© 2016 IJSRST | Volume 2 | Issue 2 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Feasibility Optimal Broadcasting Policy Framework in Wireless Network

Gokul G, Baskaran T, Ramakrishnan R

Department of Information Technology, Dhanalakshmi College of Engineering, Manimangalam, Tambaram, Chennai, Tamil Nadu, India

ABSTRACT

This project aims to deal with less reliability of nodes in a wireless network, and study behaviour of each nodes in a dynamic fashion by splitting into pre-defined number of packets and attaching an information with each packet to study the link quality once it reaches the destination, how the data is broadcasted in the network and uses a storage for packet information and once there is a node failure it studies why the failure occurs if due to a technical failure or incapacity of sever to process the request or the client to receive the data it is analysed and resend or if a deliberate rejection from client or some middle man acquiring packets or destination ip and the receivers ip don't match it drops the node and uses optimal grouping to decide the next best linking if a node is dropped to achieve the best throughput rather than a random selection of link and in turn making it very inefficient even for a small round-trip.

Keywords: Multicast, Network framework, Policies

I. INTRODUCTION

For so long the wireless or wired networks have used ping, ICMP messages to test whether the node is active or dead. Whether there has been a link failure or node failure. The demands of clients are specified by allowing each client to require a minimum of throughput for each flow. Furthermore, the wireless network is modelled as one where wireless links are unreliable.

It is a vague process to calculate each path and its efficiency every time and broadcast patterns. Therefore, it is easier to maintain real time efficiency information with link quality in a database so that server studies links. Suppose a failure occurs then the server can analyse why the error has occurred is it due to incapability or due to intentional rejection. If the server finds that node intentionally rejects packets then it drops the node from the network else the link will be repaired. Making trip times lesser and reliable connection so user has less chances of data loss.

II. METHODS AND MATERIAL

Experimental Study

The design is based on Broadcast delay constrained traffic over unreliable wireless links with network coding. This project is designed to be used by server system to study the reliability of the links.

The Registration processes for each node are done through a java JFrame interface, which connect nodes based on socket and port calls. The node program runs on various clients and connects via a main server program. The nodes are connected via links based on user decision and costs are assigned to each link.

Once the computers are connected a register page appears which prompts the user to register his/her system name and password. Once user logs in with the created name and password there is an option to link various nodes and assign the necessary cost to each of

the link created. Here Greedy algorithm is used to construct the graph and search for a efficient route. Once the server identifies a rejection or inability of link to send data it verifies the database. The database contains efficiency information and link quality based on the feedback information received from each packet sent by receiving node. This data is used to analyse and decide whether the link has efficient communication.

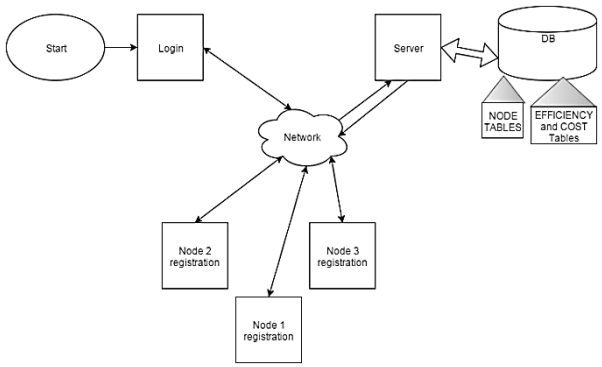

Figure 1. Architecture Diagram

III. RESULT AND DISCUSSION

Experimental Results

The first UI is a node monitoring server node that runs in the background and acts as socket server. The second swing activity of the project is a validation screen that does background initialization of other activities and runs authentication process that checks user detail with existing database storage or the user (node) can register by giving a node name and password. After the login screen the server is moved to background process which runs all the time to monitor client activities. The client process is a separate swing ui that lets user to select their data and number of packets it wants to send the data with then the content is encrypted with an AES key then the respective node destination is selected and broadcasted, the Gkey for the node is shared with the receiving node. Then the same way the receiving node then decrypts the data. This transfer is closely analysed by the server process. When there is a link failure in the network the server analyses the failure. If it is due to an inability of the client to receive the message then it waits till the buffer is free. Else the server drops the node from the network and uses Optimal Grouping algorithm to reconstruct the connection of the network. All these efficiency details are stored in a database and analysed by server in a background state for a reliable communication.

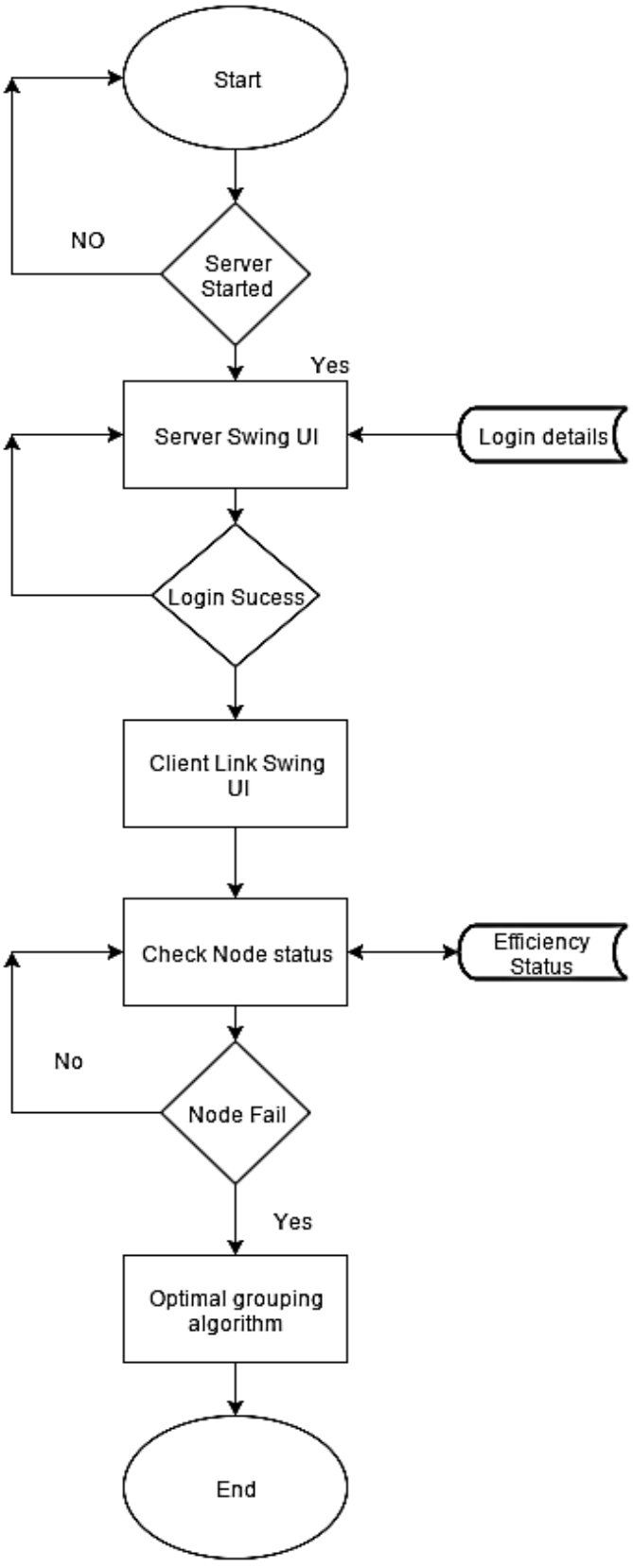

Figure 2. Flowchart

Validation is done by checking the login for characters and the password checked. Then it is verified with remote database. If the response is null, then a "no record found" is displayed. Alternatively, the "credentials invalid" is returned if invalid response is received. For the correct details, the node details and UI is displayed for the user to select files and send it using a key. The server contains all the node details, which is stored in the database. Nodes are viewed or added in the main screen.

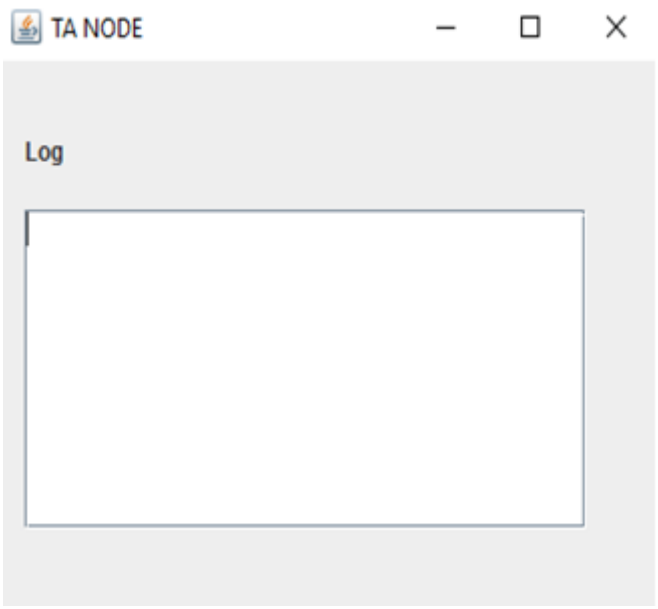

Figure 3. A Server Node

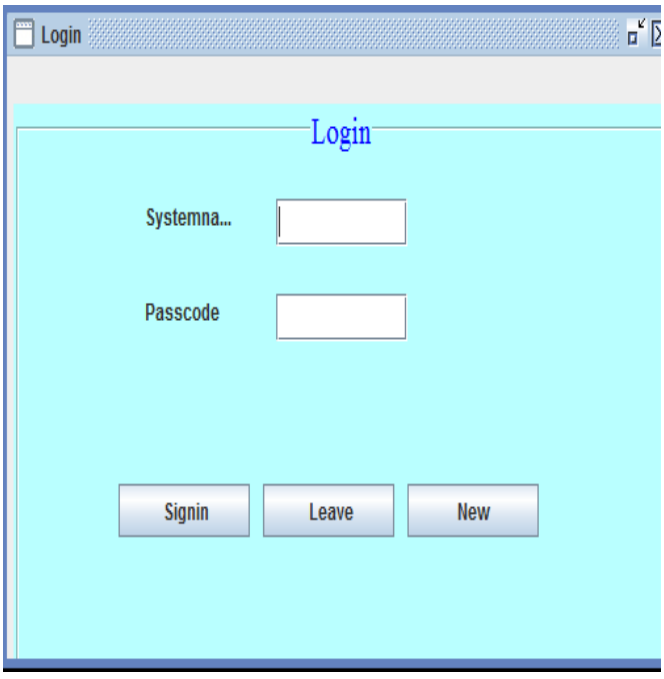

Figure 3. B Login UI

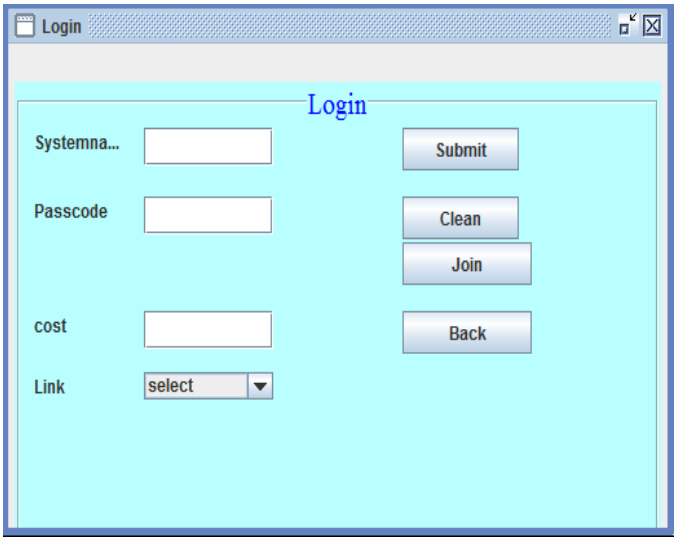

Figure 3. C New User UI

The user can add nodes and register them into the network and form links with the other computers as required there is no reverting mechanism for a client once registered if it wants to get out of the network it has to stop getting data from other nodes so that server identifies and stops the node and regroups the other nodes as a reliable link network. While joining the network user us asked a system name, passcode, cost of the link, the link of the system to other node then the join button is pressed and then submits. Then the user can go back to login and use the ckiebt side UI to send and receive data. While sending a data file to a client or receive a data from some node the AES key is asked as a prompt only after verifying the AES key the data is decrypted and displayed to the user.

IV. CONCLUSION

The number of unreliable connection in any place is undesirable. The freedom in using the wireless network is very much limited by the drawbacks of timeout and link failure and results in dissatisfaction and high failure rates. So this scheme of monitoring and controlling a registered node in a network. Smaller reasons for failure like a timeout or buffer overflow can be separated because it is analysed. So if a node deliberately stops a data transfer then it can be dropped. Which reduces dropping of a working node without any warning?

This can lead to a desirable throughput rate without any packet loss and manual selection of number of packets. And maybe lead to a practical use of WLAN and Wireless network in the future. And usage in networking devices and towers to regulate flow of information through a particular node and particular path instead of brute force pushing of data through a shortest path algorithm or best throughput route an alternative can be achieved to implement the flow of packets as to maintain a reliable link and reduce data loss and jitter in the network during crucial time and when necessity arises.

V. FUTURE ENHANCEMENTS

This is just an initial stage and focuses in implementing the core concepts. This system can be integrated into networking devices to balance load and even maintain a whole of network traffic between nodes and even between subnets so if a subnet fails there might be a different route to reach another part of data quickly and recover the network.

Since the project is in initial stage the linking and server process is a manual work. This can be made as a service and run alongside windows and just a simpler linking of nodes can be done by reducing the complex manual registering UI to simpler UI. By atomizing most of background details like selection of AES key and giving user more options of encryptions.

VI. REFERENCES

[1] M. Ghaderi, D. Towsley, and J. Kurose, "Reliability gain of network coding in lossy wireless networks," in Proc. IEEE INFOCOM, 2008, pp. 2171–2179.

[2] H. Gangammanavar and A. Eryilmaz, "Dynamic coding and rate-control for serving deadline-constrained traffic over fading channels" in Proc. IEEE ISIT, 2010, pp. 1788–1792.

[3] X. Li, C.-C. Wang, and X. Lin, "Throughput and delay analysis on uncoded and coded wireless broadcast with hard deadline constraints," in Proc. IEEE INFOCOM, 2010, pp. 1–5.

[4] W. Pu, C. Luo, F. Wu, and C. W. Chen, "QoS-driven network coded wireless multicast," IEEE Trans. Wireless Commun., vol. 8, no. 11, pp. 5662–5670, Nov. 2009.

[5] I.-H. Hou, V. Borkar, and P. Kumar, "A theory of QoS in wireless," in Proc. IEEE INFOCOM, 2009, pp. 486–494.

[6] I.-H. Hou and P. Kumar, "Admission control and scheduling for QoS guarantees for variable-bit-rate applications on wireless channels," in Proc. ACM MobiHoc, 2009, pp. 175–184.

[7] I.-H. Hou and P. Kumar, "Scheduling heterogeneous real-time trafficover fading wireless channels," in Proc. IEEE INFOCOM, 2010, pp.1–9.