Client Assignment Problem For Continuous Distributive Interactive Applications

[1]Methuku Archana, [2]Dr. R.China Appala Naidu
[1,2]Dept. of CSE, St. Martins Engineering College, Hyderabad, India

Abstract

In general, interactivity is a primary performance in Distributed Interactive Applications (DIAs) that enable clients at different locations to interact with each other in a real time. The DIAs performance depends, not only client to server latencies but also inter server network latencies, as well as synchronization delays to get the consistency and fairness requirements in DIAs. In this paper, We find the approach effectively assigning clients to servers to maximize the interactivity in DIAs by using algorithms called Greedy Assignment and distributed –modify assignment algorithms. By evaluating the proposed Greedy Assignment and Distributed-Modify Assignment the experimental results show that optimal interactivity is produced and the interaction time between clients is reduced compared to the intuitive algorithm which is assigned to each client to its nearest server.

Keywords

Distributed Interactive Applications (DIAs), Client-Server Interaction

I. Introduction

A client-server interaction model is an association of interconnected clients to servers ,to exchange information between clients and servers. In this advanced world individuals were associated with the web by Distributed interactivity applications (DIAs), such as multiplayer online games, online exam results and distributed interactive simulations, allow participants can have enjoyable interaction experiences. Normally, interactivity is Characterized by the duration from time to time, when a participant issues an operation, with the effect of the operations presented to the same participants [13] or individual participants. In DIAs network latency is major barrier to provide good interaction among participants [9]. The interaction time between any pair of clients must include the network latencies between the assigned servers and their clients. These network latencies are directly shows that how the clients are assigned to the servers. The interaction time is also influences the consistency and fairness requirements of DIAs. Client to server interaction delays are dependent on synchronization.

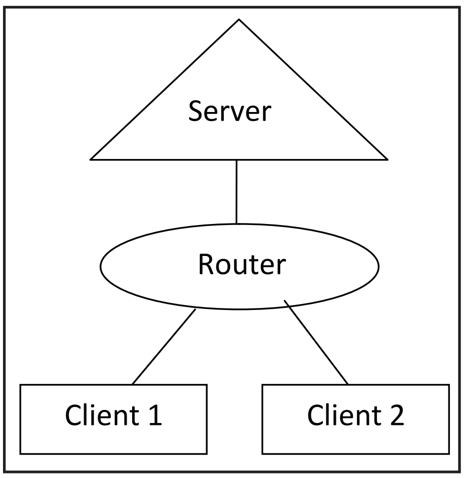

Fig. 1: Distributed Server Architecture

In this paper, we devised the problem of effectively assignment of clients to servers for maximizing the interactivity in DIAs. We focused on continuous interaction that changes the status of clients not only in response to client but also due to passing of time [14]. We start by mathematically modeling the interactivity performance of continuous DIAs under the fairness and consistency. Based on the analysis, we formulate the client assignment problem as a combinational optimization problem and prove that it is NP-complete. But our proposed Greedy Assignment and Distributed-Modify Assignment algorithm produces near optimal interactivity and significantly reduce the interaction time between clients compared to Nearest-server Assignment and Network Latency.

II. Contribution of Paper

In this paper, we are identifying the problem effectively by assigning clients to servers for maximizing the interactivity of DIAs by continuously focusing on DIAs that change their states position not only in response to user-initiated operations but also due to the passing of time. There are several heuristic assignment algorithms are whose approximate ratios are theoretically analyzed. By using Internet latency data the performance of the algorithms is evaluated experimentally. The results show that our proposed Distributed-Modify Assignment and Greedy Assignment algorithms generally produces near optimal interactivity and at the same time it reduces the interaction time between client to client and client to server as compared to the intuitive Nearest-Server Assignment algorithm assigns each client associated to its nearest server. Distributed-Modify Assignment also has good adaptivity towards client participation and network latency.

III. Related Work

In ceaseless DIAs, the improvement of the case state is naturally consider when beyond because the primary condition of the application. We shall describe the reproduction time. For example, the simulation time of a multiplayer web diversion records the point in time past in its virtual world. In the distributed server design, every server and client has a reproduction of the application state and its linked simulation time. The recreation times of all servers and customers ought to go ahead at the comparative rate. However, their readings don't need to be the same at the same wall-clock time. In all inclusive, the re-enactment time of client interims at the reproduction time of its given 1server because of the system dormancy of transport state updates from the server to the customer

Distribute Interactive Applications (DIAs), such as multiplayer web diversions and conveyed interactive simulations, let individuals at unique areas to interrelate with each other from side to side networks. In this way, the intelligence of DIAs is significant for members to have pleasant communication encounters. The interactivity is marked by the period from the time when a participant issues an operation time when the effect of the operation is available to the same participant or different members. We downgrade to this period as the correspondence time between participants. System inactivity is known as a chief barrier to give great intelligence in DIAs. It cannot be killed from the associations surrounded by supporter and has a minor hypothetical limit imposed by the rate of light. Interaction amongst the client and server is very little successful. It has more Network idleness which hindrances intelligent of DIA.

IV. Implementation

To gain the performance measure such as interactivity for DIAs we are implementing the model in various modules like,

A. Nearest-Server Assignment

The Nearest-Server Assignment algorithm, which assigns clients to their nearest servers [15-16]. It implements by having each client measure the network latencies between client to client and all servers, and select the server with the lowest latency as the assigned server. The computational complexity for each client is hence O (|S|). When the Nearest-Server Assignment reduces the Client-server latencies, it will significantly increases the latencies between the server and the client by making the interactivity far worse than optimum.

B. Greedy Assignment

The Greedy assignment approach to assign clients continuously, starting with an empty assignment, each step, the algorithm considers all the possibilities of assigning to an unassigned client to a server. If a client is assigned to a server then all unassigned clients that are not assigned also assigned, as this would not increase the maximum interaction path length.

C. Distributed-Modify Assignment

The Distributed-Modify Assignment is performance is in distributed in distributed networks .It starts with an initial assignment, the assignment continuously modifies to get the maximum interaction path length until it can't be further decreased. This process is referred as assignment modification. One server is selected correspondent server, the server responsible for calculating and selecting the server performance towards the assignment modification to calculate of the initial assignment, each server measures its distances (i.e., latencies) to all other servers. It also measures, distances towards the clients that are assigned to servers and maintain them in a sorting order. Then, each server broadcasts all the other servers lengthiest distance to its clients, and sends the inter-server distances to the coordinator. The coordinator calculates based on the information which is received

D. Server Capacity Constraint

Our proposed assignment algorithms not have any capacity limitation towards servers. Here each and every server are having there correspondent server resources which can be added to particular sites as needed by using the algorithms [17]. However, if the server capacity for each site is limited, assigning more clients to a server than its capacity may results significant increase in the processing delay at the server, damages the interactivity of the DIA.

V. Execution Results

A. Admin Login

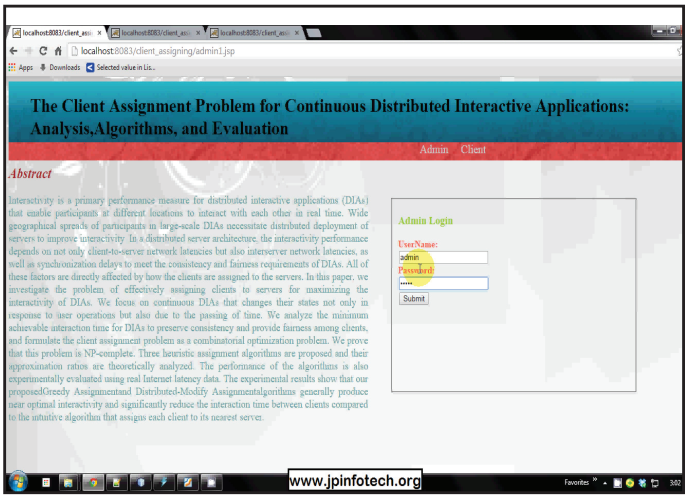

Fig. 2: Admin Login

B. First User Sending Message

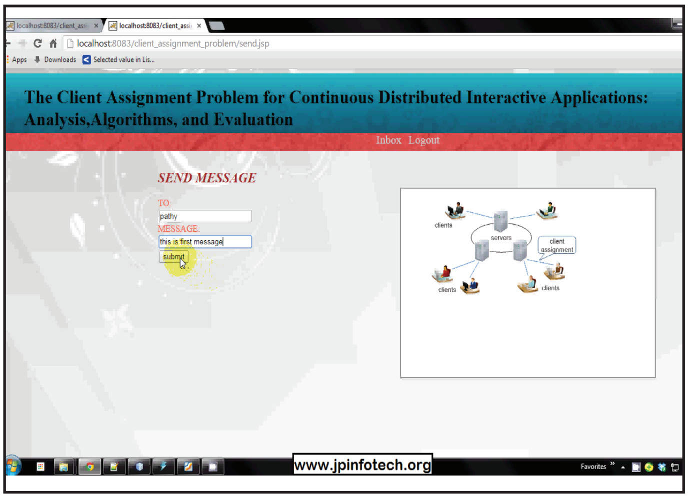

Fig. 3: First User Sending Message

C. Monitor Server

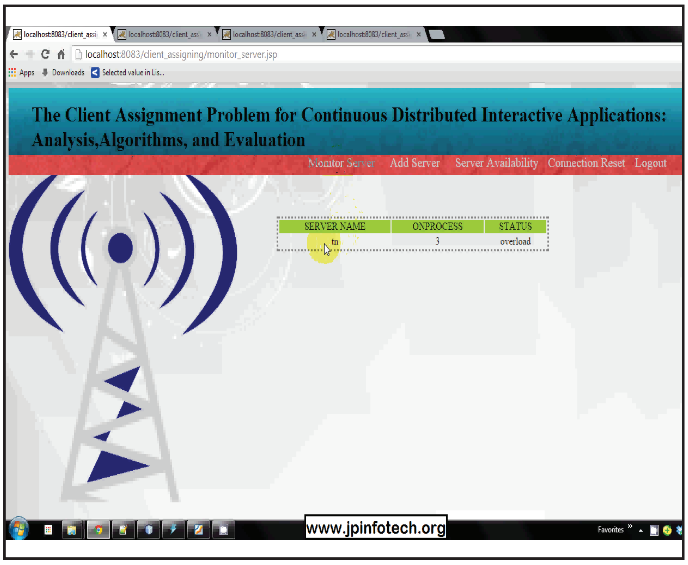

Fig. 4: Monitor Server

D. Admin Adding Server

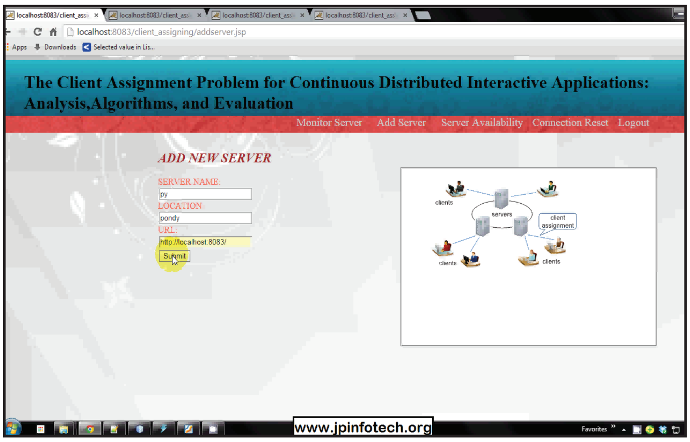

Fig. 5: Admin Adding Server

E. Server Process

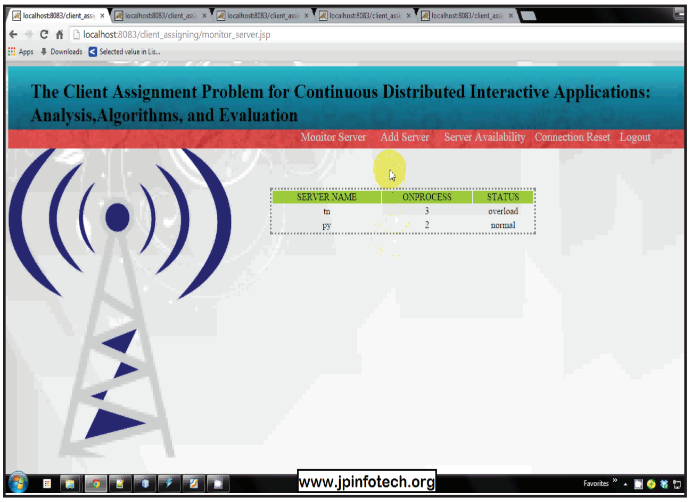

Fig. 6: Server Process

F. When all Servers are Overloaded

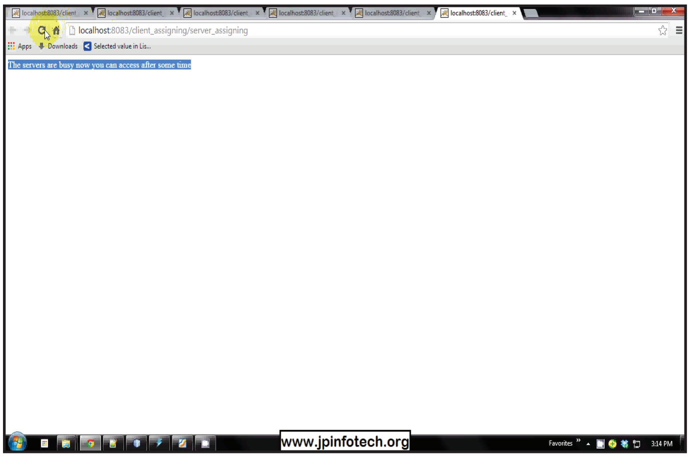

Fig. 7: When all Servers are Overloaded

Table : Test Cases

Table 1: Computation Times for Networks with 80 Servers

CPU TIME			
Algorithm	**Random**	**K-Center**	**K-Median**
Nearest-Server	0.79s	0.80s	0.81s
Greedy	3.26s	2.97s	2.22s
Distributed-Modify	1.42s	1.19s	1.20s

VI. Conclusion

The client assignment problem for interactivity enhancement in continuous DIAs has been investigated by modeling the interactivity performance under the consistency and fairness requirements. The client assignment problem is proven to be NP-complete which is overcome by using the three heuristic assignment algorithms. Their approximation ratios are theoretically calculated and performance is experimentally evaluated. The results show that Proposed as (1) Distributed Modify Assignment requires a small proportion of clients to perform assignment modifications to improve interactivity (2) Greedy Assignment and Distributed-Modify Assignment algorithms significantly performs the intuitive Nearest-Server Assignment algorithm (3) Distributed-Modify Assignment has good adaptivity in both client participation and network latency.

VII. Acknowledgment

I thank our college St.Martins Engineering College who greatly assisted in the success of the project. I express my thanks and gratitude to r. Ch. A Naidu sir, Head of the Department of CSE, St. Martin's Engineering College for his encouraging support and guidance in presenting this paper.

References

[1] Patil, Bharati, S. B. Patil,"A Novel Approach on Client Server Assignment Problem in Distributed System".
[2] J.C.S. Lui, 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.
[3] Greenberg, S., 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, pp. 207–217, October 1994.
[4] L. Gautier, C. Diot, J. Kurose,"End-to-end transmission control mechanisms for multiparty interactive applications on the internet", In Proc. IEEE INFOCOM'99, pp. 1470–1479, 1999.
[5] M.R. Garey, D.S. Johnson,"Computers and intractability: A guide to the theory of NP-completeness", WH Freeman and Company, San Francisco, Calif, 1979.
[6] "LotRO server list," [Online] Available: http://lotro-wiki.com/index.php/List_of_Worlds, 2013.
[7] "Planetlab All-Pairs-Pings," [Online] Available: http://pdos.lcs.mit.edu/strib/2013.
[8] WoW Server List, [Online] Available: http://www.wowwiki.com/Realms_ list, 2013.
[9] E. Cronin, B. Filstrup, A. Kurc,"A Distributed Multiplayer Game Server System," Technical report, Univ. of Michigan, 2001.
[10] E. Cronin, S. Jamin, C. Jin, A.R. Kurc, D. Raz, Y. Shavitt, "Constrained Mirror Placement on the Internet," IEEE J. Selected Areas Comm., Vol. 20, No.7, pp. 1369-1382, Sept. 2002.
[11] E. Cronin, A.R. Kurc, B. Filstrup, S. Jamin,"An Efficient Synchronization Mechanism for Mirrored Game Architectures", Multimedia Tools and Applications, Vol. 23, No. 1, pp. 7-30, 2004.
[12] D. Delaney, T. Ward, S. McLoone,"OnConsistency and Network Latency in Distributed Interactive Applications: A Survey-Part I," Presence: Teleoperators and Virtual Environments, Vol. 15, No. 2, pp. 218-234, 2006.

[13] C. Jay, M. Glencross, R. Hubbold,"Modeling the Effects of Delayed Haptic and Visual Feedback in a Collaborative Virtual Environment," ACM Trans. Computer-Human Interaction, Vol. 14, No. 2, 2007.
[14] M. Mauve, J. Vogel, V. Hilt, W. Effelsberg,"Local-Lag and Timewarp: Providing Consistency for Replicated Continuous Applications," IEEE Trans. Multimedia, Vol. 6, No. 1, pp. 47-57, Feb. 2004.
[15] K.W. Lee, B.J. Ko, S. Calo,"Adaptive Server Selection for Large Scale Interactive Online Games," Computer Networks, Vol. 49, No. 1, pp. 84-102, 2005.
[16] S.D. Webb, S. Soh, W. Lau,"Enhanced Mirrored Servers for Network Games," Proc. Sixth ACM SIGCOMM Workshop Network and System Support for Games, pp. 117-122, 2007.
[17] M. Marzolla, S. Ferretti, G. D'Angelo,"Dynamic Resource Provisioning for Cloud-Based Gaming Infrastructures," ACM Computers in Entertainment, to be published.

Archana M received the B.Tech degree Information Technology from the Jawaharlal Nehru Technological University (JNTU) , Hyderabad, India, in 2014 and is currently pursuing M.Tech at St.Martins Engineering College, Hyderabad, India.

Dr.R.Ch.A.Naidu completed his M.Tech, Ph.D from University of Mysore, Mysore and Andhra University, Vishakhapatnam respectively. He has more than 15 years of teaching experience. He is presently working in CSE Dept as a Professor in St Martin‟s Engineering College, Hyderabad. His area of interest is Network security, Computer networks, Digital Image processing, Data base management systems. He has life membership in professional bodies like ISTE, CSI.