IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 9, Issue 6 (Mar. - Apr. 2013), PP 05-12
www.iosrjournals.org

Implementation of Model "RSM2"A Reliable and Scalable Multicast Model

Ruchi Gupta[1]
[1](*Assistant Professor ,Computer Science & Engineering, Sri Venkateshwara University, U.P, India)*

Abstract : *We describe a Model RSM2 (A Reliable and Scalable Multicast Model). The approach we adopt uses Proactive-Routing, Minimum-Cost Path Algorithm, Combo-casting, Active Server Based Recovery, NACK-Based acknowledgement, Buffer Management, Optimized Flooding Algorithm. The main characteristic of RSM2 model is, to provide complete multicasting, i.e. at the same time more than one node can act as sender. This model provides one-to-many communications as well as many-to-many communications. Through this model, we tried to create an improvement over RMTP. To describe the generality of this Model, we have designed an algorithm MCPA(Minimum Cost Path Algorithm).*

Keywords - *Combo-Casting, Optimized Flooding Algorithm, Proactive Routing, Reliability, Scalability.*

I. Introduction

A reliable multicast is the requirement of various applications such as Multicast File-Transfer, shared white-board, distributed interactive simulation and distributed computing. There have been many protocols have been introduced like SRM, RMTP etc. to provide reliability and scalability in multicasting communications. Our research is focused on providing an idea that act as an alternative for RMTP. Our Model apt a flat approach, that makes it available to work in many-to-many kind of communication environment. Our Multicast Model tried to provide a cost-effective, delay-effective and efficient path to deliver the packets. In RSM2 Model, Dynamics Manager (DM) is the main focus of this model and plays an important role. DMs are the specialized Machines, with network computational capabilities, that makes this model, to be available in both wired and wireless networks. Dynamics Managers act as listeners and calculator to perform network computations.

II. PROBLEM STATEMENT

There have been many models and protocols (SRM, RMTP...) have been introduced, to provide reliability and scalability in multicasting communications. The goal of the research is to design an algorithm describing Multicast Model (RSM2), which achieves scalability and reliability. Along with it, our model opt flat approach. The problem with hierarchical approach is that, every time when a receiver becomes the sender, entire hierarchy was changed[1]. The hierarchical Model, RMTP, does not fit in a situation of, where, many nodes can send data simultaneously at same time. Another problem was to choose an effective Designated Receiver, to deliver the packets to all the nodes, under it. Our Multicast Model removes all the above stated problems and provides a cost-effective, delay–effective path to deliver the packets.

III. LITERATURE REVIEW

MULTICASTING provides an effective and efficient way of disseminating data from a sender to a group of receivers. Instead of sending a separate copy of the data to each individual receiver, the sender just sends a single copy to all the receivers. A multicast tree is set up in the network with sender at the root node and the receivers at the leaf nodes. Data generated by the sender flows through the multicast tree, traversing each tree edge exactly once [1]. However, distribution of data using the multicast tree in an unreliable network does not guarantee reliable delivery, which is the prime requirement for several important applications, such as distribution of software, financial information, electronic newspapers, billing records, and medical images [1].

A. Various Approaches to Reliable Multicasting

Any reliable multicast protocol requires some recovery mechanism. A generic description of a recovery mechanism consists of a prioritized list of recovery servers/receivers (clients), hierarchically and/or geographically and/or randomly organized. Recovery requests are sent to the recovery clients on the list one-by-one until the recovery effort is successful. There are many recovery strategies available in literature fitting the generic description [2].

Scalable Reliable Multicast (SRM) [5] is a simple and robust retransmission-based protocol. SRM uses IP multicast to multicast messages to all the members of the reliable multicast group. In turn, IP multicast uses underlying spanning trees to disseminate these messages to all group members in a best-effort manner, i.e., with no delivery or performance guarantees. Packet recovery in SRM is initiated when a receiver detects a loss and schedules the transmission of a request; an error control message requesting the retransmission of the missing packet. If a request for the same packet is received prior to the transmission of this local request, then the local request is rescheduled by performing an exponential back-off. When a group member receives a request for a packet that it has already received, the group member schedules a reply; a retransmission of the requested packet. If a reply for the same packet is received prior to the transmission of this local reply, then the local reply is cancelled. Using this scheme, all session members participate in the packet recovery process and share the associated overhead. SRM minimizes duplicate error control and retransmission traffic through deterministic and probabilistic suppression. These suppression techniques prescribe how requests and replies should be scheduled so that only few requests and replies are transmitted for each loss. Deterministic suppression prescribes that request and reply scheduling timers be set proportionately to the distance from the source and the requestor, respectively. Thus, the requests of ancestors suppress those of their descendants. Probabilistic suppression prescribes that members that are equidistant from the source and the requestor probabilistically vary the scheduling times of their requests and replies, respectively. Thus, sibling requestor and replier hosts are afforded the opportunity to suppress each other. Unfortunately, suppression introduces a trade-off between the number of duplicate requests and replies and the recovery latency — the scheduling of requests and replies must be delayed sufficiently so as to minimize the number of duplicate requests and replies [5].

RMTP[1] is based on a hierarchical structure in which receivers are grouped into local regions or domains and in each domain there is a special receiver called a designated receiver (DR)**,** which is, responsible for sending acknowledgments periodically to the sender, for processing acknowledgment from receivers in its domain, and for retransmitting lost packets to the corresponding receivers. Since lost packets are recovered by local retransmissions as opposed to retransmissions from the original sender, end-to-end latency is significant reduced, and the overall throughput is improved as well. Also, since only the DR's send their acknowledgments to the sender, instead of all receivers sending their acknowledgments to the sender, a single acknowledgment is generated per local region and this prevents acknowledgment implosion. Receivers in RMTP send their acknowledgments to the DR's periodically, thereby simplifying error recovery. In addition, lost packets are recovered by selective repeat retransmissions, leading to improved throughput at the cost of minimal additional buffering at the receivers [1]

IV. SYSTEM DESIGN

RSM2 Architecture and Assumptions[6]

RSM2 is based on flat architecture. Let all the nodes in the network are connected with each other, through local area switches or routers. These routers are collocated with Dynamics Manager (DM).The assumptions made in the design of model is as follows:

1. Active Server Based Local Recovery: It makes use of specially designated hosts that have all the network- computational ability, known as Dynamics Manager (DM).
2. Dynamics Manager: DMs are collocated with each router of the network. They have the entire essential network computational ability likes – to maintain a proper data of the nodes that are linked with it, to compute a Partial_cost_ matrix, and to assign a priority.
3. Cost Matrix: In a Heterogeneous environment, it is not possible that all links are alike. Hence, on the basis of their property we assign a cost to each link. Cost matrix shows the cost associated with each link.
   If there is no link between any two nodes, in that case, matrix assigns the cost as infinity.
4. Priority Matrix: In the model, priority matrix is designed from the cost matrix. To send the packets, the path is decided on the basis of priority matrix.
5. Echo packet: Whenever a node wants to send the data to others, then sender first sends an echo packet. In that packet , there are two fields :
   a. Group-id: It indicates the group to whom sender wants to communicate.
   b. Sender-id: It defines the address of the sender.
   Echo packet as moves through the network, it stores the information about path and cost.
6. Response Packet: This packet is sent by DM, to the sender, in response to Echo Packet.
7. IGMP drive: As the Echo packet received by the DMs, they run IGMP protocol, to know the group status of the nodes under them. This report is forwarded to their neighbouring DMs. These reports make the DMs up-to-date always. Also, DMs periodically run IGMP protocol and send the status report to others.
8. NACK –Based Combo-Casting Approach: Since, each data packet has a unique sequence number. Hence, if a packet is missed by a node, it sends a NACK to the DM. DM do not retransmits that packet immediately. As the DM gets an ACK / NACK for the last packet, DM analyses NACKs. On the basis of no. of NACKs, DM decides to retransmit the packet by unicasting or multicasting.

9. Buffer Management: Each DM has two buffers, one buffer for data packets, and another buffer for NACKs .Initially, the buffer capacity is assumed to be unlimited. But, the capacity is confined, as the first packet is received.

V. PROPOSED ALGORITHM : MCPA

Basis for the Proposed Algorithm – MCPA (Minimum Cost - Path Algorithm)[6]

- DESIGN OF COST MATRIX
  The cost matrix is designed to define the cost between each pair of nodes, with the following cost constraint:

$$C[i][j] = \{INFINITY, iff\ (i,j) \notin E \vee (\forall(i,j), i = j$$
$$otherwise, Assign\,Cost\}$$

- DESIGN OF WORK MATRIX
  The work matrix is designed to define the priority to the edges, with the following work constraint:

$$W[i][j] = \{Leave\,Edge(i',j'), if\ i = j' \wedge j = i',\ \forall(i,j) \wedge (i',j') \in E$$
$$otherwise, Include\,Edge(i',j')\}$$

- DESIGN OF PRIORITY MATRIX
  The priority matrix is designed to define the priority to the edges, with the following priority constraint:

$$P[i][j] = \{\ Exclude\,Edge(i,j), if\ \forall(i,j) \in E\ forms\,loop$$
$$otherwise, Assign\,\mathrm{Pr}\,iority\,\forall(i,j)\}$$

- DESIGN OF DATA-PATH
  The data delivery path is designed to deliver the data to the desirable recipient's. Desirable recipients belong to the specific group(s), to whom Sender wants to communicate.

  Data path ⟵ Choose (u,v) ∈ {specific group(s)}.(u,v) ∈ E.

- COMBO-CASTING
  Dynamic Managers first stores all the NACKs, and wait for a random amount of time, T_{NACK}. As, the time gets out DMs decide either to retransmit the data through multicasting or via unicasting. If more NACKs are received for the same packet then DM subgroups them into a new group and multicast the missed data-packet. In case, there is one or two nodes send NACK for the same packet, then, DMs unicast the missed data packet.
- PRIORITY TIMER MANAGEMENT
  The Priority matrix is used to set the priority timer. Since the packet already contains the path and priority of each path. DMs gets the priority for the path, through which packet will reach to destination via DM(s). If DM finds the link has lower priority-value, and then T_{prio} will be in function for small unit of time. Since, the link has low cost and having less chances to miss the data, if passes through this link. So, as the T_{prio} gets out data will be emptied out of the buffer.
- OPTIMIZED FLOODING ALGORITHM (OFA)
  When the DM gets the packet, starts T_{stores} timer for t_{store} time. As the t_{store} time gets out, DM drops the packet, but save its seq. No for (2 t_{store}) some time. It prevents looping. As DM receives the same packet or packet with same sequence number, then DM will discard it and not floods to the network.
  As, the DM gets updated information regarding packet or topology's change, then it immediately floods to the network.
  Procedure OFA()

```
{
  // T_store ⟵ time to store the packet.
    Step 1.    DM ⟵ getpacket();
    Step 2.    pck ⟵ no. of packets to be send in one session.
    Step 3.    For (i=1;i<=pck;i++)
             {
             T_store = 20ns.  // assume  each incoming packet will be stored in buffer for 20ns, first .
               While (T_store > 0)
                  {
               Stores the packet;
                if (Packet_in ==Packet_store)
                {
               Discard the packet
                Stop flooding ;
```

```
                }
              Else
               {
               store the packet
               forward to connected node except that from which it come;
                 }
              }
        }
Step 4.     Drop the packet;
Step 5.     Save the sequence no;
Step 6.     If (seq_noPacket_in == seq_noPacket_empty)
            Discard the packet;
Step 7.     Else store the packet and Go to Step-3.
```

VI. Conclusion

Wireless networks are an emerging new technology that will allow users to access information and services electronically, regardless of their geographic position. Wireless networks can be classified in two types.

Infrastructure network: Such a kind of wireless networks comprises of a network with fixed and wired gateways. The hosts are mobile .The mobile unit can move geographically during the communication. When it goes out of range of one base station, it connects with new base station and starts communicating through it. This is called handoff .In this approach base-station are fixed [3].

Infrastructure less (adhoc) networks: In contrast to infrastructure-based networks, in adhoc networks all the nodes are mobile and can be connected dynamically in an arbitrary manner. All nodes of these networks behave as routers and take part in discovery and maintenance of routes to other nodes in the network.

There is a wealth of literature on reliable multicasting. Several new papers have also appeared in the recent literature that focuses on Wide Area networks [1] describes the design of Reliable Multicast Transport Protocol that uses efficient local-recovery technique for serving the missing packets.

Our work is closely related to RMTP, with significant differences .Let us study in details the comparative study of RSM2 over RMTP.

[1] Design: RSM2 opt flat design approach while RMTP works on hierarchical design. Flat design facilitates RSM2 to work in all the kinds of situation i.e.

a) One-to –Many:
RSM2 fits in the situation, where one sender communicates with many receivers on the network. RMTP is also good in this scenario.

b) Many –to-Many:
RSM2 fits in this situation, where many senders want to starts communication at the same time. RMTP does not fit in this scenario, it opts hierarchical approach, and in hierarchical, at one time only one node is the root of that hierarchy, i.e. the original sender.

c) ALL-to-ALL:
RSM2 fits in this situation, where all the nodes at the network want to go in communication, with each other. Flat design approach facilitates to work in all the three scenarios. In flat design, there is no root and no leaf. RMTP again does not fit in this scenario because of its hierarchical design. Since as the number of senders will start the communication, simultaneously, then a dilemma will occur about, who will be the root of that hierarchy.

[2] Best Path:
RSM2 provides best path for the packet delivery. RSM2 use Kruskal's Algorithm for minimum spanning path, to reach to the all presentable nodes on the network. This path is further reduced if it includes some undesirable nodes. So, the data-packet is transmitted over short path, with minimum cost .But, in RMTP these concepts are not introduced. In RMTP, data is just multicasted without concerning cost or delay of the links, and the responsibility of sender is transferred to DR, i.e. Designated Receiver.

[3] Dynamics Co-operativity:
RSM2 model is designed in such a way, to work in dynamics also. RSM2 works well in Wired Networks and in Infrastructure Wireless Networks also. Working of RSM2 in Infrastructure - less Wireless Networks is the part of future scope. Hence, RSM2 is capable to co-operate dynamics well. RMTP works well in static environment only and do not cooperate well with dynamics of nodes. Since, in RMTP designated receivers are chosen statically, based on approximate location of receivers. So it halts in wireless networks. Hence, RMTP does not provide Dynamic Cooperativity.

[4] Combo-Casting:

RSM2 use Combo-Casting to serve NACKs hence reduces duplication of packets effectively. Dynamic Managers first stores all the NACKs, and wait for a random amount of time, $T_{NACK.}$ As, the time gets out DMs decide either to retransmit the data through multicasting or via unicasting. If more NACKs are received for the same packet then DM subgroups them into a new group and multicast the missed data-packet. In case, there is one or two nodes send NACK for the same packet, then, DMs unicast the missed data packet.

In RMTP, Designated Receiver buffers the packet and if NACK comes for a missing packet, then it is first served by the DR, who is looking after that area .If this DR, becomes unable to serve that request, and then request will go to its Parent DR and so on. Finally reach to the sender, if no DR serves that request.

[5] Reliability & Scalability:
The objective of our Model-RSM2 in this thesis is to guarantee *reliability* achieving high throughput, maintaining low end-to-end delay. This is achieved by reducing unnecessary retransmissions by the sender.

[6] Load on Sender:
RSM2 put fewer loads on Sender, because the load has been shifted from Sender node to Dynamic Managers.

[7] DMs Vs. DRs:
DMs have high network-computations ability for all network computations like – to create the priority matrix and updates the matrix, whenever the node topology changes.

[8] Retransmissions:
In RSM2 model, we use Combo-casting technique for the retransmission purpose, to deal with lost-packets.

[9] Complexity:
Complexity of RSM2 is O (nxn) but Complexity of RMTP is O (n).
Complexity of RSM2 > Complexity of RMTP
$O(n \times n) > O(n)$

It is concluded that RSM2 can be implemented in wireless i.e., Infrastuctured Network, unlike RMTP, if capability and computability of RSM2 is applied over Base-station. Along with it, RSM2 works easily in dynamic environment where more than one sender sends data simultaneously, unlike RMTP. A node can act as Sender and Receiver both at the same time simultaneously, unlike RMTP. If one node leaves the topology, then DMs update its Partial matrix and forwards this information to the sender and other nodes periodically. Hence, RSM2 can also be applied to hybrid networks also. This is the part for future research.

Limitation of this Model is that, it's set-up cost is greater and cannot be established for Infrastructure less (adhoc) Networks. As the number of users increases, the cost to implement this model also get increases.

VII. Mcpa: Implementation Of Rsm2

In this chapter we have some snapshots that show the implementation results of MCPA algorithm, which has been designed to describe RSM2. For the implementation of MCPA algorithm, we opt 'C' as coding language.

The description of each snapshot is given below:

- Snapshot 1: This snapshot has taken, when the topology is decided by the user and she assigns the cost to each link of the same network. It is depicted in Figure 1 and Figure 2, deciding the topology of network and assigning the cost by User.

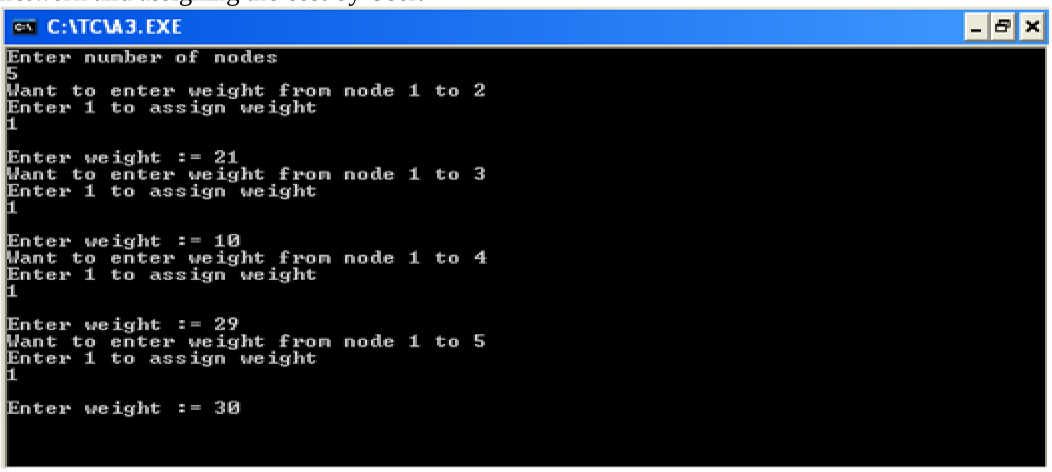
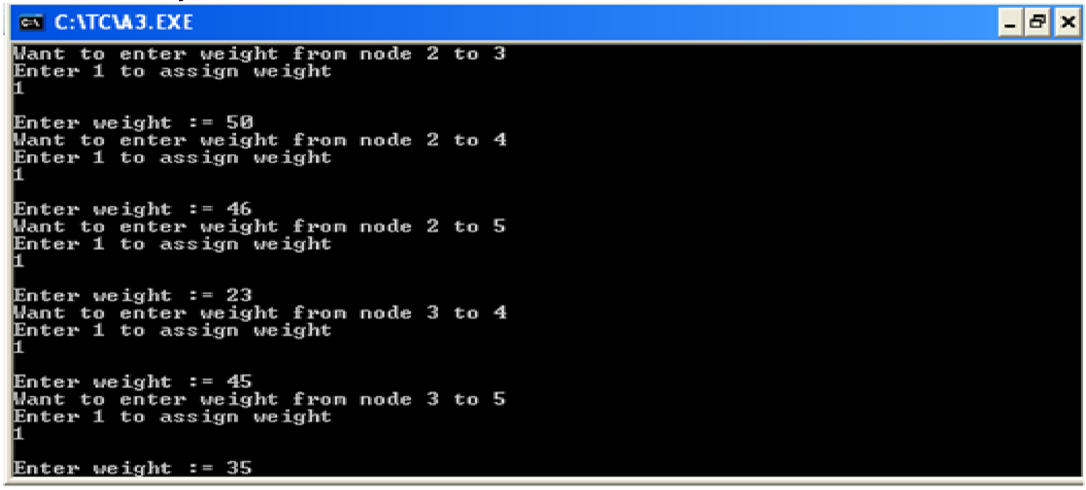


Fig.1: Deciding the topology of network and assigning the cost by User.

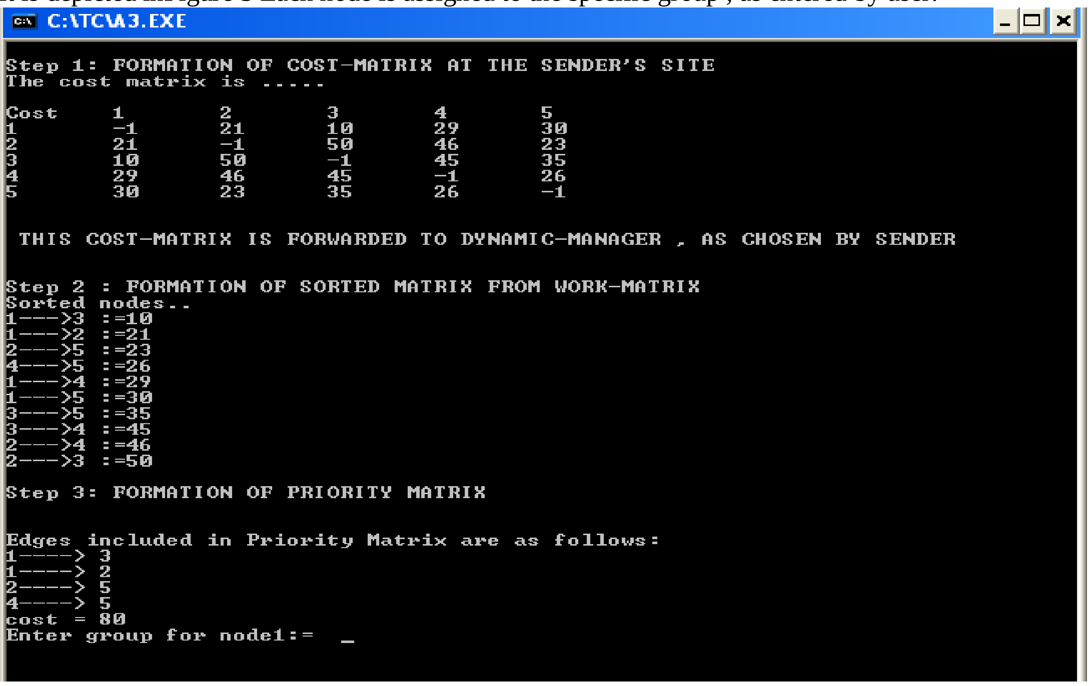
- Snapshot 2: This snapshot has taken, when the main steps of the algorithm comes into action. It shows the formation of Cost-Matrix, Sorted Work Matrix and Priority Matrix using the data that has been entered by the user in Snapshot 1. It is depicted in Figure 2, Formation of Cost-Matrix, Sorted Work Matrix & Priority Matrix.



```
C:\TC\A3.EXE
Want to enter weight from node 2 to 3
Enter 1 to assign weight
1

Enter weight := 50
Want to enter weight from node 2 to 4
Enter 1 to assign weight
1

Enter weight := 46
Want to enter weight from node 2 to 5
Enter 1 to assign weight
1

Enter weight := 23
Want to enter weight from node 3 to 4
Enter 1 to assign weight
1

Enter weight := 45
Want to enter weight from node 3 to 5
Enter 1 to assign weight
1

Enter weight := 35
```

Fig 2: Deciding the topology of network and assigning the cost by User.

- Snapshot 3: This snapshot has taken, when the user decides which group to be allocated to each node. It is depicted inFigure 3 Each node is assigned to the specific group , as entered by user.



```
C:\TC\A3.EXE

Step 1: FORMATION OF COST-MATRIX AT THE SENDER'S SITE
The cost matrix is .....

Cost    1       2       3       4       5
1       -1      21      10      29      30
2       21      -1      50      46      23
3       10      50      -1      45      35
4       29      46      45      -1      26
5       30      23      35      26      -1

 THIS COST-MATRIX IS FORWARDED TO DYNAMIC-MANAGER , AS CHOSEN BY SENDER

Step 2 : FORMATION OF SORTED MATRIX FROM WORK-MATRIX
Sorted nodes..
1--->3 :=10
1--->2 :=21
2--->5 :=23
4--->5 :=26
1--->4 :=29
1--->5 :=30
3--->5 :=35
3--->4 :=45
2--->4 :=46
2--->3 :=50

Step 3: FORMATION OF PRIORITY MATRIX

Edges included in Priority Matrix are as follows:
1----> 3
1----> 2
2----> 5
4----> 5
cost = 80
Enter group for node1:=  _
```

Fig 3: Formation of Cost-Matrix , Sorted Work Matrix & Priority Matrix.

- Snapshot 4: This snapshot has taken, when the user has been asked to decide the sender node. In RSM2 model, we can have more than one node as Sender(s). Along with it, user tells with how many groups, he wants to communicate as receiver(s). It is depicted in Figure 4, Sender Node and Receiver group is entered by user.

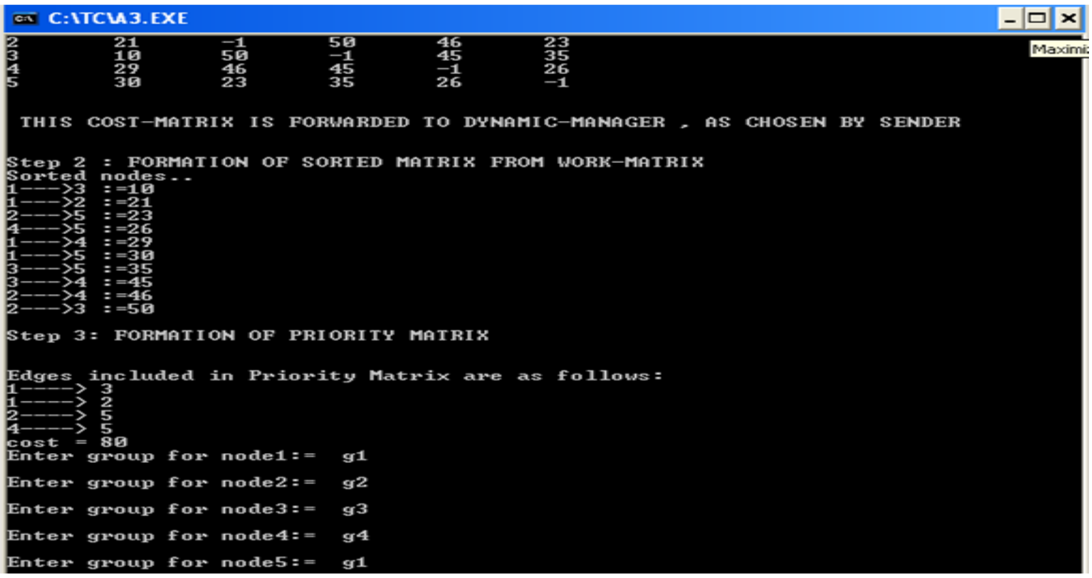


Fig 1: Each node is assigned to the specific group , as entered by user

- Snapshot 5: This snapshot has taken, when Sender finds out the Priority Matrix and uses that path to send the data. It is depicted in Figure 5 .Senders(S) use Priority Matrix to send the Data to Desirable Receivers(R)

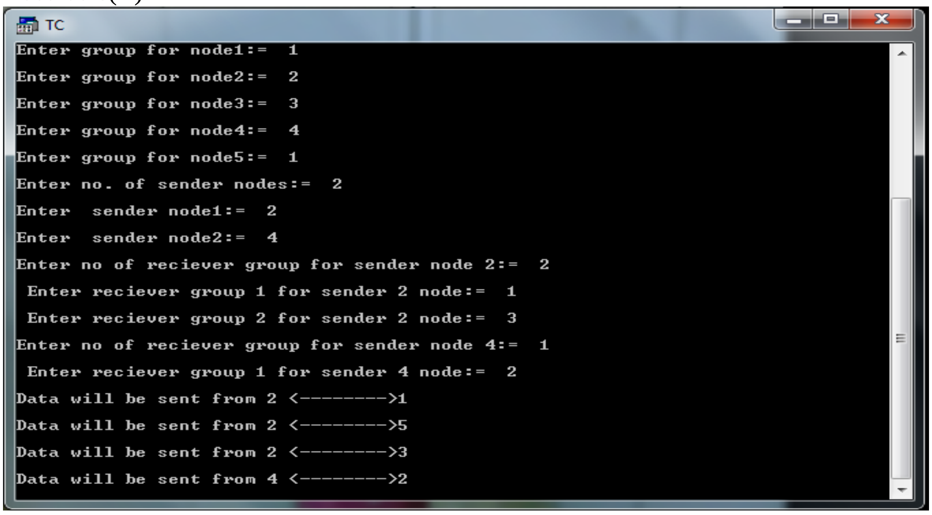


Fig 5: Sender Node and Receiver group is entered by user

- Snapshot 6: This snapshot shows Senders(S) use Priority Matrix to send the Data to Desirable Receivers(R).

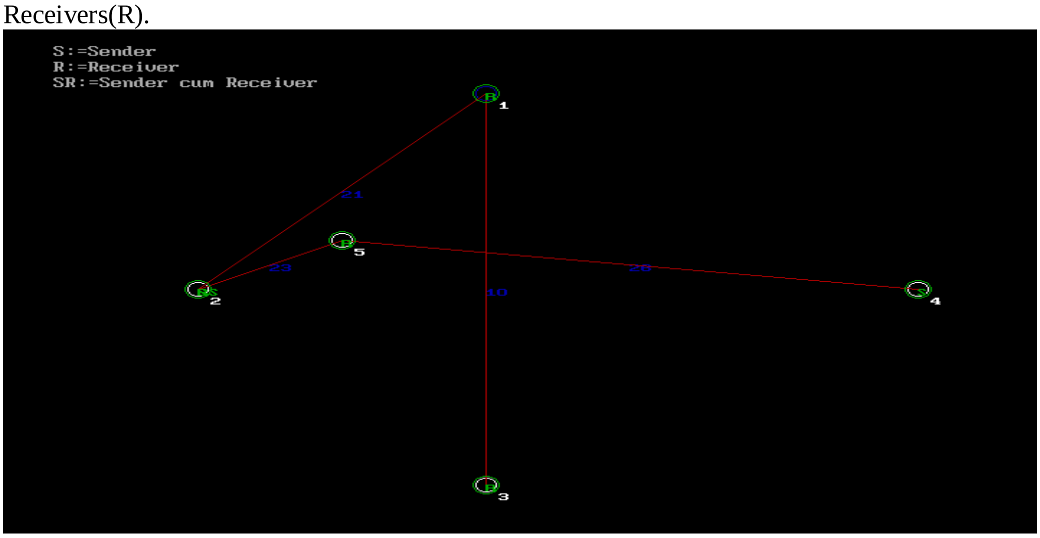


Fig 6: Senders(S) use Priority Matrix to send the Data to Desirable Receivers(R)

ACKNOWLEDGEMENTS

MR. PRAMOD KUMAR SETHY, ASSISTANT PROFESSOR, K.E.C, I would like to express my best and special thanks for all of your support.
SANJIV KUMAR, an Engineering Scholar, Mewar University, Rajasthan. I would like to express my best and special thanks to you, my younger brother , for your support and a lovely gift.
MR. RAHUL PRAKASH, Assistant Professor, CSE Dept., Sri Venkateshwara University, Gajraula, U.P, for your constant support, understanding, help, patience, care and helpful discussions.
Words are insufficient to express my profound sense of gratitude to my parents & friends, whose encouragement & blessings gave me great physical & moral strength.

REFERENCES

Journal Papers:

[1] Sanjoy Paul, Member, IEEE, Krishna K. Sabnani, Fellow, IEEE, John C.-H. Lin and Supratik Bhattacharyya "Reliable Multicast Transport Protocol (RMTP)". *IEEE journal on Selected Areas in Communications*, Vol. 15, No. 3, April 1997.

Proceedings Papers:

[2] Danyang Zhang, Sibabrata Ray, Rajgopal Kannan, S. Sitharama Iyengar "A Recovery Algorithm for Reliable Multicasting in reliable networks." Proceedings of the 2003 *International Conference on Parallel Processing* (ICPP'03).

Books:

[3] K.K.Singh, Akansha Singh,"Mobile Computing", *Umesh Publications.*

Thesis:

[4] Ruchi Gupta, A Reliable and Scalable Multicast Model (RSM2) ,Mahamaya Technical University, Noida, M.Tech, CSE, 2012.

Journal Papers:

[5] Carolos Livadas, Idit Keidar, Nancy A. Lynch "Designing a Caching-Based Reliable Multicast Protocol"
[6] Ruchi Gupta, Pramod Kumar Sethy "A Reliable & Scalable Multicast Model"(RSM2), in International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-1, Issue-2.