the QoS requirements of all connections. *Moises R. N Ribero*[13] proposed reactive method of Traffic management to an optical packet switching node.

G. Chiruvolu [14] VBR video traffic that exhibits Long-Range Dependence (LRD) based on a fractional ARIMA (1,d,0) model. The Short-Range Dependent (SRD) Auto Regressive (AR) model for prediction of VBR video traffic has also been considered for evaluation of the proposed dynamic bandwidth allocation scheme and the performance has been compared to that of LRD-based prediction.

In summary, though many of the proposals dealt the issues in the Traffic Management management, some of the open issues are not yet addressed; no mechanism for controlling the time based models, no suitable model for maintaining effective traffic management in the network environment. Hence from these perspectives, in this paper, we proposed an integrated Optimal DST approach for local Traffic management in the network environments. This model addresses the specific issues of dynamic transfer of packets management in intra circle environments.

1.1. System Model

1.1.1. Distributed Spanning Tree

The Distributed Spanning Tree (DST) [18] is an overlay structure designed to allow the use of tree traversal algorithms by avoiding the usual tree's bottlenecks with improved scalability. It supports the growth from a small number of nodes to a large one and automatically balances the load in the intended environments. DST allows more efficient executions of search algorithms in term of number of sent messages and in terms of load balancing. Though the concept of DST [15], [16] has been used to simplify and optimize the flooding algorithms and also in TTL based search algorithms, but for the effective transfer of packet management there is no well defined structure to achieve the maximum utilization.

A DST structure can be described with three perspectives; logical level, interconnection level and the physical level. The logical level is useful to understand the basics concepts of a DST and its organization. The interconnection level is used by software to run a distributed tree and it is responsible of linking the nodes together. Finally, the physical level is the mapping of the interconnection level on a physical network layer. Here we considered the logical and interconnection levels for the work presented in this paper.

1.1.2. Distributed Spanning Trees in Communication

The spanning tree is to be computed in a distributed manner. This is done by attaching to every node in the graph a communicating program that exchanges messages with the programs at its neighbor nodes. In particular the state of each program is uniquely determined by the program variables.

2. PROPOSED SYSTEM

Our proposed system forms the effective traffic management mechanism using the following DST Algorithm.

ALGORITHM (Distributed Spanning Tree Algorithm):

- *Step-1:* Every node sets its state to Inactive.
- *Step-2:* Construct a spanning tree, *T* of the underlying un-weighted graph. Any node with more than one neighbor in the tree (an interior node) must change its state to Receiving.
- *Step-3:* Each node *i* determines the identities of its neighbors in *G* and stores identity and distance. Determine *D* of the graph *G*. Each node's behavior is determined by its state.

```
For each node i ∈ V
 If the state of i is Receiving
  Run the subroutine RECEIVING NODE PROCESSING(i);
  Wait for ackmnt
   Get ackmnt do operation
                Check latest state of Data
                If requested is to read
                      Initiate Read
                Else
                      Initiate Write
 If the state of i is Transmitting
  Run the subroutine TRANSMITTING NODE PROCESSING(i)
End For
```

The above algorithm represents the behavior and state of the node. It determines whether the node is going to involve in which operation.

Step. 4: Transmit the final to every node from a saturated node. Any node contains the final all pairs shortest distance. The node(s) will create a final message consisting of *D* and send this message to all its neighbors in the spanning tree *T*. Let us discuss the formation of DST in the following sections.

Definition 1: Let S be the proposed DST for TM providing Local Traffic Management (LTM) of testing environment consisting of level up to 4 and it can be defined as,

$$S = \begin{Bmatrix} (T_1, T_2, T_3, T_4, ...T_n) \\ (R_1, R_2, R_3, R_4, ...R_n) \\ (Q_1, Q_2, Q_3, Q_4, ...Q_n) \\ (P_1, P_2, P_3, P_4, ...P_n) \end{Bmatrix} \quad (1)$$

$$S = \sum_{i=1}^{i=n} Pi \quad (2)$$

where,

- '*T*' indicates the node in the Networking environment and '*n*' is the number of nodes in DST in TM where *T* and *N* is defined as *T* > 0 & *n* > 0.
- '*R*' is the group formed by the interconnection between the interconnection nodes and '*n*' is the number of Preliminary Group (PG) formed in the proposed DST where *R* and *N* is defined as *T* > 0 & *n* > 0.
- '*Q*' is the Level1 Group (L1G) formed by the interconnection between the groups and 'n' is the number of groups formed the next level (i.e. level1) of DST where *Q* and *n* is defined as *T* > 0 & *n* > 0.
- '*P*' is the Level2 Group (L2G) formed from the interconnection of the level1 groups of the DST and '*n*' is the number of groups formed in the level2 of the DST where P and 'n' is defined as P>0 & n>0.

Any node in the spanning tree that receives *D* will store *D* locally and send *D* to all its tree neighbors except the tree neighbor which it was connected from which it received *D*. Hence the node which passing the information or writing the information from the other nodes is called the LH's of the DST.

2.3. Algorithm to Construct Optimal DST for Effective Traffic Management

The following algorithm defines the construction of optimal DST for TM in network environment. This algorithm defines how the LH's are created and defines the way it done with their operations.

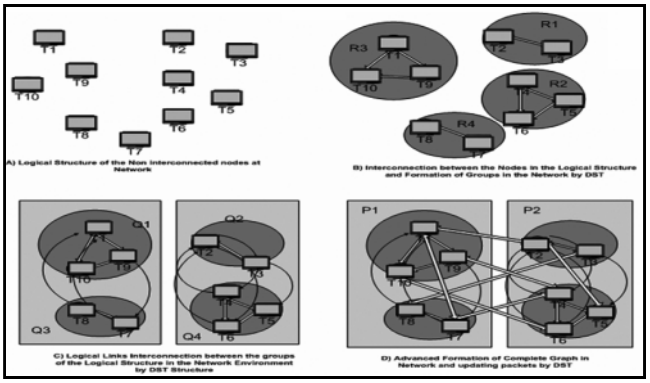

Fig.1: Effective Traffic Routing and Interconnections between the Groups

```
Procedure (initialize distance)
        For each node i ∈ V
  //i transmits an Identification message containing its identity along each edge incident at i in G
  // |, upon receiving the identities of its m neighbors, creates a distance matrix, DSTi, of size
            For each j, k ∈ indexes of DSTi
                    If j == k then DSTi[j,k] ← 0. //means both should not be identical
                    If j[i] &&k [i]≠0
        If j == i or k == i then DSTi[j,k] //Finding the level heads of the network by weightage
        End For
        Else DSTi[j,k] ← DSTi [j,i] + DSTi [i, k].
End For

Procedure (RECEIVING (i))
// Let DSTNi be the set of neighbors of node i in DST created in Step 2 algorithm.
// Let count be the number of the acknowledgement that it has received since it changed state to
Receiving. Initially count is set to 0.
//Let DSTNi of type boolean in which all entries are initialized to False.
While count < | DSTNi | − 1
Receive message DSTi from neighbor j
count++
Call ProcessMessage(DSTi)
EndWhile

Set the state of node i to Transmitting to the LH's from the GH.

Procedure (process message(DSTi))
        For each index k in DSTi //assigning indes to the nodes
        if k is not an index of DSTj // to check whether it involves in the previous operation
              // extend DSTj by one row and one column corresponding to k
                    Set DSTj [k, m] ← DSTj [m, k] ← ∞ //select the node to pass the which is
              determined by the DSTi
        End For
              Set DSTj [k, k] ← 0
        End If
        For each k, m, n indexes of DSTj
              if DSTj[k,m] > DSTj[k,n] + DSTj[n,m] //Adding the nodes to the level head
              DSTj[k,m] ← DSTj[k,n] + DSTj[n,m]
        End For

Procedure(Transmission process(i))
1. Node i transmits DSTi to its only neighbor in T from which it has not acknowledgement.
2. If i receives another acknowledgement, say from j, then i calls ProcessMessage(DSTj)
```

With the above algorithm the effective DST is formed. The efficiency of the algorithm results in efficient group formation and maintaining interconnections between the groups and nodes. In the above sections, we have seen the definition and the formation of DST using an optimal algorithm. Let us see the step wise formation of interconnection between the nodes, and the formation of Level Heads (HN's).

Now, after the formation of (L1G) which was formed by the PG's they had interconnection between themselves and form the superior group. Following the same principle, all the (L1N's) are interconnected and form the superior group of what they are concerned with. The formations of the (L1N's) are as follows:

While in the formation of the L1G's the no of PG's which forms a L1G's is not a constraint. It should form on their own. The interconnection of the L1G's is done by various factors like read operation, replying relativity, access rate etc. Like that the consecutive Level nodes are clustered for intercommunication.

The effective optimal Network DST is formed for providing the effective data effective transfer of packet management by the above paradigms. The operations and effectiveness in the effective transfer of packet management are discussed in the following sections.

3. MECHANISM OF LOCAL TRAFFIC MANAGEMENT IN DST (LTM-DST)

The concept of LTM can be described in terms of two primary operations read and write operations. The sequences of operations are explained as follows:

Read Operation: It refers the process of reading the recently updated documents for further processes and it can be briefly described in four different phases.

Phase 1: The LH which wants to read the latest updated replica send a read request message to its overall HN. If the HN is concerned with other operation it will sent wait message to the requested LH. The read requested node which receives the wait message understands that the HN is locked with some task. Here HN sends the message when it is free of operations. The node which requires the operations get the acknowledgement from the HN.

Phase 2: The read requested LH which is acknowledged by its HN waits to receive the requested data item. HN sends a reply message which contains the latest replica of the data item.

Phase 3: Since each HN having is the latest replica of data so there no need by the requester LH to check for the latest version of received data item.

Phase 4: Since the LH's are updated with latest replicas, there is no need for the other nodes to get the latest replica to read from the HN. They get their updates from the LH's itself. Hence by read operation is done faster when compared to traditional method. Therefore all the connected nodes and groups are updated at the same time and the new data is overwritten.

Write Operation: It refers the process of writing the recently updated documents for further processes and it can be briefly described in four different phases.

Phase 1: The LH which wants to write the latest updated replica send a write request message to its overall HN. If the HN is concerned with other operation it will sent wait message to the requested LH. The write requested node which receives the wait message understands that the HN is locked with some uncompleted task and request whenever the HN becomes free.

Phase 2: The write requested LH which is acknowledged by its HN waits to receive the requested data item. HN sends a reply message which contains the latest replica of the data item if it is free from any other operations.

Phase 3: Since each HN having is the latest replica of data so there no need by the requester LH to check for the latest version of received data item.

Phase4: Since the LH's are updated with latest replicas, there is no need for the other nodes to get the latest replica to write from the HN. They get their updates from the LH's itself. Hence by write operation is done faster when compared to traditional method.

4. SIMULATION AND ANALYSIS

In this section, we show simulation results to investigate the characteristics of the proposed DST technique for efficient Local Traffic Management on in DST. We used OMNeT++, an object-oriented modular discrete event network simulator. An Network of 19 nodes (comp1, comp2, comp3 …comp19) interconnected randomly and spread in some distant geographical location is used to measure efficiency of our DST technique. It is assumed that the interconnection channel have propagation delay of 100 ms. On simulation proceeds, it is observed that nearly 75 read operations are performed by nearly 19 nodes in 300 seconds (making 10 ms delay can reduce time to 30 seconds). Thus by using DST as a interconnection structure we are reducing the message pass for Read Operation at high level which makes the operation fast and consistent. In simulated DST for TM in networking environment contains LH's comp2, comp3, comp4, and comp5, respectively other nodes are connected to the LH (Level Heads) which they are connected with. Values of i, j, k, l, m and n in our simulation values shows that the LHs are not uniformly distributed among the HNs which implies that the number of LHs under the interconnection made by the nodes in the runtime. With this formulation of nodes by DST increases performance in message passing, updating the packet in consistent and constant manner. The overall interpretation of the DST based Networking environment is that the total number of Message Pass required single Read Operation in DST is greater than or equal to number of message pass between LH, HN and the number of nodes.

Table 1
Performance Comparison of No. of Read and Write Operations Without and with DST in TM

Computer Name	*No. of Read Operations without DST*	*No. of Read Operations with DST*	*No. of Write Operations without DST*	*No. of Write Operations with DST*
Comp1	7	6	6	4
Comp2	5	4	5	2
Comp3	7	5	4	3
Comp4	4	3	7	4
Comp5	6	5	4	2
Comp6	9	6	9	4
Comp7	6	5	6	3
Comp8	3	4	5	1
Comp9	5	4	3	3

Table 1 Contd...

Table 1 Contd...

Comp10	3	4	7	4
Comp11	6	5	3	4
Comp12	3	2	4	2
Comp13	2	3	6	4
Comp14	5	3	5	3
Comp15	3	2	4	2
Comp16	4	3	2	3
Comp17	2	5	5	4
Comp18	5	3	9	3
Comp19	6	3	6	4
Total Operations	91	75	100	59

Our DST based simulation deals with 59 write operations are performed by nearly 19 nodes in 400 seconds (making 10 ms delay can reduce time to 40 seconds). Thus by using or implementing DST in the Networking environment we are reducing the no. of message pass for Read Operation and number of write operations which makes the operation fast and consistent. Table 1 show Number of Read and write operations performed by nodes of the Networking environment. Through maintaining effective traffic in packet management deals with the updating of the packet at the right time that is achieved in the graph. Hence in the DST operations also the read and write operations takes place simultaneously to avoid congestion.

The Total numbers of Operations are summed to give clear visibility of comparison. The above values in the table show the no. of read and write operations which show the higher priority with minimal number of message passing.

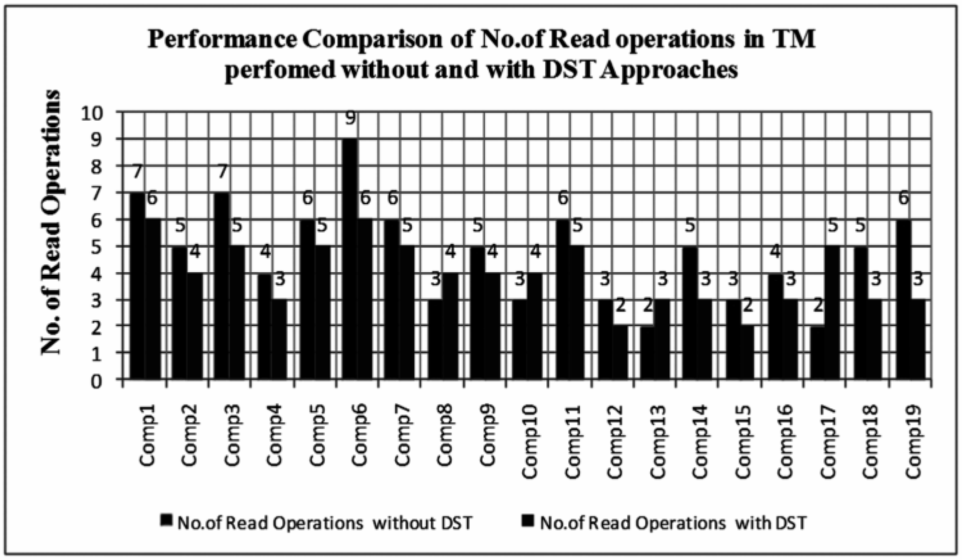

Fig. 2: Performance Comparison of No. of Read Operations in TM Performed Without and with DST Approaches

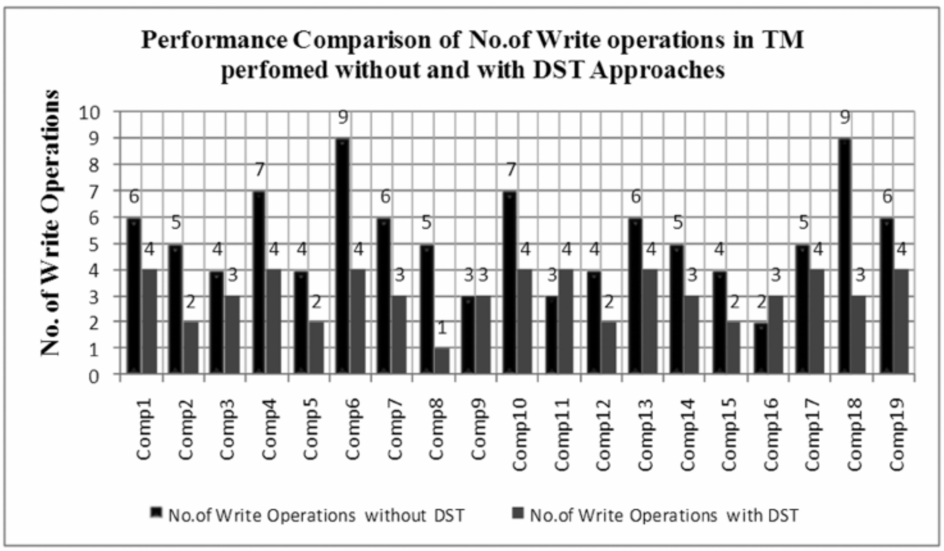

Fig. 3: Performance Comparison of No. of Write Operations in TM Performed Without and with DST Approaches

Fig no.2 show the no. of read operations which is carried out without and with DST while read operations take place. The above graph shows the performance of operations which is done with the 19 computers. It shows that the higher performance in read operations. Here we get the decrease in the message passes when compared to the ordinary read methods.

Fig no.3 shows the no. of write operations which is carried out without and with DST while writes operations take place. The below graph shows the performance of operations which is done with the 19 computers. It shows that the higher performance in read operations. Here we get the decrease in the write operations when compared to the ordinary write methods.

It shows that the higher performance in write operations with congestion free traffic. When we closely observe the above graph write operations have the higher priority when compared to the read operations with less no. of operations to maintain the updated packet in every LH's.

Table 2
Comparison

Operations	*Without DST*	*Optimized using DST*
Time taken (in sec)	227	168
No. of read operations performed in 300s with response delay of 100ms	8.54	6.77
No. of nodes involved in read operation in 300s with response delay of 100ms	27	31
No. of write operations performed in 300s with response delay of 100ms	18	24
No. of nodes involved in write operation in 300s with response delay of 100ms	16	21

The above Table no 2 figures out the comparison between the two approaches and by the DST optimized Networking environment it performs well when compared to the ordinary Networking environment without DST.

5. CONCLUSION AND FUTURE WORKS

In this paper, we have presented the way for maintaining the effective TM using DST for network environment as an interconnection structure. DST in TM increases the availability, reliability of data, updating frequency, number of message operations and Bottle neck problems. From the simulation analysis it is shown that network environment with DST shows high performance TM typical network architecture. By employing DST in network environment with cost of few message pass we can assure consistency[7], [8], [9],[10], [11] of Data item, easy recovery of Data on occurrence of crash, load balancing among the wireless nodes.

DST also provides concurrent read operation of latest packet and dynamically network environment creation makes the system Fault-tolerant. We simulated Dynamic routing simulation can make solution for many bottleneck issues in Internet.

Our work can also be extended for many packet management types like global, peer etc. to achieve TM as required by the application. More simulation results will be carried out from the practical viewpoint. Furthermore, theoretical modeling for effective traffic control should be expanded and sophisticated.

REFERENCES

[1] Y. Chang, N. Golmie, David Su, "Interoperability between EFCI and ER Switch Mechanisms forABR Traffic in an ATM Network", *Computer Communications,* **19**, 1996, 653-658.

[2] Andrea Borella, Paolo Micucci b, "Dynamic Management of Non-isochronous Multi-priority Traffic in DQDBmetropolitan Area Networks", *Computer Communications*, **20**, 1997, 7 15-72.

[3] J.W.-K. Hong, S.-S. Kwon, "WebTrafMon: Web-based Internet/Intranet Network Traffic Monitoring and Analysis System", *Computer Communications*, 22, 1999, 1333–1342.

[4] Intae Ryoo, Hitoshi Aida, Tadao Saito, "Design and Performance Analysis of a New RTM in ATM Networks", *Computer Communications,* **19**, 1996, 205-215.

[5] T. Hara, "Effective Replica Allocation in Ad Hoc Networks for Improving Data Accessibility," *Proc. IEEE INFOCOM, pp. 1568-1576, 2001.*

[6] T. Hara, "Replica Allocation Methods in Ad Hoc Networks with Data Update," *ACM-Kluwer J. Mobile Networks and Applications,* **8**, No. 4, pp. 343-354, 2003.

[7] T. Hara and S.K. Madria, "Consistency Management among Replicas in Peer-to-Peer Mobile Ad Hoc Networks," *Proceedings IEEE Symposium Reliable Distributed Systems (SRDS '05),* pp. 3-12, 2005.

[8] Saroja Kanchi and David Vineyard, "An Optimal Distributed Algorithm for All-pairs Shortest-path", *International Journal "Information Theories & Applications"* **11**, 141-146.

[9] T. Hara and S.K. Madria, "Data Replication for Improving Data Accessibility in Ad Hoc Networks," *IEEE Trans. Mobile Computing,* **5**, No. 11, pp. 1515-1532, Nov. 2006.

[10] Zhou SU, Jiro KATTO, Yasuhiko YASUDA, and Yu CHEN, *Members, IEEE*, *Consistency and Update in Mobile Overlay Networks.*

[11] Awerbuch, Y. Bartal, and A. Fiat, "Heat and Dump:Competitive Distributed Paging," *Proc. IEEE Symposium. Foundations of Computer Science (FOCS '93), 1993.*

[12] Vincenzo Catania, Giuseppe Ficili, Daniela Panno, "On the Impact of Traffic Control Algorithms on Resource Management in ATM Networks", *Computer Communications*, **22,** 1999, 258–265.

[13] Moises R.N Riberio, "Traffic Management in Photonic Packets Switching Nodes by Priority Assignments and Selective Discarding", *Computer Communications,* **24**, 2001, *1689–1701.*

[14] G. Chiruvolua, R. Sankarb, N. Ranganathan, "VBR Video Traffic Management using a Predictor-based Architecture, *Computer Communications*, **23**, 2000, 62–70.

[15] Sylvain Dahan, Jean-Marc Nicod and Laurent Philippe, "The Distributed Spanning Tree:A Scalable Interconnection Topology for Efficient and Equitable Traversal", *International Symposium on Cluster Computing and the Grid, 2005 IEEE.*

[16] Sylvain Dahan, "Distributed Spanning Tree Algorithms for Large Scale Traversals", *11th International Conference on Parallel and Distributed Systems (ICPADS'05),* 2005 IEEE.

[17] Lei Xiao, Liangzhong Shen, ZhongYi Hu, Fei Zhou, "Improving Search in P2P by Location Identification," *Environmental Science and Information Application Technology, International Conference on*, **1**, pp. 519-522, 2009.

[18] Sylvain Dahan, Laurent Philippe, and Jean-Marc Nicod, "The Distributed Spanning Tree Structure", *IEEE Transanctions on Parallel and Distributed Systems*, **20**, No. 12, Dec 2009.