A Novel Bandwidth Allocation Algorithm for mobile devices in Multimedia Cloud

Preetha Evangeline D\textsuperscript{1}, Cephas Paul Edward V\textsuperscript{2}, Anandhakumar P \textsuperscript{3}

\textsuperscript{1}Research Scholar, \textsuperscript{2}PG Student, \textsuperscript{3}Professor, Department of Computer Technology, Anna University
Chennai, India

Abstract - Multimedia cloud streaming is the latest trend to meet up with the intensive bandwidth requirements needed by conventional multimedia streaming. Multimedia cloud streaming involves a distributed environment. In such environments, resource allocation plays a major role in the efficiency of streaming to the end users. Bandwidth is a limited resource. The streaming services involve both desktop users and users on mobile devices. We propose a novel allocation algorithm for bandwidth to be shared by both fixed and mobile users in an optimal manner. Also we propose an additional algorithm for fair pricing scheme. We have displayed the results of the simulation of our proposed algorithms.

Keywords — Bandwidth allocation, Pricing, Resource allocation, Streaming, Multimedia Cloud.

I. INTRODUCTION

Recently technology has been incorporated with the concept of mobility. The device have become mobile and also pervasive. The users view multimedia content on-demand from the servers and view them on the go. The change of their location and other features should not interfere with the multimedia content being streamed. Mobile devices usually have a limited bandwidth. This scarce bandwidth resource has to be managed efficiently for uninterrupted services and services at the required level of Quality of Experience (QoE). This is one of the biggest challenges in the area of multimedia cloud streaming. The concept of QoE is entirely exclusive to multimedia cloud environments, which differs a lot from the conventional Quality of Service (QoS). Multimedia providers can guarantee QoS levels as desired by the users rather easily but it is a truly challenging task for them to provide the user with the multimedia services at the desired level of QoE. Providing QoE-aware multimedia services in a cloud environment is an active research area. One of the main concerns in such an environment is the energy consumption by the numerous servers. Multimedia services require intensive computational resources. This requirement is met using a number of high-end servers with high performance capabilities. [1]

Traditional multimedia services include various services such as Streaming, Video conferencing, content sharing, real-time monitoring and broadcasting. If the multimedia services rely on the client systems for their processing the results are very drastic as these services computationally require high processing capabilities. On the other hand, the development of cloud based environment was centered to be highly distributed and to share the workload between a number of servers in the cloud. Hence, there is a great hope for multimedia services to efficiently use the cloud resources and deliver content to the specified user in the manner which he expects. This gave rise to a new area of much interest - Multimedia Cloud, which entirely deals with how the user requests are processed and the needed resources are allocated. Though this may seem easy, there are difficulties additionally in the multimedia cloud, which do not occur in traditional cloud environments. [2]

Peer-to-peer networks[3] can be utilized for the purpose of multimedia streaming. But there is basically one problem. Such networks are usually homogenous in nature. They accept a group of fixed nodes or a group of mobile nodes. But our scenario is a heterogeneous and distributed environment. But the so-called homogeneity is not on the whole because although nodes are fixed, they differ in their computational power, bandwidth requirements etc. And these variations in the features greatly influence the Quality of Service (QoS) parameters. Now in such a scenario, he mobile users have to access all content from the cloud server, there is a great shortage in the bandwidth. Moreover the multimedia service provider (MSP) has to spend more on the traffic he incurs. This traffic is due to the redundant requests from various mobile devices. This cost to the multimedia service provider is leveraged as high costs to the mobile content requestors. And one more parameter which affects the offered QoS to the user directly is the distance between the user client and the multimedia service provider server. Since multimedia files are usually large in size, distance does matter and affects drastically the QoS and the QoE of the video content delivered to the user. This is also a problem to be addressed. So we define an
algorithm to combine the requests of the mobile users by clustering similar ones. We assign a fixed device for each of such cluster. This fixed device requests and retrieves the multimedia content from the multimedia service provider. This solves the problem of so many redundant requests and hence reduces the cost incurred by the multimedia service provider due to high traffic by the redundant requests. Also, since a group of mobile nodes share the video with the fixed node, there is still a further possibility of lowering the prices incurred to the clients. However the clustering of the mobile nodes under a single fixed node introduce a few more issues to be addressed by the proposed algorithm. The fixed node too has a predefined bandwidth. Hence, optimally we have to share the bandwidth between the mobile devices and we have to pay attention to whether the number of mobile users do not cross the particular threshold. And even we can add priority to a particular content from others. If the bandwidth requirements cross a particular threshold, a new fixed desktop node has to be setup and a few members under the existing cluster should be optimally migrated to the new cluster-head(fixed node). To solve this optimally, we have a centralized pool of bandwidth and each fixed node contributes to the bandwidth pool. The mobile nodes (cluster members) draw the required number of bandwidth from the pool for their usage. After their contented usage, the bandwidth resources is released back to the common pool. This has been optimally designed in our proposed algorithm – Mobile Multimedia Bandwidth Allocation (MMBA) algorithm. Owing to this novel bandwidth sharing strategy, the QoE and the underlying QoS are guaranteed. We also incorporate a dynamic pricing system in our MMBA algorithm that dynamically allocates prices to the various users. Moreover, the various entities in each cluster are completely aware of the other entities behavior in the same cluster as they are. They are completely hidden from other entities in other clusters. There is a centralized system to take care of managing the bandwidth resources between the various clusters i.e., for managing the bandwidth pool.

II. RELATED WORK

This section lists out The various research works related to the problem that we have considered and the features and the drawbacks of such approaches. As already mentioned, multimedia cloud services require computational resources to a great extent.

[4] proposes a queue based methodology to optimally provision the QoE and allocate the required resources. It deals with optimization problems such as minimizing the response time (The time lapse between the submission of requests and the start of reply from the server). Also the method includes minimizing the amount of resources required for the QoE provisioning. It deals with the adaptive mobile computing environment for providing multimedia services. The paper deals with high-quality graphically intensive gaming services played from a mobile device. Based on the user's actions on the device, the user interface is rendered accordingly. This is usually of a three dimensional rendering and hence this is performed on the server side. The results of the computations needed for rendering is sent to the user in a compressed format. The user device has only the overhead to decompress the data and make use of it. [6] deals with the design of a novel stream dispatcher. The responsibility of this dispatcher is to analyse the incoming requests and identify fluctuations. The content to be delivered is segments and based upon the current characteristics of the environment, the dispatcher module makes an optimal decision to send the most appropriate segment depending on the scenario. Goudarzi et al., [5] have developed a network specific bandwidth allocation framework. They have adopted a cross-layer design approach and have also proposed a few novel metrics to measure the QoE guaranteed between the end-to-end systems.

Fulp et al., [7] proposed a bandwidth allocation strategy with specific reference to scalability. This was achieved by plotting a demand vs price curve and analyzing it. The approach helped in maximizing the profit and to reduce the probability of blocking. Shin et al., [8] proposed the idea of noting down the user preferences and they have designed a ranker to rank out the contents of the videos. Also they have applied a filtering approach to produce nice recommendations for various classes of users. Zang et al., involved in the work of applying social networking concepts to multimedia streaming to effectively increase the QoE to the users. It is based on the reputation values of the peers in the network. The same authors in another of their research paper have proposed a trust based approach which involves recording of information and using it in future for various tasks. [9] proposes a multimedia content recommendation framework. At first, the various services available and the various social network sub-groups were listed out and analysis was performed on them. Based on the results of the analysis, the content can be recommended to a group of users dynamically. The authors of [10] have made a deep study on the unique position and colluders. They have come up with a multimedia streaming social network based framework. They have carefully studied the various utility functions and the parameters of these colluders which can help them in their decision making process. In [11], a pricing model is proposed with the aim that bandwidth sharing among the peers must be optimal. The algorithm paid attention to uncertainty in the bandwidth and adopted a genetic-based approach to produce convincing results. [13] proposes a game theoretic approach to deal with the
pricing mechanism. [12] brings out a pricing policy based on auction. The users can bid many number of times for a connection and the auction manager sees to that the profit is maximized.

III. PROPOSED BANDWIDTH ALLOCATION ALGORITHM

All Multimedia service providers are those with multiple servers and provide the required multimedia content to the users. As discussed before, the bandwidth of the mobile device users is limited. Hence we propose a novel bandwidth allocation algorithm (MMBA algorithm). In our proposed architecture, we have three layers of operation. At the cloud level, we deploy servers to deal with the storage/retrieval of content and to manage the overall resource allocation process and the request processing. We have separate file server for storage and a processing server to perform the various management tasks. The storage and retrieval and other tasks like processing requests and resource management have been put onto the cloud side because the processing power is higher and the throughput is greater. The system is fault tolerant too. This ensures that the client systems need to be just as thin clients which require only minimal processing. Moreover, the requests can be uploaded to server and the client can go to sleep mode, thereby power conservation too. The client wakes up on a reply from the server. The client side is homogeneous in nature. The client devices are categorized into two – fixed nodes and mobile nodes. So we define an algorithm to combine the requests of the mobile users by clustering similar ones. We assign a fixed device for each of such cluster. This fixed device requests and retrieves the multimedia content from the multimedia service provider. This solves the problem of so many redundant requests and hence reduces the cost incurred by the multimedia service provider due to high traffic by the redundant requests. Also, since a group of mobile nodes share the video with the fixed node, there is still a further possibility of lowering the prices incurred to the clients. However the clustering of the mobile nodes under a single fixed node introduce a few more issues to be addressed by the proposed algorithm. The fixed node too has a predefined bandwidth. Hence, optimally we have to share the bandwidth between the mobile devices and we have to pay attention to whether the number of mobile users do not cross the particular threshold. And even we can add priority to a particular content from others. If the bandwidth requirements cross a particular threshold, a new fixed desktop node has to be setup and a few members under the existing cluster should be optimally migrated to the new cluster-head(fixed node). To solve this optimally, we have a centralized pool of bandwidth and each fixed node contributes to the bandwidth pool. The mobile nodes(i.e., cluster members) draw the required number of bandwidth from the pool for their usage. After their contented usage, the bandwidth resources is released back to the common pool. Further the mobile nodes are categorized based on their characteristics. They are able to communicate with each other and also their fixed-counterparts. We provide the structural diagram representing 2 mobile clusters and three fixed nodes with two videos streaming as follows:

![Diagram](http://www.ijcttjournal.org)

**Fig 1. Overview of the system**

From the above diagram it is clear that, the file server is located at the cloud end. Initially let there be one fixed node streaming a sport video. It has under itself a cluster of two mobile nodes. Now let a new mobile node request for the sports category video. Since there is already a fixed node streaming the video, just a bandwidth allocation is involved and the mobile node is added to the sports cluster. Let another mobile node request an Education video. Since there is already a fixed node streaming the video, the system looks for another free fixed node and starts up the same category of sports streaming and moves two of the nodes to the new fixed node, thus making it equal in both the clusters.

We propose a bandwidth allocation model based on the scenario of a marketing environment. The fixed nodes market their bandwidth wirelessly to the mobile nodes for a particular cost. Each mobile node incurs a cost or a price to access the video from the server.

Consider the mobile nodes in the system: 

\[ m_1, m_2, \ldots, m_n \] 

Let there be fixed nodes \( f_1, f_2, \ldots, f_k \).

Now we cluster these \( n \) mobile nodes into \( m \) groups: \( C_1, C_2, \ldots, C_m \). We cluster these mobile nodes into the various clusters by analyzing the similar characteristics of the various mobile nodes. Each mobile node can connect with only one fixed node at a time. It cannot simultaneously connect with two or more fixed nodes. Within a cluster all the mobile
nodes are transparent and can communicate with each other and analyze each others behavior. They can even exchange their private data with each other. All the mobile nodes within a cluster are treated equally by the server.

The fixed nodes are said to provide the bandwidth and allocate it to the mobile nodes based on their requests. The number of users who connect to the fixed node i is given by \( n_i \). The portion of bandwidth supplied by the fixed node be \( b_i \). Hence, the bandwidth per user is computed as \( b_i/n_i \).

We consider the concept of time slots \( t_1, t_2, \ldots \). The users requests are dynamic. A few requests are usually received. But the situation can change at any instant. The requests may increase drastically and the bandwidth allocation algorithm has to dynamically adapt according to it. First of all, the bandwidth allocation algorithm analyses the current scenario and has to make a decision whether the requested bandwidth can be provisioned or not. If no, the algorithm exits stating allocation is not possible. If the decision produces a successful result, the next step is obtaining the various properties of the mobile nodes and grouping the mobile nodes into clusters.

Initially, the price as per the setting of the fixed node is taken and then this is subsequently updated with the price result obtained from the pricing mechanism as per the proposed MMBA algorithm. This process can be iterated until a convincing pricing is obtained. At time slot \( t_z \), the final slot, it is assumed that all requests have been processed and completed and the pricing report is sent to the fixed nodes and this pricing is adopted by the mobile nodes in turn. We take \( n \) mobile nodes and \( k \) fixed nodes. A fixed node i informs the server with parameters \( b_i \) and \( c_i \), the bandwidth contributed by it and the pricing cost and the mobile node j submits his bandwidth requirement. The final pricing is denoted by \( c_z \). The fixed node is like a retailer, it has to provide the good quality bandwidth and get a good pricing for it as much as possible. The mobile node has to receive the video at his desired level of QoS and QoE.

So for any fixed node \( n_i \), the total cost can be computed as

\[
\text{cost}_\text{total} = c_i \times n_i
\]

we have already defined the quantity \( c_i \).

Now consider any particular mobile user j, let \( b_j \) be the bandwidth associated with it.

\[
b_j = \frac{\sum b_i}{\sum n_i}
\]

And the utilization factor \( \mu \) of a mobile node j be defined as:

\[
\mu = \log (\text{some constant} \times b_j)
\]

Now the overall utilization can be derived as:

\[
\mu_{\text{overall}} = \mu - p_j
\]

Where \( p_j = \sum c_i \times \text{toggle} \).

Toggle is a binary quantity, containing a value of one if there exists a connection between mobile node j and fixed node i, multiplied by per unit pricing. In this manner, the bandwidth is allocated between various mobile nodes.

Now, we turn our attention to the pricing scheme of the MMBA algorithm. The pricing factor is also a critical quantity in the bandwidth allocation process. We can model this situation mathematically as a function involving number of nodes under a particular cluster as the dependent variable \( x \) and the pricing as the independent variable \( y \).

\[
y = f(x)
\]

There is a tradeoff situation that arises here. We can compromise either the quality or the price, but not both. A group of users prefer to spend a lesser cost toward the streaming and hence they can compromise on the quality and opt for lower quality content. This can be mathematically modeled as:

\[
y' = y_{\text{max}} - (\text{const})(\text{price})
\]

A few applications like highly sensitive applications cannot compromise on the quality and hence are forced to higher costs. This can be similarly modeled as:

\[
y'' = \frac{y_{\text{const}}}{\sqrt{\text{price}}}
\]

Consider \( n' \) to be the relative count of the mobile nodes in the former kind and \( n'' \) be the relative count of the mobile nodes in the latter kind. So for a fixed node, we can obtain as:

\[
y_i = (n' \times y') + (n'' \times y'')
\]

The profit obtained is proportional to this quantity. Hence we have to maximize the above quantity. This can be expressed as an optimization problem mathematically as:

\[
\text{max}\{y_i\}
\]

Subject to :

\[
\sum_{i=1}^{n} \text{toggle} = 1
\]
where the first constraint provides us a guarantee that each mobile node can be a member of only one cluster at any time instant $t$, and the second constraint guarantees that each and every mobile node is provided with a bandwidth allocation guarantee.

Mathematically, it can be proved that the highest profit value leads to the maximization of the quantity $y_i$. And this occurs at a value of

$$
(0.5 \times (c_1 - \sqrt{c_2} ))^{\frac{1}{3}} + (0.5 \times (c_1 + \sqrt{c_2} ))^{\frac{1}{3}} + \frac{C_3}{3 \times Const}
$$

And this is satisfied when the maximization value is greater than or equal to $e_i \times b_i$ where $e_i$ is the energy consumed for streaming a particular content. We know that the requests fluctuate dynamically and hence the algorithm must adapt the parameters accordingly as and when there is a fluctuation to obtain the maximal profit. So we cannot fix the values of these parameters before hand. But we can adjust the equation in such a way that a few parameters can be fixed and the others can be dynamically updated.

**Step 1:** Initialization. The initial connection is established. This is bounded by the two constraints of the maximization function. The first constraint provides us a guarantee that each mobile node can be a member of only one cluster at any time instant $t$, and the second constraint guarantees that each and every mobile node is provided with a bandwidth allocation guarantee. And the established connection $conn_{init}$ is returned to the next step.

**Step 2:** The fixed node sets its own initial value of pricing.

**Step 3:** Perform initial clustering based on the utilization factor. Then the mobile nodes exchange private parameters and the re-clustering takes place until there is no significant change in the clusters formed.

**Step 4:** Pricing mechanism applied as discussed above. Initialize the tuple $<y_{const}, y_{max}, n', n'', const>$ for each fixed node in the system. Compute $c_i$ value. Optimize the maximization criterion. $c_i$ is accepted if this value is greater than $e_i \times b_i$ per unit mobile node. Iterate through this step until there is no significant change in consecutive values. Thus we have obtained an optimal allocated bandwidth and pricing for each node in the system.

**Input:** set of fixed nodes, set of mobile nodes

**Output:** connections with optimal bandwidth and pricing values

for each $m_1, m_2, ..., m_n$

set up $conn_{init}$ such that the criteria is satisfied.

end for

set initial pricing as per desktop

for each cluster $C_i$ do

  for each mobile node $j$ in cluster $C_i$ do

    exchange_private_parameters();

    assign to some other cluster

  end for

end for

compute utilization factor and recluster

until no change in cluster

end for

set initial pricing tuple $<y_{const}, y_{max}, n', n'', const>$ for each fixed node in the system

for all fixed nodes $f_i$ do

  Compute $c_i$ and maximize it

  if ($c_i \geq e_i \times b_i$ per unit mobile node)

    accept $c_i$

  else recomputed $c_i$

  end if

end for

until no considerable change in pricing

end for

return;

end MMBA
IV. IMPLEMENTATION AND RESULTS

We have implemented the proposed MMBA algorithm and we analyse the performance pertaining to various parameters and settings. We have implemented the system for two fixed nodes and correspondingly two clusters under each fixed node. Hence we have two fixed nodes f1, f2. The two clusters include C1 and C2. B1 and b2 denote the bandwidth contributed by f1 and f2 respectively. m1, m2 be the number of mobile nodes in clusters C1 and C2 respectively. We take k to be 80, per unit price constant as 0.5 and e1, e2, e3 as 0.2 each. y\textsubscript{max} and y\textsubscript{const} to be 20 and 10. n1 and n2 ranges from 0.4 to 0.8.

The following graph shows the comparison between the average utilization factor values of two mobile nodes m1 and m2 in the clusters C1 and C2 respectively. Let these values be represented by U1 and U2 respectively. From the graph it is understood that initially there is a variation in the values of average utilization factors due to the fact that the initial allotment was randomly done by each fixed node. As iterations continue, finally we have the values in a steady state.

The following graph depicts the variation in the number of mobile nodes in cluster C1 as initial pricing varies. We have plotted the results for three initial pricing values. From the graph, it is clear that as the number of mobile nodes in cluster C1 increases, it is directly proportional to the number of users with f1. And also it is clear that when the pricing increases it reduces the number of mobile nodes.

We have two fixed nodes f1, f2. The two clusters include C1 and C2. b1 and b2 denote the bandwidth contributed by f1 and f2 respectively. m1, m2 be the number of mobile nodes in clusters C1 and C2 respectively. We take k to be 80, per unit price constant as 0.5 and e1, e2, e3 as 0.2 each. y\textsubscript{max} and y\textsubscript{const} to be 20 and 10. n1 and n2 ranges from 0.4 to 0.8. We have in the following figure two sets of plotted values as varying the values of n’ and n”. As n’ increases the number of mobile nodes with f1 also increases and it is also noted that there is a point of intersection between the expected and the proposed curves.
The following graphs show the utilization factors of the node under $f_1$ and $f_2$ separately and also the overall utilization factor value. The curves are increasing.

![Utilization factor](image)

**Fig. 6. Utilization factor of $f_1$ with respect to pricing updation.**

The following graph shows the variation in the overall utilization factor across all the iterations. The iterations are continued until there is no significant difference between the previous and the current values. This is clearly seen in the graph, in which the initial portion is increasing whereas final part is nearly flat.

![Utilization factors](image)

**Fig 7. Utilization factors**

The following graph shows the utilization factor of $f_1$ with respect to pricing updation.

![Utilization factor](image)

**Fig. 6. Utilization factor of $f_1$ with respect to pricing updation.**

The following graph shows the variation in the overall utilization factor across all the iterations. The iterations are continued until there is no significant difference between the previous and the current values. This is clearly seen in the graph, in which the initial portion is increasing whereas final part is nearly flat.

![Utilization factors](image)

**Fig 7. Utilization factors**

The following graph shows the variation in the overall utilization factor across all the iterations. The iterations are continued until there is no significant difference between the previous and the current values. This is clearly seen in the graph, in which the initial portion is increasing whereas final part is nearly flat.

![Utilization factors](image)

**Fig 7. Utilization factors**

The following graph shows the variation in the overall utilization factor across all the iterations. The iterations are continued until there is no significant difference between the previous and the current values. This is clearly seen in the graph, in which the initial portion is increasing whereas final part is nearly flat.

![Utilization factors](image)

**Fig 7. Utilization factors**

The following graph shows the variation in the overall utilization factor across all the iterations. The iterations are continued until there is no significant difference between the previous and the current values. This is clearly seen in the graph, in which the initial portion is increasing whereas final part is nearly flat.

**V. CONCLUSIONS AND FUTURE WORK**

We have proposed a novel bandwidth allocation algorithm – MMBA (Mobile Multimedia Bandwidth Allocation) algorithm. The client devices are categorized as fixed nodes and mobile nodes. So we define an algorithm to combine the requests of the mobile users by clustering similar ones. We assign a fixed device for each of such cluster. This fixed device requests and retrieves the multimedia content from the multimedia service provider. This solves the problem of so many redundant requests and hence reduces the cost incurred by the multimedia service provider due to high traffic by the redundant requests. Also, since a group of mobile nodes share the video with the fixed node, there is still a further possibility of lowering the prices incurred to the clients. However the clustering of the mobile nodes under a single fixed node introduce a few more issues to be addressed by the proposed algorithm. The fixed node too has a predefined bandwidth. Hence, optimally we have to share the bandwidth between the mobile devices and we have to pay attention to whether the number of mobile users do not cross the particular threshold. And even we can add priority to a particular content from others. If the bandwidth requirements cross a particular threshold, a new fixed desktop node has to be setup and a few members under the existing cluster should be optimally migrated to the new cluster-head(fixed node). To solve this optimally, we have a centralized pool of bandwidth and each fixed node contributes to the bandwidth pool. The mobile nodes(i.e., cluster members) draw the required number of bandwidth from the pool for their usage. After their contented usage, the bandwidth resources is released back to the common pool. We have modeled this problem as a mathematical optimization problem and solved for it providing optimal bandwidth and pricing allocation to the various nodes.
REFERENCES


