

Traffic free distributed interactive applications in client assignment problem

B. Senthil kumar¹, N. Uma Maheshwari²

Assistant Professor, Department of Computer Science, Sree Narayana Guru College, Coimbatore, Tamil Nadu¹

M.Phil. Scholar, Department of Computer Science, Sree Narayana Guru College, Coimbatore, Tamil Nadu²

Abstract: Inter server communication is most common word for the distributed system, it provide an interactive performance among the system. With the modern internet facility connection of server or interconnected networks is used for share the resources. An ideal system in the distributed architecture provide equal responsibility and computational power but an engaged system it fails to provide that facility among the distributed network and it also provide a load balancing problem when two or more packet send at the same time and some of the data were loosed because of collusion. For this issue many researchers try to balance the load of the server, but it improved in slight level only and if sequence of data occurred it may cause congestion problem. To recover the problem here we propose a Heuristic Algorithm (HA) to fully solve the load balancing and traffic avoidance problem and we also focus on security problem with the help of DSA Algorithm.

Keywords: Distributed System, DIA, CAP, Heuristic Algorithm, Load Balancing, Traffic Control and Security.

1. INTRODUCTION

In this modern world people were connected with the internet with the connection of network for the sharing resources. A network is a connection of nodes interconnected with each other to distribute the resource and computational power among the nodes, typically called Distributed System (DS) [1]. This system contains servers and clients, hence servers are more computational and resource powerful than clients. Examples of DS are E-Mail, e-commerce, instant messaging, social networks etc. With this distribution of the networks, it shares the resources among the nodes [2]. While users communication, it have many difficulties which includes overhead of transmission among the nodes which leads to lower performance. Communication patterns among the client and server with congestion less is a major factor among the users, which is related to Client Server assignment problem [3, 4]. It is relevant one for the host of applications from above said DS. Continuous sharing of resources, Exchanging of resources will reduces the inter-server communication lead a congestion problem and have to balance the communication load. It is the example of optimal client server assignment problem. For this reason distributed interactive applications (DIAs) were user [5]. DIAs is networked systems that allow multiple participants at different locations to interact with each other. DIAs often requires geographical distribution of servers to meet the latency requirements of the applications [6].

Wide spread of client locations in large-scale DIAs often requires geographical distribution of servers to meet the latency requirements of the applications. In the distributed server architecture, client assignment to servers directly affects the network latency involved in the interactions between clients [7]. Many researchers were focuses on client assignment problem (CAP) for enhancing the interactivity performance of DIAs.

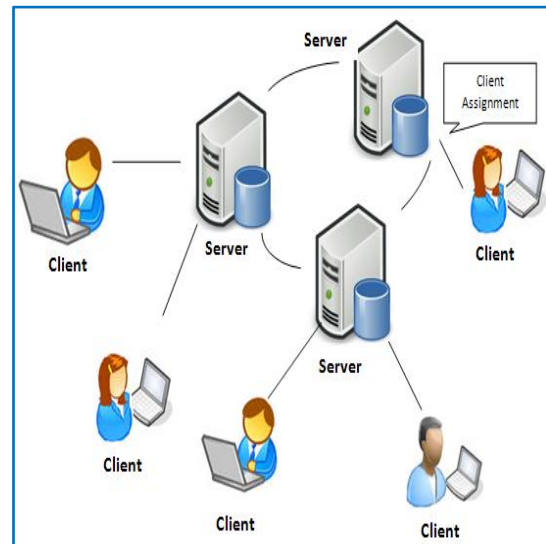


Fig1: Architecture of Distributed servers

While in Client server assignment they modeled on the fact that the delay incurred by a client is dependent on load of the server and the distance to the assigned server [8]. Delay on the client side is the sum of network delay (proportional to distance to its server) and congestion delay at the server. Hence it focuses on distance between the client and the server and the load on the server in client-server assignment. The problem is defined as Load-distance balancing (LDB) problem. For this issue this paper uses a Distributed Load Balancing Algorithm (DLB). And also have an assignment problem between the client and server. To solve this client-server assignment problem based on total communication load and security of the resources here the use of heuristic algorithm with the combination of Greedy Assignment to assign clients iteratively, Distributed greedy assignment for distribute global knowledge and optical location allotment algorithm to minimize the average incurred delay.

2. LITERATURE REVIEW

From the author Umar Farooq, John Glauert [9] presents their proposed hierarchical infrastructure called JoHNUM for the improvement of scalable and reliable distribution of data. Their method is used to estimate the number of different parameters in the networks. And also they additionally added the load distribution algorithm based on aggregation of regions, to provide a better reliable one to the distributed network for load while acquire normal and contiguous areas for assignments. Their proposed method greatly reduces communication and computational cost. They concentrate the area in the virtual environment (VE). Their infrastructure comprises of three components: Partitioning, Assignment and Merging. This is used to generate and exceed the capacity of regional server to provide a Maximum Server Capacity (MSC). Assignment is used to aggregation regions at each level against the MSC and tries to reach as balanced the load as possible. Merging is used to server generating and merging when it observes a decrease in the assigned player's density. The evaluation results demonstrate that it works well with small, medium, and large scale environments.

Hanying Zheng and Xueyan Tang [11] studied the problem of Distributed Interactive Applications and proposed an efficient greedy server provisioning heuristic method. Their technique is used to formulate the server provisioning problem with an objective of reducing the network latency involved in the interaction between participants. Their proposed GREEDY algorithm substantially outperforms traditional k -median and k -center server placements. Each iteration, the k median heuristic adds a new server location that results in the largest reduction in the total latency from the clients to their nearest servers. Similarly, the k -center heuristic adds a new server location that results in the lowest value of the maximum latency from the clients to their nearest servers. From the computability perspective, the mutual interaction feature also makes server provisioning for DIAs much more challenging than that for web content delivery. With the client-to-server latency as the sole optimization objective, placing more servers in the network can only improve the web access performance.

Swathi Balakrishna and Ling Ding [13] propose an algorithm that is based on Semi-definite programming, to obtain an approximately optimal solution for the client-server assignment problem. They analyze the client-server assignment important role in network and propose the programming. Their approach is mostly reliant on two significant factors, 1) Total communication load and 2) Load balancing of the servers. In their approach they propose an algorithm which is based on Semi-definite programming, to gain an estimated optimal solution for the client-server assignment problem [14]. Semi-definite programming is a sub field of convex optimization. Semi-definite programming has recently emerged to prominence because it admits a new class of problem previously unsolvable by convex optimization techniques. They show the efficiency approach by the simulation experiments compared with the existing client-server assignment.

Delaney, et al [15] study the processing power at contributing nodes and the accessibility of greater communications bandwidth, the essential restriction to the use of Distributed Interactive Applications (DIAs) is the problem of maintaining a reliable worldview in the occurrence of latency and jitter. They study the link between constancy and a number of portions of DIAs such as temporal and spatial synchronisation, causality or arranging and concurrency was discovered. The link with reaction and reliability was also discovered, in that reduced reliability can direct to problems of fidelity, and fidelity is often forfeit to preserve both consistency and responsiveness. To evade discrepancy such as discrepancy, causality violation or expectation violation, different techniques and technique have been proposed and implemented over the years. They have submitted to any element employed to make sure an adequate, identical dynamic shared state for all applicant in a DIA as a reliability maintenance mechanism. Consistency control mechanisms can be classified as either pessimistic or optimistic [16, 17]. *Pessimistic* mechanisms prevent concurrent operations and allow only a single participant to issue state changes at a certain point in time. This can be done by a locking or a floor control algorithm where exclusive access rights have to be obtained before the participant is allowed to change the state of an object. This approach is used by the collaborative virtual environment MASSIVE-3. Pessimistic approaches therefore reduce the consistency control challenge to the ordering of operations that origin from a single source.

3. PROBLEM DEFINITION

The problem in the existing system that where a critical element of CDN architecture is the request routing mechanism. It permits to direct users' requirements for content to the appropriate server based on a specified set of parameters. The closeness principle, by means of which a request is always provided by the server that is contiguous to the client, can occasionally fail [18, 19]. Certainly, the routing method linked with a request might take into account several parameters (like traffic while sending the packets, bandwidth, and servers' computational capability) in order to supply the best routine in terms of time of service, delay, etc. Additionally, an effective request routing method should be able to face temporary, and potentially localized, high request rates (the so-called *flash crowds*) in regulate to evade affecting the quality of service perceived by other users [20].

4. PROPOSED WORK

To solve the aforementioned problems, our system proposed a client-server assignment problem based on total communication load and security of the resources with the help of heuristic algorithm for the possible optimal solution. For the optimal solution, here the combination of Greedy Assignment to assign clients iteratively, Distributed greedy assignment for distributes global knowledge and optical location allotment algorithm to minimize the average incurred delay and also focused

on security problem with the help of DSA Algorithm. A heuristic method is used for making the decision at each step, to find the best method. For end-to-end transfer or transfer among the nodes they were traffic occur because of sequence of data transfer from one node to another. For free the traffic and to send the packets here we propose a Greedy Assignment method is used to select the candidate service node with the major “potential” value. When a node with the largest “potential” value is selected, the “potential” value of next candidate service nodes and the current gathered connection possibility of each client node are updated this technique were used to free the traffic among the nodes. For distributing the packets among the network when the selected node is busy means it alternatively chooses the other node with the help of our Distributed greedy assignment technique. Significantly outperform the intuitive Nearest-Server by using this technique. And finally we propose an optical location allotment algorithm to avoid or minimize the deserved delay among the network. And also for the security here we propose a DSA Algorithm. Digital Signatures Algorithm is used to detect unauthorized modifications to data and to authenticate the identity of the signatory.

4.1 Heuristic algorithm

Heuristic algorithm is used to discover an optimal solution for transferring the data. This method is used to progress efficiency of client server assignment. Heuristic algorithms are used in distribution system for allocating server, used to solve the NP hard problems. This algorithm provides complete an optimal, accurate and complete solution. The client assignment problem is NP-complete; there is no polynomial time to find the optimal client server assignment. In existing approach they where lacks to discover the optimal solution in responsible time even for small number of clients and servers. Moreover, servers can fail and may not be able to respond the incoming client requests. For this reason, heuristic algorithms are preferred to find near optimal solutions to the client assignment problem.

4.2 Distributed Load Balancing Algorithm

In network, more number of packets is sending from client to server and server to client it may arise delay in transmission because of over load of data among the nodes. Delay on the client side is the sum of network delay and congestion delay at the server. Here the DLB algorithm used based on two conditions (i) least connection and (ii) least response time. In the first condition selection of sever based on which server currently has a least client connection and the second condition is based on the selecting server based on server quick response time. By using this method congestion free transmission will achieve for sending the packets. With the help of DLB algorithm it minimizes the maximum incurred delay using an approximation and optimal algorithm. It minimizes the communication load between servers and it sends the packets in the manner of congestion free transmission.

4.3 Distributed Load Balancing Algorithm

```

// peer status update
prob_space[0]=0; load_diff = 0; load_diff_sum = 0;
for(j=1; j<=n; j++){
    if(load_i - peer[j].load){
        load_diff = load_i - peer[j].load;
        //insert the new difference
        build_prob_space(load_diff, prob_space);
        load_diff_sum = load_diff_sum + load_diff;
    }
    //normalize the vector elements
    update_prob_space(load_diff_sum, prob_space);
}

// balancing process
if(prob_space[] == NULL) //no neighbors with lower load
    //serve locally the request
    serve_request();
else{
    float x = rand(); //random number generator
    int req_sent = 0; int i = 0;
    while(prob_space[i] == 1 or req_sent == 1){
        if(prob_space[i-1] <= x < prob_space[i]){
            //send request to the chosen peer
            send_to(peer[i-1].addr);
            req_sent = 1;
        }
        i++;
    }
}

```

4.4 Distributed greedy assignment technique

From the distribution network transferring of data is available at very frequent situation. In this situation sequence of data transfer may cause the packet loss, congestion, jamming data, etc. to solve these issues our proposed technique were used to select the nearest server to transfer the data from one node to another when the node were busy it transfer to the next nearest node based on our Distributed greedy assignment technique. The Fig 2 shows the clear view of the transferring data in the busy server architecture.

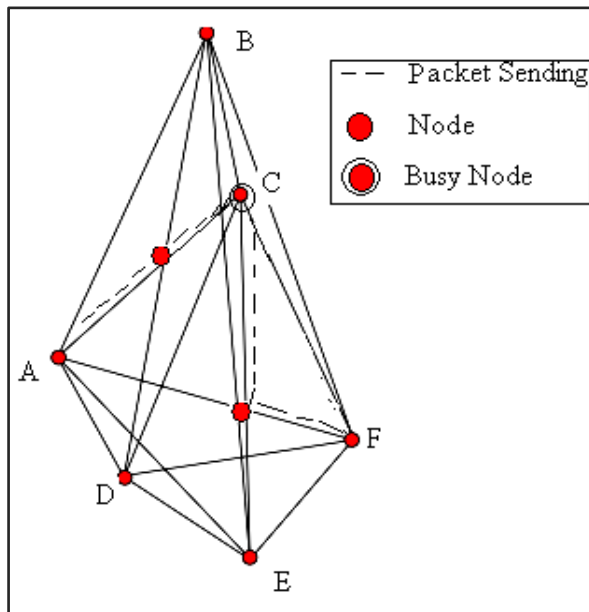


Fig 2: Selecting of nearest node

From the above diagram it shows that when a data is transfer from one node to another if the selected node is busy in the network means it automatically send the data to the nearest node and it pass to the designation node without loss of data. Here it avoids the congestion in the network based on our Distributed greedy assignment technique. Our proposed method proves better result when compared to the existence one.

5. METHODOLOGY

5.1 Content Delivery Network

The consistency requirement for continuous DIAs is to ensure that all clients share the same view of the application state when their respective simulation times reach the same value. This is automatically guaranteed among the clients assigned to the same server because they all inherit the application state from their assigned server through state updates. Nevertheless, the application states seen by the clients assigned to different servers may not be identical at the same simulation time if the application states maintained by their assigned servers are not consistent. Since the state of a continuous DIA changes due to both user operations and time passing, to ensure consistency among the application states at the servers, each user operation must be executed by all servers at the same simulation time.

5.2 Distributed-Modify Assignment

Distributed-Modify Assignment is performed in a distributed manner without requiring the global knowledge of the network at any single server. It starts with an initial assignment. Then, the assignment is continuously modified for reducing the maximum interaction path length D until it cannot be further reduced. This process is referred to as the assignment modification. One server is elected as a coordinator responsible for calculating D and selecting the server to execute the transfer modification. To compute D of the primary assignment, each server measures its distance which is said to be network latencies

to all the other servers. It also computes its distances to all the clients that are allocated to it and preserve them as a sorted list. Then, each server s broadcasts to all the other servers its longest distance $l(s)$ to its clients, and sends the inter server distances to the coordinator. The coordinator calculates D based on the received information.

5.3 Transaction-Least-Work-Left

The Transaction-Least-Work-Left (TLWL) algorithm addresses load balancing issue by assigning different weights to different transactions based on their relative costs. Counters are preserved by the load balancer representing the weighted number of transactions allotting to each server. Here in the transaction new calls are allocated to the server with the lowest contradictor. TLWL guess the server load which is based on the weighted number of communication among the server which server is currently handling the data. TLWL can be personalized to workloads with other transaction types by using dissimilar loads based on the overheads of the transaction types. In addition, the comparative costs used for TLWL could be adaptively varied to progress the performance of network.

5.4 Load balancer

In the network, client sends the data which is said to be parser which is parsed the request to the designation node with the help of session recognition were implemented. This determines if the request communicate to an already accessible session by querying the Session State. If it happens, the request is forwarded to the server to which the session was previously assigned. If it not exists, our Server Selection module assigns the new session to a server using TLWL algorithm. For numerous of the load balancing algorithms, this effort may be depend on Load Estimates, it preserves the data for each of the servers. The Sender one is it forwards requests to servers and updates Load Estimates and Session State as needed. The Receiver also receives responses sent by servers. The client to receive the response is recognized by the Session Recognition module which acquires this information by querying the Session State. The Sender then sends the response to the client and updates Load Estimates and Session State as needed. The Trigger module updates Session State and Load Estimates after a session has expired.

5.5 Attackers and Security

We believe one attacker in the area, which is not authenticated and linked with the network. The purpose of the jammer is to corrupt communication of rightful nodes by affecting packet collisions at receivers. Intended collision leads to retransmission and thus further energy consumption for a confident amount of throughput, or equivalently compact throughput for a given amount of consumed energy. We studied convenient jamming attacks in this distributed network, which are easy to commence and complicated to notice and tackle. The derived solutions to the optimization problems speak optimal attack and network security strategies. For this

issue we additionally propose a DSA Algorithm for the secure transmission of data in the distributed systems. It is used to detect the modification of data from the unauthorized person based on the identity of the signatory. These above methods were performed in the different parameter based on their network nodes to provide a better transmission among the nodes.

6. CONCLUSION

In this paper we analyze the problem of DIA with the Client Assignment Problem and the data transmission from the node may contain the problem of Load balancing, congestion, jamming, packet loss, and delay transmission from one node to another. To solve these problems we propose a heuristic algorithm to avoid the above mentioned problems. With the help of HA the we provide the three combinations they were Greedy Assignment to assign clients iteratively, Distributed greedy assignment for distribute global knowledge and optical location allotment algorithm to minimize the average incurred delay in the network. Additionally we also focus on the security of the network with the help of DSA Algorithm. It is used to detect unauthorized modifications to data and to authenticate the identity of the signatory.

REFERENCES

- [1]. J.C.S. Lui and M.F. Chan, —An Efficient Partitioning Algorithm for Distributed Virtual Environment Systems,|| IEEE Trans.Parallel and Distributed Systems, vol. 13, no. 3, pp. 193-211, Mar .
- [2]. Greenberg, S. and Marwood, D. *Real Time Groupware as a Distributed System: Concurrency Control and its Effect on the Interface*. In: Proc. ACM CSCW, Chapel Hill, NC, USA, pages 207–217, October 1994.
- [3]. Patil, Bharati, and S. B. Patil. "A Novel Approach on Client Server Assignment Problem in Distributed System"
- [4]. Zhang, L.; Tang, X.; , "Optimizing Client Assignment for Enhancing Interactivity in Distributed Interactive Applications," Networking,IEEE/ACM Transactions on , vol.PP, no.99, pp.1, 0doi: 10.1109/TNET.2012.2187674.
- [5]. Lu Zhang; Xueyan Tang; , "Client assignment for improving interactivity in distributed interactive applications," INFOCOM, 2011Proceedings IEEE , vol., no., pp.3227-3235, 10-15 April 2011doi: 10.1109/INFCOM.2011.5935173
- [6]. Ucar, Seyhan, Huseyin Guler, and Ozgur Ozkasap. "Online Client Assignment in Dynamic Real-Time Distributed Interactive Applications" *Distributed Simulation and Real Time Applications (DS-RT), 2013 IEEE/ACM 17th International Symposium on*, IEEE, 2013.
- [7]. Nguyen, Cong Duc, Farzad Safaei, and Paul Boustead. Distributed server architecture for providing immersive audio communication to massively multiplayer online games." *Networks, 2004.(ICON 2004), Proceedings. 12th IEEE International Conference on*. Vol. 1. IEEE, 2004.
- [8]. Hiroshi Nishida, Member, IEEE, and Tinh Nguyen, Member, IEEE—Optimal Client-Server Assignment for Internet Distributed Systems||-ieee transactions on parallel and distributed systems, vol. 24, no.3, march 2013.
- [9]. Farooq, Umar, and John Glauert. "Managing Scalability and Load Distribution for Large Scale Virtual Worlds", *Proceedings of the UEA School of Computing Sciences Symposium*. 2009.
- [10]. C. Jay, M. Glencross, and R. Hubbard. Modeling the effects of delayed haptic and visual feedback in a collaborative virtual environment. *ACM Transactions on Computer-Human Interaction*, 14(2), 2007
- [11]. Zheng, Hanying, and Xueyan Tang. "On Server Provisioning for Distributed Interactive Applications" *Distributed Computing Systems (ICDCS), 2013 IEEE 33rd International Conference on*, IEEE, 2013.
- [12]. Jiang, Hongbo, et al. "Design, implementation, and performance of a load balancer for SIP server clusters." *IEEE/ACM Transactions on Networking (TON)*20.4 (2012): 1190-1202.
- [13]. Balakrishna, Swathi, and Ling Ding. "An Efficient Client Server Assignment for Internet Distributed Systems." *Dept. Comp. Sc. & Sys., Univ. of Washington, Tacoma*.
- [14]. Nishida, H.; Tinh Nguyen; , "Optimal Client-Server Assignment for Internet Distributed Systems," *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on* , vol., no., pp.1-6, July 31 2011-Aug. 4 2011.
- [15]. Delaney, Declan, Tomás Ward, and Seamus McLoone. "On consistency and network latency in distributed interactive applications: A survey—Part I." *Presence: Teleoperators and Virtual Environments* 15.2 (2006).
- [16]. L. Zhang and X. Tang. Client assignment for improving interactivity in distributed interactive applications. In Proc. IEEE INFOCOM'11, pages 3227–3235, 2011. (An extended version is accepted to appear in IEEE/ACM Transactions on Networking.)
- [17]. L. Gautier, C. Diot, and J. Kurose. End-to-end transmission control mechanisms for multiparty interactive applications on the internet. In Proc. IEEE INFOCOM'99, pages 1470–1479, 1999.
- [18]. M.R. Garey and D.S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness*. WH Freeman and Company, San Francisco, Calif, 1979.
- [19]. S.D. Webb, S. Soh, and W. Lau. Enhanced mirrored servers for network games. In Proc. 6th ACM SIGCOMM Workshop on Network and System Support for Games, pages 117–122, 2007.
- [20]. Zhang, Lu, and Xueyan Tang. "The client assignment problem for continuous distributed interactive applications: analysis, algorithms, and evaluation." *Parallel and Distributed Systems, IEEE Transactions on* 25.3 (2014): 785-795.