Asian Jouranl of Information Technology 4 (12): 1152-1158, 2005

A Scalable Framework for End to End QOS Provisioning in Wireless Networks Using Adaptive Multimedia System Architecture

T. Sasipraba and S.K. Srivatsa
Sathyabama Deemed university , Chennai, India
Madras Institute of Technology , Chennai, India

INTRODUCTION

With the growth of bandwidth available in wireless networks, it is feasible to stream multimedia rich audio/video content to mobile clients. Some minimum QOS must be provided to support smooth audio/video playback. Fluctuations in network resource availability due to channel fading, variable error rate, mobility, and handoff, makes QOS provisioning more complex in wireless networks. This study present an adaptive end-system based architecture for improving QOS in wireless networks. The adaptation is based on user preference in order to increase the perceptual value of the multimedia stream by making better use of available bandwidth. The end-system based architecture consists of modules at the two ends of the network, namely, the mobile client and the multimedia server. After the connection is established with a multimedia server, the client periodically sends feedback about bandwidth availability to the server. The server stores multiple copies of streaming data encoded at different fidelity levels. Based on feedback and user preference, the scheduler at the server dynamically selects the appropriate copy of audio/video stream. User preferences are specified in terms of user level QOS parameters such as resolution and frame rate, to keep the interface simple for the user.

The requirements of this proposals are

- The changes in the existing architecture
- The number of clients
- The choice of fidelity levels
- The encoding scheme to be used
- The user preference to be given
- The method to determine perceptual value

This study proposes an end-system based architecture for improving QOS in wireless networks. The architecture consists of two modules at the two ends of the network ,mobile client and multimedia server. The server stores multiple copies of the data encoded at different fidelity levels. The user gives his/her preference in terms of parameters such as resolution, frame rate etc. After the connection is established with the multimedia server, the client keeps sending feedback to the server about the bandwidth available to it. Depending on the user preference and the feedback received, the server chooses the best fidelity level and transmits the data. The expected data and the received data are compared to obtain the perceptual value.

ISSUES

- There is an overhead involved due to the constant exchange of information between the client and the server in the form of feedback messages.
- There is an overhead introduced due to the use of client preferences which causes a slight reduction in the playback time.
- The format of the messages to be transmitted may vary depending on the user preferences provided and also on the transmission protocol used.

MOTIVATION

Multimedia-rich audio /video content is being streamed to the clients nowadays due to the increase in the bandwidth available. The available bandwidth has increased from 9.6Kbps-14.4Kbps (2G-GSM and TDMA wireless networks of 1960’s) to 64Kbps (3G networks). Increasing bandwidth is a necessary first step for accommodating real-time streaming applications. However it is not sufficient due to unpredictable and large bandwidth fluctuations experienced in wireless networks. Some minimum Quality of Service (QOS) must be provided to support smooth audio/video playback. Fluctuation in network resource availability due to channel fading, variable error rate, mobility, and handoff, makes QOS provisioning more complex in wireless networks.

Thus there must be an architecture which would allow the server to monitor the bandwidth available at the client side and transmit the data to the mobile client in such a way that there is minimum amount of data loss and there is maximum satisfaction of the client.

Corresponding Author: T. Sasipraba, Sathyabama Deemed University , Chennai, India

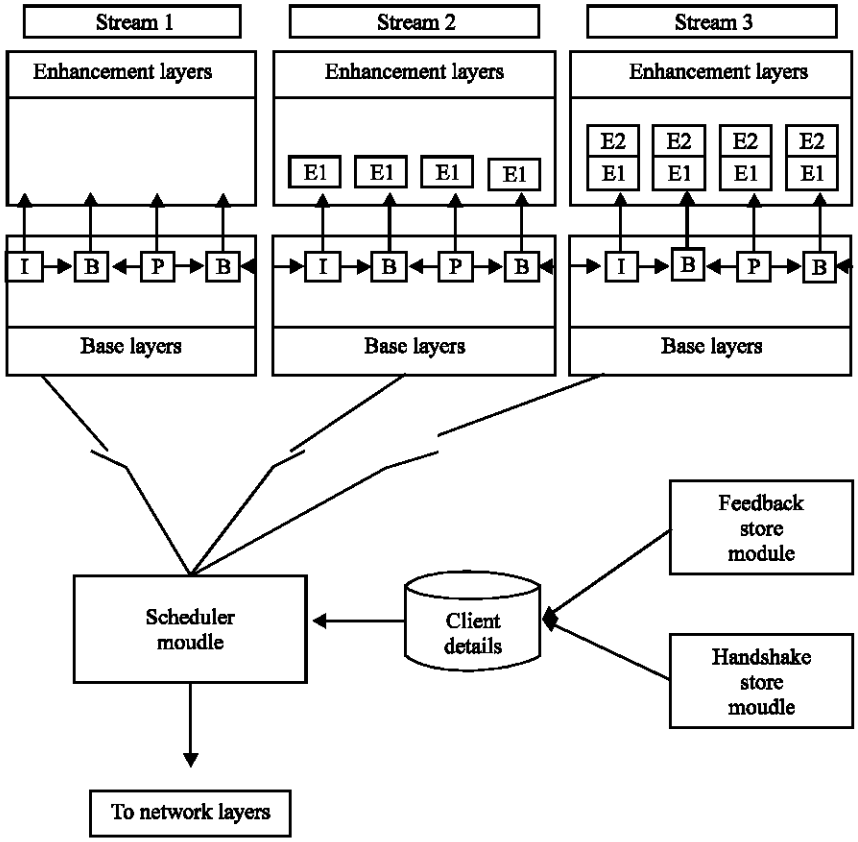

Fig . 1: Scheduling base and enhancement layers

SCOPE

- This architecture can be adapted to any wireless network including the 3G wireless networks.
- This takes the user preference into account so it is able to maintain a minimum amount of the QOS as specified by the server.
- This architecture also takes the user's preference thereby satisfying the user.

FACTORS DICTATING THE DESIGN

The end-system based architecture consists of modules at the two ends of the network, namely, the mobile client and the multimedia server. Thus, the system does not have any dependency on the underlying network, making its implementation possible in any wireless network. After the connection is established with the multimedia server, the client periodically sends feedback about bandwidth availability to the server. The server stores multiple copies of streaming data encoded at different fidelity levels. Based on feedback and user preference, the scheduler at the server dynamically selects the appropriate copy of the audio/video stream. The adaptation to the available bandwidth also provides means of avoiding the network congestion. User preferences are specified in terms of QOS parameters such as resolution and frame rate, to keep the interface simple for the user. The factors dictating the design of the system are given below.

The end-system based architecture consists of modules only at the two ends of the network, namely the mobile client and the multimedia server[1]. Using the mechanism explained next, the client components have the best knowledge of bandwidth available and user preferences. Client's current knowledge of bandwidth is sent to the server. The server will periodically send some control packet at higher bandwidth than reported by the client. Depending on the rate at which the client is able to receive data, any decrease or increase in bandwidth will be detected by the client. The server components have the best knowledge of the levels of data fidelity stored in the database. Hence a system with participation of both client and server components should yield better results. The two end-systems (client and server) can be relatively easily modified and updated. With the current size of the

Table 1:Perceptual value of Data

		No user preference		With user preference	
Preference	Expected data	Received data	REMR (%)	Received data	REMR (%)
Video	Au + BL	Au + BL	100%	Au + BL	100%
Frame-rate	BL + 2EL (@ 12 fps)	Au + BL (@25fps)	50%	BL + 2EL (@ 12 fps)	100%
Resolution	BL + EL	Au + BL	50%	BL + EL	100%

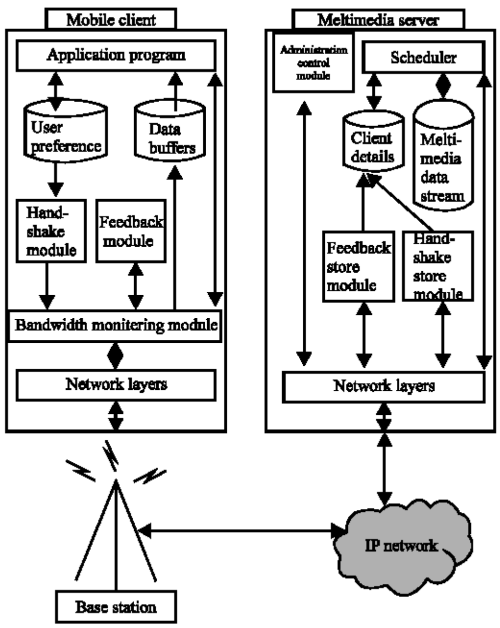

Fig. 2:Adaptive multimedia system architecture

Internet, it is a quantum task to effect any change in the network. Based on this intuition, this system is designed such that it has no dependency on the underlying network bandwidth[2].

Figure 1 shows the way in which the system gracefully adapts to the changes in the bandwidth. This gives a means of degrading the quality when the bandwidth and other resources are limited and change frequently. For better video quality, the base layer and two enhancement layers or a single enhancement layer are sent depending on the user preference and the availability of resources.

Table 1 indicates that if a user would like to view the video at a higher quality, then, the expected data is the audio and the base video layer. When the user preference is not given, the server sends the audio layer as well as the base video layer since it is not aware of the user's preferences. This would be the basic combination for viewing the video. When the user preferences are given, the server sends the base video layer and an enhancement layer. This would mean that the user gets better quality data.

The adaptive multimedia end-system architecture is as shown in Fig. 2.

The following are the modules that constitute the system architecture.

Client application: This module accepts the client preferences in a user friendly interface.

Two simple choices have to be made: the first is preference between audio and video, and the second is the preference between frame rate and resolution. The client application stores these preferences in a database for later use.

Handshake module: This accesses the preference database and sends the information in MSG_HANDSHAKE message to the server.

Bandwidth monitoring module: This keeps a check on the current network state by keeping track of the amount of data being received by the client. The information is used to determine the bandwidth available to the client device. It periodically invokes the services of feedback module to update the server about the bandwidth variation.

Feedback module: This module is periodically invoked by the bandwidth monitoring module to update the server (by MSG_FEEDBACK) about the bandwidth available to the mobile client.

Admission control module: This module decides whether the multimedia server has sufficient free resources to service a new request from a client. Based on available resources, the server decides to accept or reject a connection request.

Handshake store module: This module processes the MSG_HANDSHAKE message received from the client during the initialization phase. It then stores the client preferences received in the message in the *client details* database.

Feedback store module: This module processes the MSG_FEEDBACK message received from the client and stores the bandwidth availability value in the *client details* database.

Scheduler module: This module uses data from the *client details* database to select the appropriate stream of multimedia data. Both client preferences and bandwidth available to the client are used to decide the appropriate stream. The scheduler also prepares data packets for transmission to the client.

Client-server interaction: The server and client interaction starts with the connection initialization phase in which the client requests streaming data from the server. The server accepts or rejects the request based on admission control.

In the handshake phase, the user preferences are transferred to the server and stored in the client database for future use. The handshake message, MSG_HANDSHAKE, has two bits for user preference (audio/video and resolution/ frame rate). The available bandwidth is monitored by the client and reported to the server. The feedback message, MSG_FEEDBACK, has four values of 1 byte each for bandwidth over the previous 30 seconds (Bw30), 60 seconds (Bw60), 120 seconds (Bw120) and 180 seconds (Bw180). Thus, the size of MSG_HANDSHAKE is 29 bytes and of MSG_FEEDBACK is 32 bytes, with IP header 20 bytes and UDP header 8 bytes. The server adapts the fidelity of data in the adaptation phase and transmits the adapted video stream.

RESULTS AND DISCUSSION

This study is proposing an adaptive end system architecture in which user will be asked to give their preferences like frame rate and resolution. Also in this proposed architecture the client will be periodically informing the server about the bandwidth available at the client side. So the server will have the knowledge of the bandwidth available at the client side which will make the server to serve the client in an efficient way without any data loss at the client.

To understand how efficient the client is served in this architecture, we have considered three scenarios[1] which are

- Without Feedback and No User Preference
- With Feedback and No User Preference
- With Feedback and User Preference

Here feedback refers to the information about the bandwidth available at the client side which is given to the server.

The output of the project will be three separate graphs for the above said three scenarios and another graph showing the comparison of the three scenarios. The graphs are drawn by taking the simulation time at the X-axis and playback time along the Y-axis. The playback time is nothing but the time for which the audio or the video is played back.

Without feedback and no user preference: In this scenario the client will not send the feedback to the server. Also the user is not asked to give their

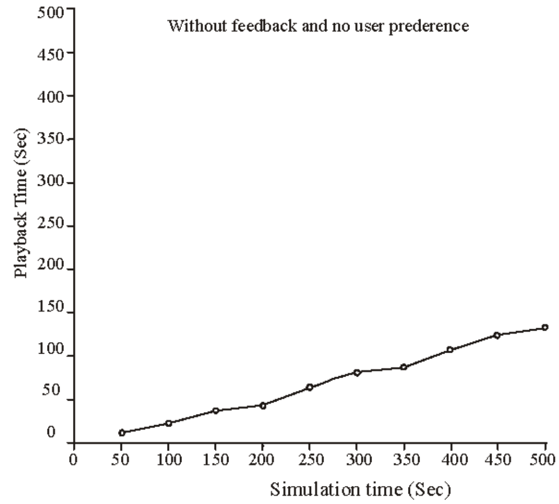

Fig. 3: Without feedback and no user preference

Table 2: Average playback percentage of time

Simulation time (sec)	Percentage Average Playback Time (sec)
100	21.0
200	22.5
300	26.0
400	26.5
500	26.2

preferences. The user can only select the audio or the video he/she wants. Since the server does not know the bandwidth available, it will send the requested data in the standard bandwidth (data rate). So there will be chance of the data being lost due to less bandwidth at the client. So there will be gaps in the playback which will reduce the playback time. The graph between the simulation time and the playback time is shown in the Fig. 3.

The average playback percentage of time at various simulation time for this scenario is given in the Table 2.

With feedback and no user preference: In this scenario the server will be knowing the band width available at the client side that is the feedback will be sent to the server. But the user is not asked to give their preference. The user can only choose either an audio or a video. The server will send the requested data in the available bandwidth (data rate) which leads to less loss of data at the client side. So the playback time will be increased or more. The graph between the simulation time and the playback time for this scenario is shown in Fig. 4.

With feedback and user preference : In this scenario the user will be asked to enter the frame rate and the resolution apart from the selection of an audio or a video. Also the server will be knowing the bandwidth available

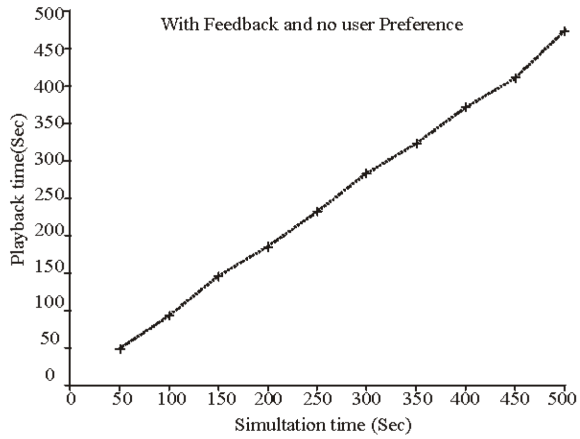

Fig. 4: With feedback and no user preference

Table 3: Average Playback percentage of time

Simulation time (sec)	percentage average playback time (sec)
100	95.0
200	92.0
300	93.33
400	92.75
500	94.2

Table 4: Average playback percentage of time

Simulation time (sec)	percentage average playback time (sec)
100	98.0
200	88.0
300	89.33
400	88.0
500	92.4

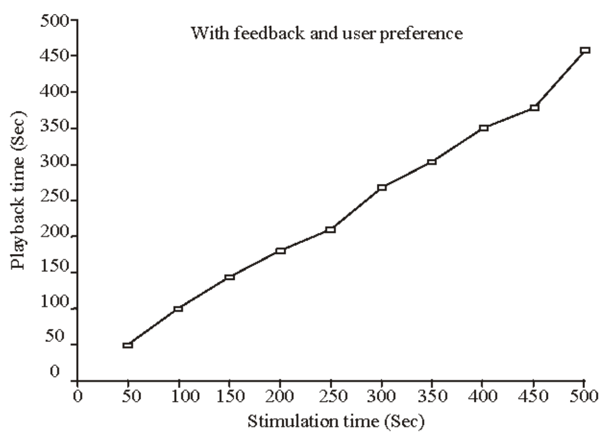

Fig. 5: With Feedback and User Preference

The average playback percentage of time at various simulation time for this scenario is given in the at the client side. The requested data will be sent in the available bandwidth which reduces the data loss and the user will be satisfied more. This results in the increase of the playback time. The average playback percentage of time at various simulation time for this scenario is given in the between playback time and the simulation time for this scenario is shown in the (Fig. 5)

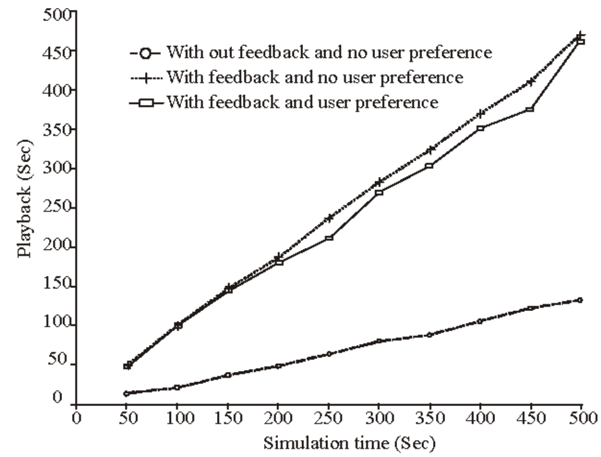

Fig. 6: Comparison graph

COMPARISON

The comparison graph between playback time and the simulation time of all the three scenarios is shown in the Figure6. Without feedback the multimedia data is played for 21 to 26% of time. With feedback it is played for 87 to 98% of time.

Here the decrease in the playback time for with feedback and user preference scenario is because of the client overhead problem. The overhead problems are negligible. The proposed end system is having the advantage of increased playback time and also the user will be better satisfied since they can choose the frame rate and resolution according to their wish.

CONCLUSION AND FUTURE ENCHANCEMENTS

This study is based on improving the perception of the user. Thus the perceptual value is of utmost importance here. Perception is how the user sees things. In this project, we allow the user to give the preferences that are required by him. Hence it helps to improve the perceptual value. Significant improvements in playback time are obtained by using the proposed adaptive multimedia system. The system does not have much of overhead which makes it suitable for less resourceful mobile client. The perceptual-value based system takes the user preference into account resulting in better adaptation. Our system does not have any dependency on the underlying network, making its implementation possible in any wireless network. The system adapts to both user preferences and network resources to improve the perceptual value of the data delivered to the user. The technology is developing at a very fast rate.

Hence the future enhancements may propose a better architecture than the existing one. This end-system based adaptive architecture does not depend on the underlying network. Thus it can be implemented on any wireless networks including the future 4G wireless networks. Due to the messages being exchanged between the mobile client and the multimedia server, there is a small overhead. The overhead is caused by using the client preferences and it causes a slight reduction in playback time. The future enhancements may be able to improve on this architecture and propose a solution where this overhead will be removed.

REFERENCES

1. Mahajan, A., P. Mundur and A. Joshi. An Adaptive Multimedia system Architecture for improving QOS in wireless networks.
2. Margaritidis. M. and G. Polyzos, 2000. Mobiweb: Enabling adaptive continuous media applications over 3G wireless links. IEEE Personal communications magazine.
3. Pedro, M. and R.E. Garcia, improving User perceived QOS in mobile and wireless IP networks using real time adaptive multimedia applications.
4. Sisalem, D., 1997. End-to-End Quality of service control using Adaptive Applications. IFIP Fifth International workshop on quality of Service.
5. Noble, B., m. Satyanarayanan, 1999. Experience with adaptive mobile applications in odyssey.Mobile networks and applications 4: 254-254.
6. Bahl, P., 2000. Supporting digital video in a managed wireless network. IEEE Communications Magazine special Issue on wireless Viedo., 36: 94-102
7. Bharghavan, V. and V. Gupta, 1997. A framework for application adaptation in mobile computing environments .Computer software and applications Conf.,Bethesda MD.
8. Joshi, A., 2000. On proxy agents , mobility, and web access.ACM/Baltzer. J. mob. networks and nomadic applications., (MONET) 5: 233-241.
9. Puri, A. and A. Eleftheriadi, 1998. MPEG - 4: An object based multimedia coding standard supporting mobile application .Mobile networks and applications 3: 5-32
10. Lei Huang, S. Kumar and C.C. Jay kuo, 2000. Adaptive resource allocation for multimedia QOS management in wireless networks.
11. Lee, K., 1995. Adaptive network support for mobile multimedia, in proc. of ACM Mobicom, pp: 62-74.