Genetic Algorithm for optimizing load distribution on Video on Demand Servers

Yadav Aruna, Kumar Sanjeev

Abstract: As internet usage is increasing day by day web services are expanding very fast. Video on demand servers have heavy web traffic .To speed up the growth of web traffic the concept of load balancer was introduced. The role of load balancer is to distribute the tasks among the web servers efficiently. The most popular algorithms used for distributing the load are: FCFS, Genetic, Randomized and Heuristic algorithms. Performance of algorithms is calculated on the basis of makespan and average resource utilization. Genetic algorithm is giving better result over other server selecton techniques. Genetic algorithm gave lower makespan and higher resource utilization.

Keywords: Video on demand server, optimized algorithms, makespan, average-cpu-utilization, fitness metrics, FCFS, Genetic Algorithm.

1 INTRODUCTION

VIDEO on demand server is a challenging issue in a current scenario. There is more and more demand of video content on web so to meet such demand we require multiple web servers or server farms. Web server is a program that provides content like web pages over the World Wide Web, on the requests from clients. These web servers have their own operating system may be of different types. They regularly get large number of requests from user, so there is issue of balancing load of requests to improve performance of web servers.

Now we focus on a specific application of web servers: Video on Demand. VOD servers must be consistent to transfer higher data rate. Multimedia Mail, Multimedia Mall, Digital Libraries, Video Conference (VC), IP telephony are some popular applications [1,2]. Use of on demand video services have amplified significantly in the recent years and affected the performance of web servers and is predictable to rise further due to expansion in technology . So challenge is to meet the high QoS required by VOD applications. Although when user demands and user access rates increase some problems are faced like high block rate, long startup delay, service interruption and frame losing. The QoS as recognized by the users are generally biased in nature. So they must be traced to an appropriate objective (quantitative) parameter so that it will be technically correct application.

- *Aruna Yadav is currently pursuing Masters degree program in Computer Science Engineering in Banasthali University,Rajasthan,India and affiliated to KIET, Ghaziabad. aruna.yadav21@gmail.com*
- *Sanjeev Kumar is Associate Professor in Department of Computer Science & Engineering, KIET Ghaziabad,India.He has 12 years of experience in teachin and research and published various research papers . sanjeevgreen@gmail.com*

The simultaneous open connections to the web server may have some limitations. Thus the waiting time becomes high when the number of requests to the web server increases, resulting in DOS (Denial of Service) attack .To resolve this problem multiple servers are used known as clustered Web Servers or a server farm.

Multimedia communications is all about to accept requests and transfer the information. These services should be done with minimum delay and continuously and server farm helps it to make better. The performance of a server farm can improve with the improvement of type of routing method, server capacity and scheduling policies used.

The server capacity is of two types homogeneous or heterogeneous. Heterogeneous systems give better results than homogeneous systems if tasks are of different sizes. Heterogeneous systems can also include task-specific systems, i.e. for more computation oriented tasks an array processor can be used.

These servers should have individual operating systems and may provide load balancing approach. Load balancing on servers plays vital role to improve the performance when there are lot of server requests. In loads balancing policy we focus on task location policy which describes scheduling algorithm for various tasks. Scheduling algorithms are described by policy through which they allocate tasks to different web servers. Here we will go through the scheduling algorithms i.e. first come first serve and genetic algorithm. Makespan is a parameter through which we compare different scheduling algorithms. Maximum time consumed to complete all the tasks in task bar given to the dispatcher or load balancer is called Makespan.

2 LITERATURE SURVEY

According to Chande and Sinha[3] Genetic Algorithm is a versatile optimization tool. Genetic Algorithm invented by John Holland is an abstraction of biological evolution. Algorithm works by performing representation, evaluation,

selection, recombination, mutation and inversion. They quoted some application of GA in real world : Nutrional Counseling , Stylometry, Parametric Design of a aircraft, Robot trajectory generation, Strategy acquisition for simulated airplanes, Redistricting ,Problem solving and in-circuit emulators , Acoustics , Aerospace engineering , Bandwidth optimization in near video on demand system , Medical , Scheduling , Musical Composition , Finance ,Identifying criminal suspect , Seeking Routes.

According to Shopova[4] showed that GA involves real representation schemes for both real and integer variables. In representation methods are dynamic representation, real representation of integer variables. In selection for reproduction methods are Roulette Wheel Selection, Rank based selection and Tournament selection. In Crossover methods are N-point-crossover, uniform crossover, arithmetical crossover, blend crossover. In mutation schemes are uniform mutation, non-uniform mutation and breeder mutation.

According to Bajpai and Kumar[5] genetic algorithm works as a global optimization approach.GA are intrinsically parallel and perform well in problem for which the fitness landscape is complex.

Selecting a server and optimized load balancing is studied various research papers. Some server selection algorithms [6,7] are the closest server algorithm that selects server based on the requirement of the client, optimized closest server algorithm that chooses the closest server among the free channels.

According to Gupta et al. [8] analyzed that minimum expected cost algorithm computes mainly server parameters like latency and bandwidth.

Niyato et al. [9] analyzed load balancing for Internet video and audio server and compared algorithms like round-robin, FCFS, max-min and random traditional algorithms along with Adaptive bidding, Diffusion and State change broadcast. Wang et al. studied load balancing in different servers with different service rates and then observed it for heterogeneous systems of multiple servers. This was implemented by heuristic methods by multiple thresholds setting. Ciardo et al. [10] worked on a idea based on size distributions of the requested documents to allocate the tasks to web servers. Zhang et al.[11] derived average response time and the rejection rate and compared three different routing policies to analyze the central load balancing model .

To serve a higher number of requests server replication comes with an additional cost of installing new servers and beyond a certain number of servers, further increase will only lead to more installation cost without improving the Qos like throughput, speedup etc.

For better results decisions are taken under uncertainty which is a difficult problem. Then, after solving the associated optimization problem, one can select the decision that has the best average performance over time or one can select the decision that has the best performance in the expected outcome. Robust discrete optimization, on the other hand, seeks to identify decisions that will perform well under any circumstances.

3 PROPOSED GENETIC ALGORITHM FOR VIDEO ON DEMAND SERVERS

Let there be some servers and a task set A consisting of (A1 A2... An) Tasks hence the basic problem is mapping of a task Ai among the m possible servers so challenge is to find best server selection strategy to improve the performance metrics. There are many factors like server load, response time, makespan that impact the quality of service. The server selection algorithms can be compared by different metrics and this depends on the task set where a particular metric or metrics are of more significance than others. For getting better results ,allocate the task to the servers using different policy after that calculate makespan of each string(Makespan is the largest completion time of all the tasks in the system.),average utilization and Fitness=(1/ makespan) x (average utilization) x (1/no. of processors).Those two strings have better fitness value select them and apply crossover and mutation operation. To get the optimized result perform crossover operation until fitness value reaches greater than 0.4.

VOD_Algorithm(A,N)

1) For i <--1 to n
2) Create_evpop(A) //
3) Calculate Makespan
4) Calculate Average_Utilization
5) Calculate Fitness Value
6) Goto step 2
7) Select two best fitness f1 and f2.
8) Crossover(f1,f2) //
9) If probability < 0.4
10) Goto step 8
11) Mutation(x1, x2)
12) End

Intermediate results

popcurrent[0]= 13 16 10 14 10 10
Value=73
popcurrent[1]= 14 4 22 4 7 3
Value=54
popcurrent[2]= 9 12 16 12 7 19
Value=75
popcurrent[3]= 17 14 1 23 15 23
Value=93

makespan =93

avg_cpu_utilization= 0.196237

fitness_value=0.008527

We have strings before crossover

prev[0][0]= 13 16 10 14 10 10
prev[0][1]= 14 4 22 4 7 3
prev[0][2]= 9 12 16 12 7 19
prev[0][3]= 17 14 1 23 15 23
prev[1][0]= 16 19 3 12 5 14
prev[1][1]= 14 14 1 19 19 9
prev[1][2]= 8 9 12 9 11 19
prev[1][3]= 12 14 15 12 0 2

We have strings after crossover:

after[0][0]= 13 16 10 14 10 10
after[0][1]= 14 4 22 4 7 3
after[0][2]= 9 12 16 12 7 19
after[0][3]= 12 14 15 12 0 2
after[1][0]= 16 19 3 12 5 14
after[1][1]= 14 14 1 19 19 9
after[1][2]= 8 9 12 9 11 19
after[1][3]= 17 14 1 23 15 23

we have strings before mutation:

prev[0][0]= 13 16 10 14 10 10
prev[0][1]= 14 4 22 4 7 3
prev[0][2]= 9 12 16 12 7 19
prev[0][3]= 12 14 15 12 0 2
prev[1][0]= 16 19 3 12 5 14
prev[1][1]= 14 14 1 19 19 9

We have strings after mutation:

after[0][0]= 13 16 10 14 10 10
after[0][1]= 14 17 22 4 7 3
after[0][2]= 9 12 16 12 7 19
after[0][3]= 12 14 15 12 0 2
after[1][0]= 16 19 3 12 5 14
after[1][1]= 14 9 1 19 19 9

4 COMPARING SOLUTION OF VOD USING GENETIC ALGORITHM WITH FCFS

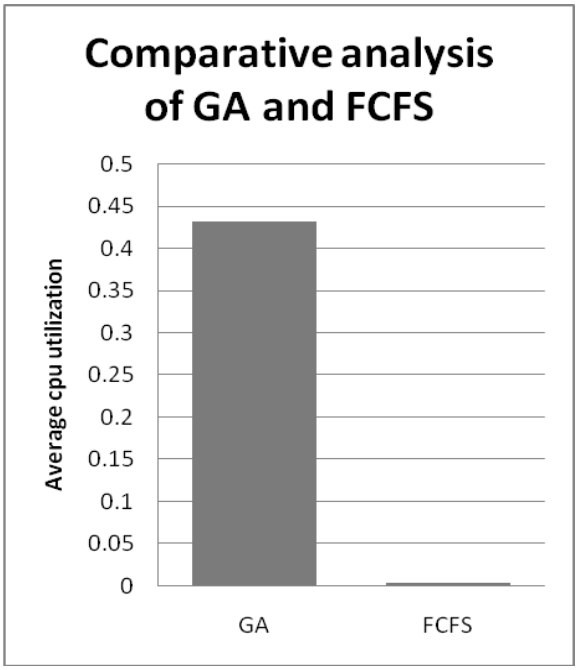

Fig 1: Comparison between GA and FCFS

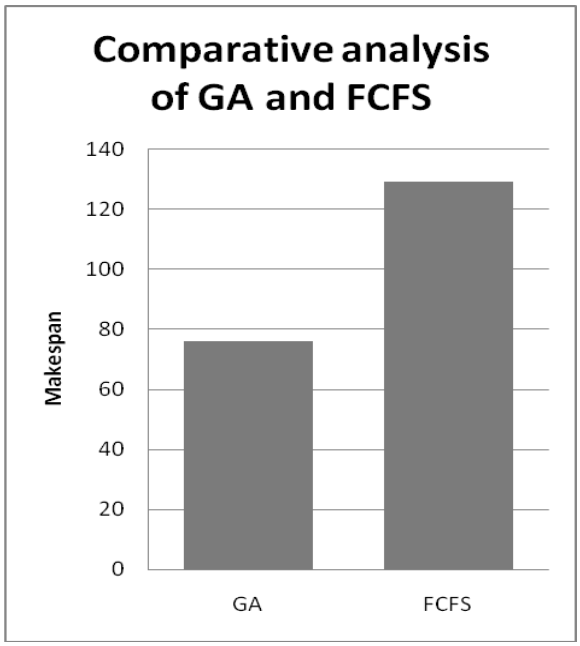

Fig 2: Comparison between GA and FCFS

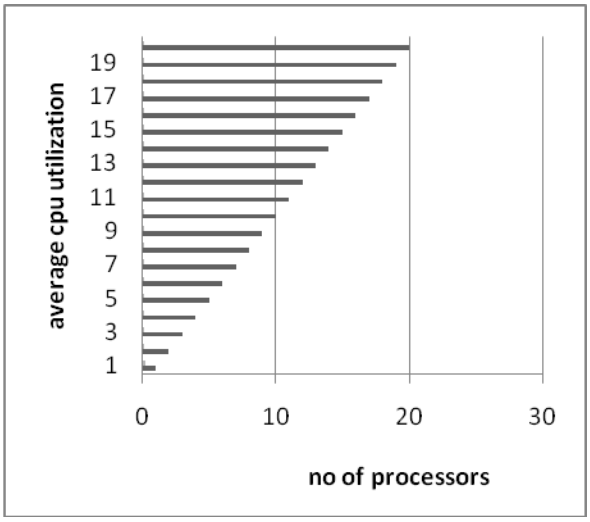

Fig 3: Average CPU Utilization Vs No. of processors in GA

5 CONCLUSION

GA provides far better results than FCFS.As it is very clear from the graphs that GA is efficient algorithm than FCFs. In different criteria GA proved itself that it provides better average cpu utilization and lowest makespan.

7 FUTURE SCOPE

GA was found to be providing the lowest makespan which was due to the knowledge and rejection of the unit candidates while the algorithm progressed. There are many types of Genetic Algorithm,we can use them for better results.Use of Two-Space algorithm can provide better resuls. So Two-Space-GA proposed aimed at taking advantage of efficient algorithms. The proposed solution worked well for large number of tasks.

REFERENCES

[1] N. Panigrahi and B. Sahoo. Qos Based Retrieval Strategy for Video on Demand.Available online at http://dspace.nitrkl.ac.in:8080/dspace/bitstream/2080/789/1/bdsahoo-2009.pdf.Last visited may 08 2011.

[2]Das.Suraj,Prusty.Alok and Sahoo.Bibhudatta" Performance Analysis of Server Selection Schemes for Video on Demand servers"N.I.T.Rourkela.

[3] Chande Swati V. and Sinha Madhavi : Genetic Algorithm: A Versatile Optimization Tool, Bidyapeeth Internatonal Journal of Information Technology, 2008.

[4] E.G. Shopova, N.G. Vaklieva-Bancheva," Basic – a Genetic Algorithm for. Engineering Problem Solution, Computers and Chemical Engineering". Volume 30, march 2006

[5] M. Guo et al.Selecting among Replicated Batching Video on Demand Servers. Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video,May 2002.

[6] Tsutsui, Shigeyoshi, and Ashish Ghosh, "Genetic Algorithms with a Robust Solution Searching Scheme," *IEEE Transactions on Evolutionary Computation*, Volume 1,Number 3, pages 201-208, 1997.

[7] M. Ko and I. Koo. An Overview of Interactive Video On Demand System.Technical Report, The University of British Columbia, Dec 13, 1996

[8] V. Gupta, M.H. Balter, K. Sigman and W.Whitt. Analysis of Join-the-Shortest-Queue Routing for Web Server Farms.Performance Evaluation,vol. 64, no. 9-12, pp. 1062-1081, October 2007.

[9] D. Niyato and C.Srinilta. Load Balancing Algorithms for Internet Video and Audio Server. Proceedings of 9th IEEE International Conference on Networks, pp. 76, 2001.

[10] G.Ciardo, A.Riska and E.Smirni. EQUILOAD: A Load Balancing Policy for Clustered Web Servers. Performance Evaluation,vol. 46, no.2-3, pp. 101-124, October 2001.

[11] Z. Zhang and W.Fan. Web Server Load Balancing: A Queueing Analysis.European Journal of Operational Research, vol. 186, no. 2, pp. 681-693, April 2008..

[12] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Massachusetts, 1989.