Dynamic Load Sharing Policy in Distributed VoD using agents

H S Guruprasad
Asst Prof & HOD
Dept of ISE, BMSCE
Bangalore, India

Dr. H D Maheshappa
Prof & HOD
Dept of E & C, RevaITM
Bangalore, India

Summary
This paper proposes a load sharing algorithm with a VoD architecture using agent technology. A mobile agent periodically updates the popularity of the videos which is used for efficiently allocating the channels. The proposed approach reduces the load on the central multimedia server, reduces storage redundancy among the proxy servers and maximizes the channel utilization between the neighboring proxy servers and the central multimedia server. The simulation results prove the load sharing among the neighboring proxy servers and hence reducing the load on central multimedia server, 100% channel utilization and more channel allocation for popular videos.

Keywords:
Load sharing, channel allocation, mobile agents, Popularity, VoD architecture, channel utilization.

1. Introduction

Agents are autonomous programs which can understand an environment, take actions depending upon the current status of the environment using its knowledge base and also learn so as to act in the future. Autonomy, reactive, proactive and temporally continuous are mandatory properties of an agent. The other important properties are commutative, mobile, learning and dependable. These properties make an agent different from other programs. The agents can move around in a heterogeneous network to accomplish their assigned tasks. The mobile code should be independent of the platform so that it can execute at any remote host in a heterogeneous network [1, 7, 9].

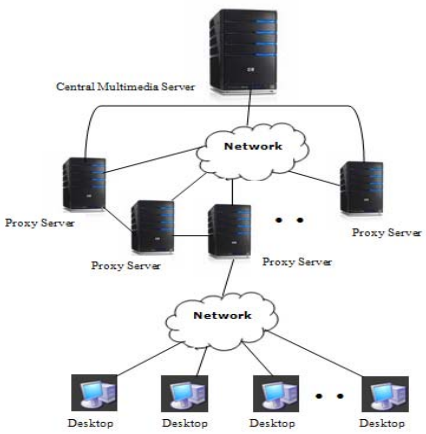
A video-on-demand system can be designed using any of the 3 major network configurations – centralized, networked and distributed. In a centralized system configuration, all the clients are connected to one central server which stores all the videos. All the client requests are satisfied by this central server. In a network system configuration, many video servers exist within the network. Each video server is connected to a small set of clients and this video server manages a subset of the videos. In a distributed system configuration, there is a central server which stores all the videos and smaller servers are located near the network edges. When a client requests a particular video, the video server responsible for the requests ensures continuous playback for the video [2].

Proxy servers are widely used in multimedia networks to reduce the load on the central server and to serve the client requests faster. In [4], Tay and Pang have proposed an algorithm called GWQ [Global Waiting Queue] which shares the load in a distributed VoD system and hence reduces the waiting time for the client requests. This load sharing algorithm balances the load between heavily loaded proxy servers and lightly loaded proxy servers in a distributed VoD. They assumed that videos are replicated in all the servers and videos are evenly required, which requires very large storage capacity in the individual servers. In [5], Sonia Gonzalez, Navarro, Zapata proposed a more realistic algorithm for load sharing in a distributed VoD system. Their algorithm maintains small waiting times using less storage capacity servers by allowing partial replication of videos. The percentage of replication is determined by the popularity of the videos.

In this paper, we propose a new load sharing algorithm and VoD architecture for distributed VoD system. This architecture consists of a central multimedia server which is connected to a group of Proxy servers and these proxy servers are assumed to be interconnected in a ring fashion. The rest of the paper is organized as follows: Section 2 presents the proposed architecture, section 3 presents the proposed algorithm, Section 4 presents the simulation model, Section 5 presents the simulation results and discussion, Section 6 finally concludes the paper and further work.

Manuscript received October 5, 2008
Manuscript revised October 20, 2008

2. Proposed Architecture



In the proposed architecture, a group of proxy servers are connected to a Central Multimedia Server [CMS]. All these proxy servers are connected together in the form of a ring. All the proxy servers are connected together through fiber optic cables. Each proxy server is connected to a set of clients (users). The proxy server stores the video content that is currently requested by its clients.

Initially, all the N number of videos are stored in the CMS. The distribution of the videos is done as follows:

First, all the N videos are arranged with respect to their popularity. The popularity of a video is directly proportional to the number of users requesting for this video. The number of requests to a video follows Zipf law of distribution. We select the first k videos from the popularity based sorted list and stored in each proxy server. The remaining videos are stored depending on the local popularity in the proxy servers.

A mobile agent is invoked by the CMS periodically which travels across the Proxy Servers and updates the video popularity profile at the Proxy servers and the CMS.

When a request for a video arrives at the proxy server [PS], the following 3 cases happen:

- The requested video may be present in the PS
- The requested video is not present in the PS, but is present in either left neighbor Proxy Server[LNPS] or Right neighbor Proxy Server[RNPS]
- The requested video is present in both LNPS and RNPS
- The requested video is not present in LNPS and RNPS

If the requested video is present in the PS, then the real time transmission of the video starts immediately with the video content being streamed to the client from the PS. If the requested video is not present in the PS, then we check whether it is present in the LNPS or in RNPS.

If the requested video is present only in LNPS, then we check the number of channels allocated for popular videos b/w LNPS & PS and CMS & PS. If more numbers of channels are allocated for popular videos b/w LNPS & PS, then we select the path LNPS-PS, otherwise we select the path CMS-PS. If the requested video is the first request, then all the channels are allocated for this video. Otherwise, the popularity of the requested video is checked. If it is more popular than the videos being streamed in the channels, then more number of channels are allocated for the requested video by deallocating channels from the lesser popular videos being streamed in the channels. Otherwise, appropriate numbers of channels are allocated depending on its popularity and the popularity of the videos streamed in the channels and the channel allocation of the other videos are dynamically adjusted.

If the requested video is present only in RNPS, then we check the number of channels allocated for popular videos b/w RNPS & PS and CMS & PS. If more numbers of channels are allocated for popular videos b/w RNPS & PS, then we select the path RNPS-PS, otherwise we select the path CMS-PS. If the requested video is the first request, then all the channels are allocated for this video. Otherwise, the channel allocation is done as the same way as when the video is found in LNPS only.

If the requested video is present in both LNPS and RNPS, then we check the number of channels allocated for popular videos b/w LNPS & PS, RNPS & PS and CMS & PS. We select one of these three paths, in which more number of channels are allocated for most popular videos. If the requested video is the first request, then all the channels are allocated for this video. Otherwise, the channel allocation is done as the same way given above.

If the requested video is not present in LNPS and RNPS, then we select the path b/w CMS-PS. If the requested video is the first request, then all the channels are allocated for this video. Otherwise, the channel allocation is done as the same way given above.

3. Proposed Algorithm

[Nomenclature: PS: Proxy Server
CMS: Central Multimedia Server
LNPS: Left Neighbor Proxy Server
RNPS: Right Neighbor Proxy Server
NCAPV(x): Number of Channels allocated for popular videos b/w x & PS]

When a request Rm for a video m arrives at a particular time t, do the following:

```
If (requested video is present in PS)
        Start streaming from PS
else
  {
   If (requested video is present in LNPS only)
     {
     - If (NCAPV (LNPS) > NCAPV (CMS))
         If (requested video is the first request)
              Allocate all channels b/w PS &LNPS to this
              video
          else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & LNPS and the channel
              allocation of the other videos is dynamically
              adjusted
     - If (NCAPV (CMS) > NCAPV (LNPS))
         If (requested video is the first request)
              Allocate all channels b/w PS & CMS to this
              video
        else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & CMS and the channel
              allocation of the other videos is dynamically
              adjusted
     }
 If (requested video is present in RNPS only)
     {
     - If (NCAPV (RNPS) > NCAPV (CMS))
         If (requested video is the first request)
              Allocate all channels b/w PS & RNPS to this
              video
          else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & RNPS and the channel
              allocation of the other videos is dynamically
              adjusted
     - If (NCAPV (CMS) > NCAPV (RNPS))
         If (requested video is the first request)
              Allocate all channels b/w PS & CMS to this
              video
        else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & CMS and the channel
              allocation of the other videos is dynamically
              adjusted
     }
 If (requested video is present in LNPS & RNPS)
     {
     - If (NCAPV (RNPS) >= NCAPV (CMS) and
          (NCAPV (RNPS) >=NCAPV (LNPS))
         If (requested video is the first request)
              Allocate all channels b/w PS & RNPS to this
              video
         else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & RNPS and the channel
              allocation of the other videos is dynamically
              adjusted
     - If (NCAPV (LNPS) >= NCAPV (CMS) and
          (NCAPV (LNPS) >=NCAPV (RNPS))
         If (requested video is the first request)
              Allocate all channels b/w PS & LNPS to this
              video
         else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & LNPS and the channel
              allocation of the other videos is dynamically
              adjusted
     -   If (NCAPV (CMS) >= NCAPV (LNPS) and
          (NCAPV (CMS) >=NCAPV (RNPS))
         If (requested video is the first request)
              Allocate all channels b/w PS & CMS to this
              video
         else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & CMS and the channel
              allocation of the other videos is dynamically
              adjusted
     }
If (requested video is not present in LNPS & RNPS)
     {
     -   If (requested video is the first request)
              Allocate all channels b/w PS & CMS to this
              video
         else
              Depending on the popularity of the requested
              video
              Appropriate numbers of channels are
              allocated b/w PS & CMS and the channel
              allocation of the other videos is dynamically
              adjusted
     }
```

4. Simulation Model

The simulation model consists of a single Central Multimedia Server [CMS], and a few proxy servers. The following are the assumptions made in the model:

The user requests for the video follows Zipf law of distribution. The sizes of the videos are uniformly distributed over a range. The number of channels b/w the PS & LNPS, b/w PS & RNPS and b/w PS & CMS are assumed to be same.

The performance parameters are load sharing among the proxy servers, reduction of load on the CMS and channel utilization b/w PS & LNPS, PS & RNPS and PS & CMS.

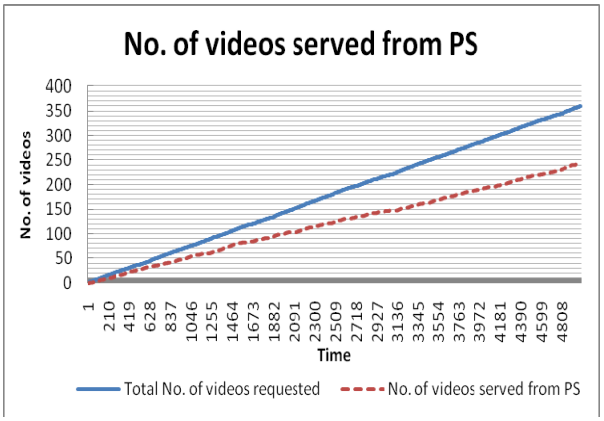


Fig 1 No. of videos served from PS

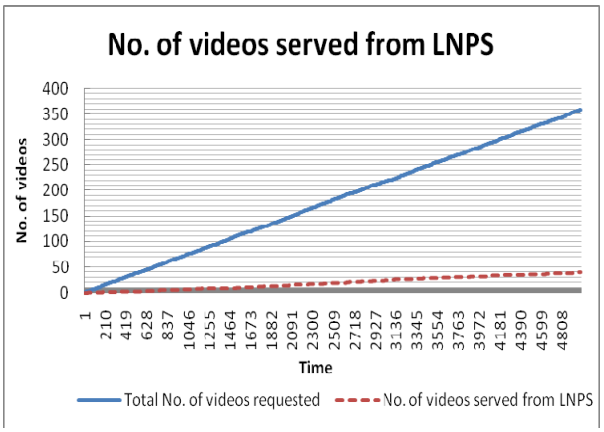


Fig 2. No. of videos served from LNPS

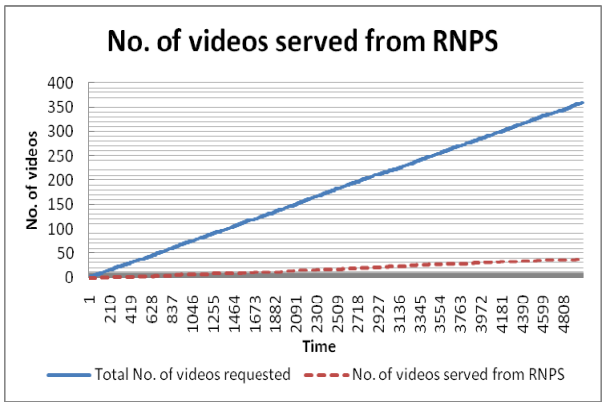


Fig 3. No. of videos served from RNPS

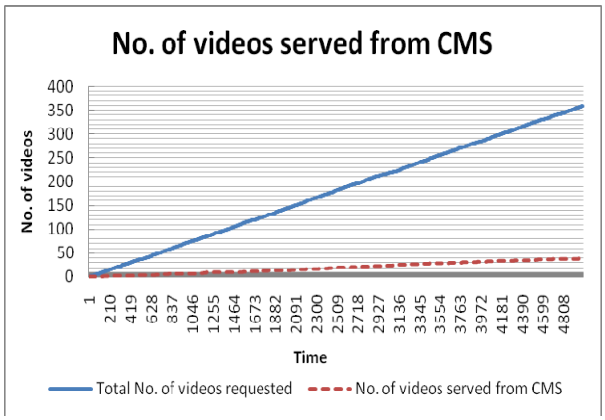


Fig 4. No. of videos served from CMS

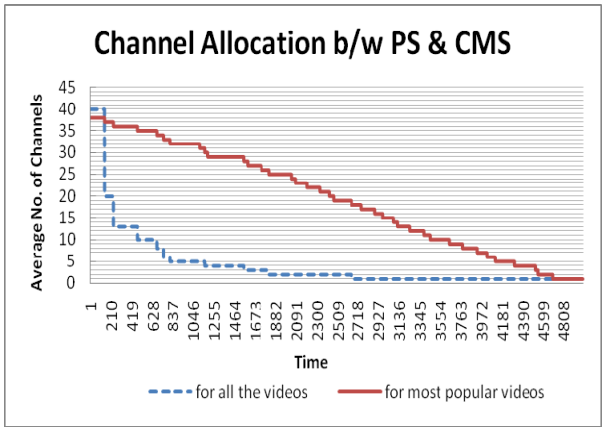


Fig 5. Channel Allocation b/w PS & CMS

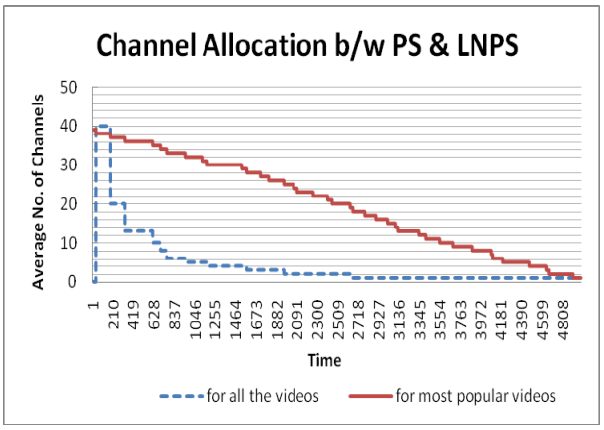


Fig 6. Channel Allocation b/w PS & LNPS

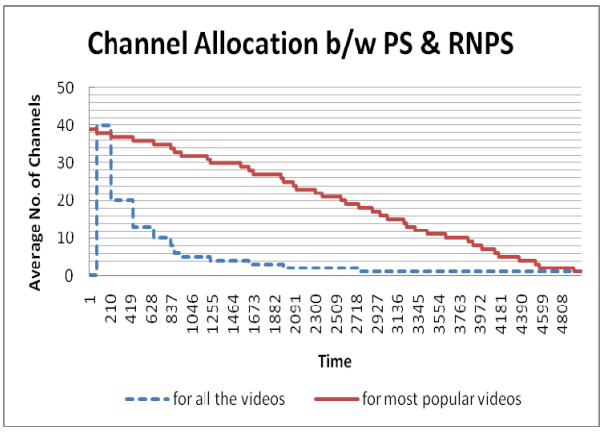


Fig 7. Channel Allocation b/w PS & RNPS

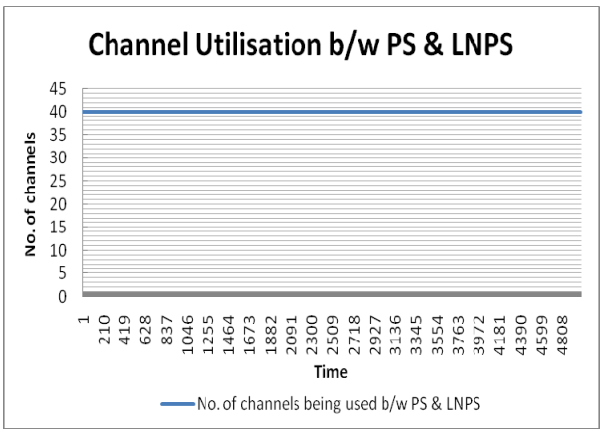


Fig 8. Channel Utilisation b/w PS & LNPS

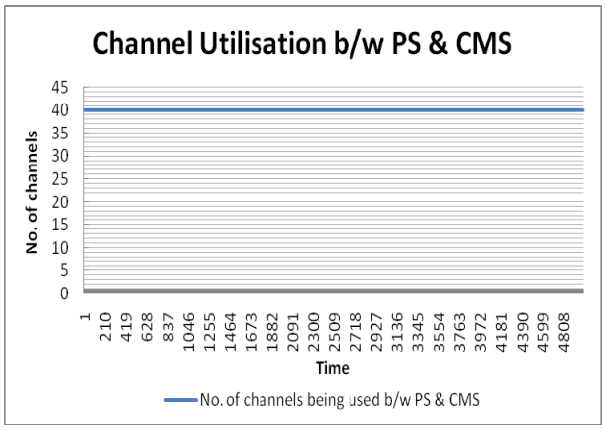


Fig 9. Channel Utilisation b/w PS & CMS

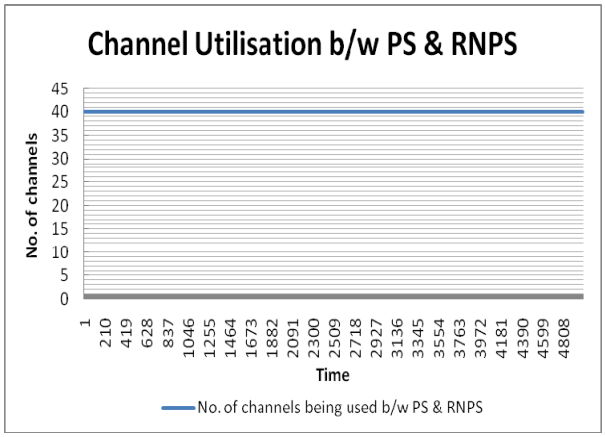


Fig 10. Channel Utilisation b/w PS & RNPS

5. Results & Discussion

Fig 1 shows the total number of videos requested and the number of videos served from the PS. A mobile agent periodically updating the local video popularity profile at the PS and higher priority given to the popular videos results in more number of videos being streamed from the PS.

Fig 2, Fig 3 and Fig 4 show the number of videos served from LNPS, RNPS and CMS respectively. These figures show the load sharing between the neighboring proxy servers and not overloading the CMS.

Fig 5 shows the average number of channels allocated for all the videos and the most popular videos between PS & CMS, between PS & LNPS and between PS & RNPS respectively. Most popular videos are given more number of channels and as the number of videos being streamed

increases, the average number of channels allocated for the most popular videos decreases.

Fig 8, Fig 9 and Fig 10 show the channel utilization between PS & LNPS, PS & CMS and PS & RNPS respectively. Irrespective of the number of videos being served, the channel utilization is the maximum always.

6. Conclusion

In this paper, we have concentrated on the load sharing among the proxy servers and central multimedia server using agents. The simulation shows promising load sharing results. The algorithm always uses 100% of the channels between the neighboring proxy servers and the central multimedia server by allocating more channels to the most popular videos so that they are streamed faster. Further work is being carried out to investigate load balancing by grouping a set of local proxy servers.

References

[1] R Ashok kumar, H S Guru Prasad, H D Maheshappa, Ganesan, "Mobile Agent Based Efficient Channel Allocation For VoD", *IASTED International Conference on Communication Systems & Applications [CSA 2006*] 3rd - 5th July, 2006, Banff, CANADA

[2] Santosh Kulkarni "Bandwidth Efficient Video on Demand Algorithm (BEVA) " *10th International conference on Telecommunications*, Vol 2 pp 1335-1342, 2003

[3] Hongliang Yu, Dongdong Zheng, Ben Y. Zhao, Weimin Zheng, "Understanding User Behavior in Large-Scale Video-on-Demand Systems", *Proceedings of the 2006 EuroSys conference*, Volume 40 , Issue 4 (October 2006), PP: 333 - 344

[4] Y.C Tay and HweeHwa Pang, "Load Sharing in Distributed Multimedia-On-Demand Systems", *IEEE Transactions on Knowledge and data Engineering*, Vol.12, No.3, May/June 2000.

[5] S. Gonzalez, A. Navarro, J. Lopez and E.L. Zapata, "Load Sharing in Distributed VoD (Video on Demand) Systems". *Int'l Conf. on Advances in Infrastructure for e-Business, e-Education, e-Science, and e-Medicine on the Internet (SSGRR 2002w*), L'Aquila, Italy, January 21-27, 2002.

[6] Meng Guo and Mostafa H. Ammar and Ellen W. Zegura, "Selecting among Replicated Batching Video-on-Demand Servers", *Proceedings of the 12th International Workshop on Network and Operating System Support for Digital Audio and Video*, pp 155—163, 2002

[7] Mohammed A. M. Ibrahim, "Distributed Network Management with Secured Mobile Agent Support", *International Conference on Hybrid Information Technology (ICHIT'06)*, 2006

[8] Frederic Thouin, Mark Coates, Dominic Goodwill, "Video-on-Demand Equipment Allocation," *Fifth IEEE International Symposium on Network Computing and Applications (NCA'06*), pp 103-110, 2006

[9] S S Manvi and P Venkataram, "Mobile Agent based online Bandwidth allocation Scheme in Multimedia Communications", *IEEE GLOBECOM 2001 Conference* USA

[10] A Dan, D Sitaram and P Shabuddin, "Dynamic batching policies for an on demand video server ", *Multimedia Systems*, pp 51-58, 1996

H S Guruprasad is an Assistant Professor and Head of the Department of Information Science & Engineering, BMS College of Engineering, Bangalore, India. He is an Engineering Graduate from Mysore University and did his Post Graduation at BITS, Pilani, India in 1995. He has vast experience in teaching and has guided many post graduate students. His research interests include multimedia Communications, distributed systems, computer networks and agent technology.

Dr. H.D. Maheshappa Graduated in Electronics & Communication Engineering from University of Mysore, India in 1983. He has a Post Graduation in Industrial Electronics from University of Mysore, India in 1987. He holds a Doctoral Degree in Engineering from Indian Institute of Science, Bangalore, India, since 2001. He is specialized in Electrical contacts, Micro contacts, Signal integrity interconnects etc. His special interests in research are Bandwidth Utilization in Computer Networks. He has been teaching engineering for UG & P G for last 25 years. He served various engineering colleges as a teacher and at present he is a Professor & Head of the Department of Electronics & Communication in Reva Institute of Technology & Management, Bangalore, India. He has more than 35 research papers in various National and International Journals & Conferences. He is a member of IEEE, ISTE, CSI & ISOI. He is a member of Doctoral Committee of Coventry University UK. He has been a Reviewer of many Text Books for the publishers McGraw-Hill Education (India) Pvt., Ltd, Chaired Technical Sessions, National Conferences and also has served on the advisory and technical national conferences.