

Analysis of M/M/1 Queuing model of Reservation Management for Media Gateway Controller

Pedram Hajipour*, Nahid Amani*, Farzaneh Seyed Mostafaei*

* Iran Telecommunication Research Center, Tehran, Iran

pedramhajipour@gmail.com, n_amani@itrc.ac.ir, f.seyedmostafaei@yahoo.com

Abstract — In recent years, Internet Protocol (IP) telephony has become a good alternative to the traditional Public Switched Telephone Networks (PSTN). IP telephony offers more flexibility in the implementation of new features and services. The Media Gateway Controller (MEGACO) is becoming a popular signalling protocol for Voice over IP (VoIP) based applications. This paper analyzes performance of MEGACO model (Media Gateway Controller for Telephones) based on COPS (Common Open Policy Server) which is a protocol defined in IETF (Internet Engineering Task Force) to transport configuration requests and deliver the policies. There is a basic “request <> response” protocol for the policy and information exchange, the COPS protocol buffer can be divided into three distinct conceptual layers: basic protocol, directives depending on the customer type and the representation of the policy data; moreover, MGC provides call control function of MG (Media Gateway) to set up, tear down and manage VoIP calls in carrier class VoIP network. In this paper, we simulated the M/M/1 performance model of the Media Gateway Controller and studied some of the key performance benchmarks response time to process the MEGACO calls and mean number of MEGACO calls in the system; therefore, we have used these criteria by queuing theory. Numbers of jobs based on variation service rate are presented beside some numerical examples of call setup time based on variation service rate.

Keywords—MEGACO, COPS, Media Gateway, Media Gateway Controller, M/M/1

I. INTRODUCTION

Voice over IP (VoIP) technology is currently finding its way into telecommunication market. It enables a telecommunication company to cut down its costs by allowing a single network to transmit both data and voice traffic. In addition, VoIP technology is gaining popularity in both commercial and residential markets because the voice quality resulting from packets transmitted over the IP network is comparable to the voice quality resulting from analogue signals sent over the Public Switched Telephone Network (PSTN). In recent years the MEGACO/H.248 signalling protocol has been introduced by the Internet Engineering Task Force (IETF) and International Telecommunication Union (ITU) to help controlling and managing the growing volume of VoIP

traffic. With emerge of VoIP technology; voice traffic is no longer restricted to the circuit-switched network.

New IP-based products, such as IP phones and voice cable modems, have been introduced to integrate voice services over the data network. To properly manage and control these voice services, various signaling protocols have been developed. One of these protocols is MEGACO/H.248. It provides the master/slave architecture for controlling VoIP traffic.

The MEGACO/H.248 signaling protocol employs a call control concept. The call control “intelligence” or the master server resides in the Media Gateway Controller (MGC), while the Media Gateway (MG) serves as the slave device (dumb terminal). This concept reduces the complexity of the gateway and making it simpler and more suitable for mass deployment. Six MEGACO commands are implemented: Add, Audit Capabilities, Audit Value, Modify, Notify, Service Change, and simultaneously. Several MGs are connected to a single MGC via a router. We verified all signalling commands and simulated the MG registration and call-establishment scenarios. To verify that voice calls were actually established, we generated voice packets encoded using the RTP protocol. Comparison of PSTN services such as the 1-800 services and 911 emergency services with the Internet telephony services needs to evaluate the performance, scalability and reliability of the VOIP network [1]. According to the simulation of M/M/1 queuing model proposed in [2], we obtained the bandwidth management results of our proposed call flow. We consider a single MGC in this model to avoid the propagation delay between the MGCs.

II. MEGACO/H.248 PROTOCOL

A. History

In traditional circuit-switched networks, call setups are performed primarily through backbone of the telephone network. As a result, a proprietary signalling protocol can be used for establishing and deleting connections. Nonetheless, a well defined signalling protocol is required for VoIP since VoIP traffic is routed through the public network infrastructure. Various signalling protocols have been designed to control VoIP

traffic. For example Peer-to-peer protocols, such as SIP and H.323 have been introduced. Still these protocols have scalability problems in terms of large-scale deployments. Hence a new architecture for signalling protocols was proposed; the control and the media gateway components were re-defined using the master/slave architecture. Figure 1 shows the evolution of the MEGACO/H.248 protocol.

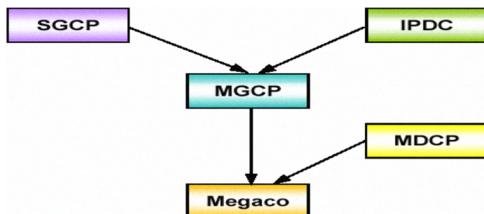


Figure 1. Evolution of the MEGACO/H.248 Protocol [3]

B. Gateway Architecture

In order to specifically model the MEGACO scenario, it is necessary to understand the design and functions of MEGACO. The MEGACO/H.248 protocol employs the master/slave architecture, where the MGC acts as the master server and MG behaves as the slave device. Figure 2 illustrates the simplified gateway architecture. In a deployed telecommunication network, one MGC may control multiple MGs.

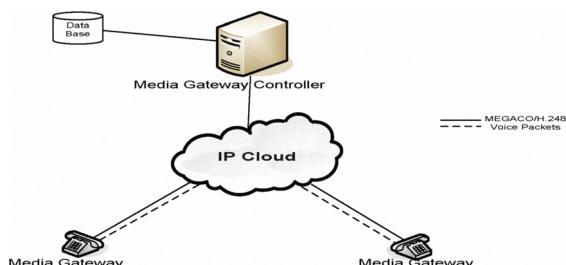


Figure 2: The Master/Slave Architecture

C. Media Gateway Controller

The MGC is the central point of intelligence for call signaling. It maintains the states of each MG and responds appropriately to any event notifications. For instance, upon receiving an off-hook event from a MG, the MGC instructs the MG to play the dial tone and listen to the Dual Tone Multi-Frequency (DTMF) tones.

D. Media Gateway Controller

The master/slave architecture was designed to eliminate processor-intensive functionalities from the MG. The cost of MG is much lower than the cost of MGC due to the reduced complexity which makes it more affordable to the commercial identical markets. Essentially, the MG is a dumb terminal and a waiting command from the MGC until its next actions. Upon the successful creation of a connection, the MG is also responsible for streaming the voice packets over the IP backbone using various encoding/compression algorithms.

E. MEGACO/H.248 Command Set

The commands supported by the MEGACO/H.248 protocol are[3]:

[MGC ↔ MG]

- Service Change – Notify the responder of the new service state

[MGC ↔ MG]

- Audit Value – Determine the characteristics of an endpoint
- Audit Capabilities – Determine the capability of an endpoint
- Add – Add a connection
- Modify – Change a connection characteristic
- Subtract – Tear down a connection
- Move – Move an endpoint from one connection to another connection (call-waiting)

[MG ↔ MGC]

- Notify – Notify the responder of an event (on hook)

III. DESIGN ARCHITECTURE

We have designed two core modules for the MEGACO/H.248 protocol: MGC and MG [4,5].

A. MGC Architecture

The MGC consists of three main components as shown in Figure3: Message Receiver (MR), Message Processor (MP), and Message Sender (MS).

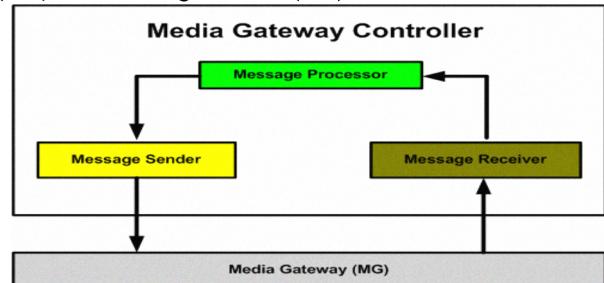


Figure 3: Components' Responsibilities in the MGC

B. MGC Node

Figure 4 shows the node model of the MGC. The MGC node model consists of a transmitter, a receiver, and the MGC processor. The MGC processor is responsible for parsing MEGACO messages, determining necessary actions for the MGs, and composing the MEGACO messages.



Figure 4: MGC Node Model

The MGC process model, shown in Figure 5, has four states: *init*, *idle*, *send*, and *process*. The *init* state initializes all the resources in the MGC. Upon successful initialization, the system proceeds to the *idle* state, where it either waits for the MGs to respond or for the MGs to send new MEGACO requests. The system enters to the *process* state when new MEGACO messages arrive. In this state, the received messages are parsed and appropriate responses to the MGs are generated. These

responses are placed in an outgoing queue where they are periodically transmitted when the system enters the *send* state.

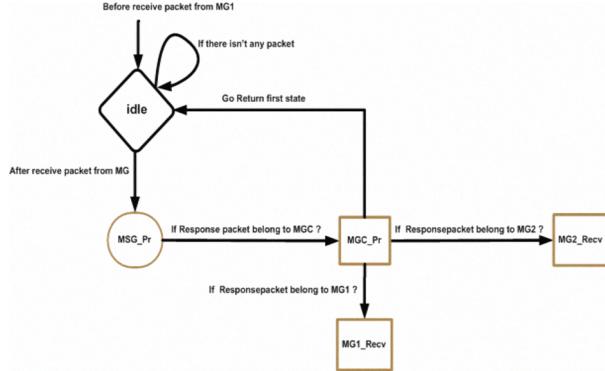


Figure 5: MGC Finite State Machine (Process Model)

C. MG Architecture

The MG consists of five components: Message Receiver (MR), Message Processor (MP), Message Sender (MS), User Interface (UI), and Voice Generator (VG). Figure 6 illustrates the architecture of the MG. The functionalities of the Message Receiver and Message Sender in the MG are similar to those in the MGC. The Message Processor is built to handle any commands sent from the MGC. Upon receiving a command from the MGC, the Message Processor responds accordingly, as listed in Section 2.5. The User Interface component initiates various events, such as off-hook, flash-hook, and on-hook. Once an event is initiated, the corresponding event notification message is sent to the MGC via the Message Processor and the Message Sender.

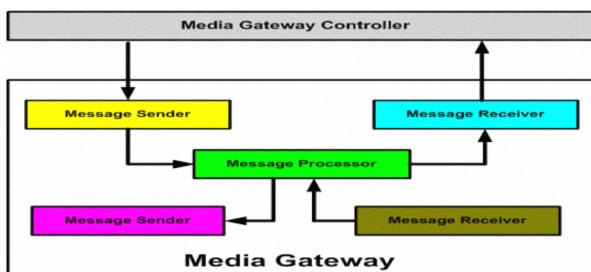


Figure 6: MG Design Architecture [3]

The Voice Generator is responsible for generating voice packets upon call establishment. These voice packets are then encapsulated into the RTP payload and sent over the network. Similar to the MGC, communications between MG components are accomplished through local function calls.

D. MG Node

The MG node, shown in Figure 7, consists of a transmitter, a receiver, and the MG processor. The MG processor is responsible for handling MEGACO/H.248 commands sent from the MGC and detects events

initiated by the user. Furthermore, RTP packets are generated by the MG processor for voice transmission.



Figure 7: MG Node Model

Figure 8 depicts the state machine of the MG processor, which consists of the *init*, *idle*, *msg_pr*, *mg_pr*, and *usr_pr* states. The MG objects are initialized in the *init* state. After initialization, the state machine proceeds to *idle* state. In this state, the process waits for the periodic interrupt and packet arrival. When the state machine receives a regular interrupt in *idle* state, it enters *usr_pr* state. The state machine checks whether an event was initiated by the user. If an event was initiated, a corresponding event notification message is sent to the MGC. In this state, the state machine also checks for call establishment. Once a call is established, the state machine generates an RTP voice stream to the corresponding MG. Upon receiving a message in the *idle* state, the state machine proceeds to the *msg_pr* state where the packet type is determined. If a signaling command is received, the state machine moves to the *mg_pr* state where the command gets processed. Alternately, if a RTP packet is received, the state machine moves to the *usr_pr* state where the MG object statistics are updated.

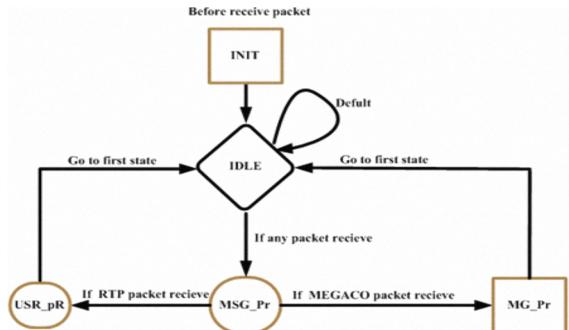


Figure 8: MG Finite State Machine (Process Model)

IV. CALL FLOW SCENARIO

For assessment of bandwidth management call flow, the following test bed including two Media Gateways, Media Gateway Controller, COPS (Bandwidth manager), and two routers which are routing between two Media Gateways territory and a bandwidth interface is presented in figure 9.

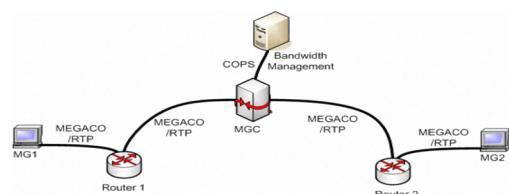


Figure 9: MEGACO Call with Bandwidth management in different domain

As seen in Figure 10, four basic call flow scenarios are defined to validate the implementation of six MEGACO/H.248 commands. Call Setups with Resource Reservation in MEGACO are described in the following subsections, where the MEGACO/H.248, COPS commands, and main parameters are included within the sequence diagrams. Since their input and output processes traffic in one direction and according to Markov model, their previous values are not dependent. We use the M/M/1 queue model to analyse call setup scenario. The traffic model is considered in each process requiring information processing. As illustrated in Figure 11, this model has been considered for establishing a call between the Media Gateway with Bandwidth Management caused by COPS. COPS protocol is a

protocol to create policy for reservation of resources. This process begins with a Notify Request from MG1 to MGC to establish a call. For this purpose MG1 sends Off-hook signal to MGC, then MGC sends Notify Reply for MG1. Media gateway controller sends necessary signal communication including tone signal and numbering signal for MG1. MG1 processes of above signals and sends MG2 address MGC for establishing a call. Then MGC for communicating sends Add Request command to MG1. MGC sends Add Request to MG2. At this time with needs to reserve resources such as required bandwidth for call setup, MGC requests from Bandwidth Manager by COPS-DEC command. The call will be established between two MGs after resources provision.

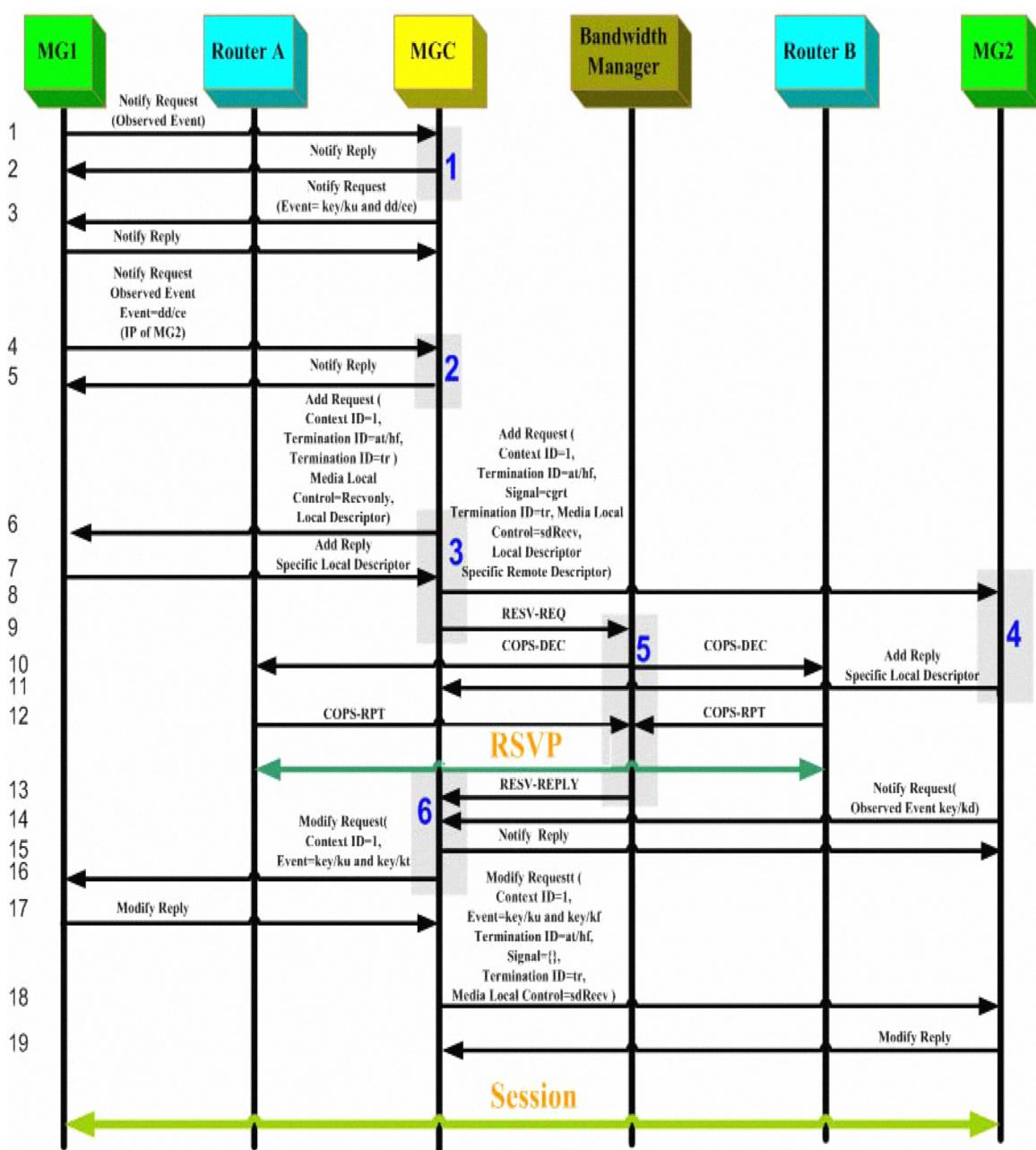


Figure 10: MEGACO actual message transaction diagram

V. MODELING THE MGC AND COPS

Most of the researches in VOIP related to MEGACO are focused on engineering principles, protocol definitions, enhancement and other improvement for example. A few research works are done in the area of performance modeling of MEGACO. Lots of IP telephony industries have been focusing on various MEGACO and MG performance metrics. Here we propose analytical M/M/1 models which there are 8 successive queues in order to process various MEGACO packets with deterministic service time. They primarily considered the mean response time and the mean number of calls in the system to analyze performance and reliability of MEGACO protocol. Mean response time of a MGC is the difference between the time it takes for an NOTIFY Request sent from MG1 to reach MGC until the final response is sent by MGC to MG1. Mean number of calls is defined as the mean number of sessions that are currently in the system. They modeled a MEGACO as an open feed forward queuing network. The MGC is modeled as queuing with eight queues. Mean number of calls N (random variable) in the system at study state is given by the system at study state is given by currently in the system.

Refer to“(1), (2)” Mean number of calls N (random variable) in

$$N = \sum_{k=1}^J \rho_k / (1 - \rho_k) \quad \text{Where } \rho_k = \lambda_k / \mu_k \quad (1)$$

$$\lambda_j = \sum_{k=1}^{j-1} (\lambda_k Q[k, j]) \quad \text{for } 1 < j \leq J \quad (2)$$

$J=8$ is the number of stations in the queuing model. Q is the one step probability matrix corresponding to the queuing model; that is, $Q[i,j]$ the probability that a job departing station i goes to station j . The mean response time for calls is by Little's law $R=N/\lambda$. they assumed the service rate is fixed at 0.5ms^{-1} and the arrival rate at 0.3ms^{-1} , Alouf et al [6,7,8].

This architecture is a developed model based on the limited capacity of the single server queues for estimating the buffer size and studying the intensity of the generated traffic.

VI. M/M/1 BASED MGC AND COPS MODEL

We consider the analytical model proposed by Gurbani et al [6] and its consequences to assess the advantages of queuing model. In this work, the MGC and COPS is modeled as an open feed-forward queuing network with incoming NOTIFY Request message from MG1 as arrival calls, and there are sequence of eight M/M/1 based queuing stations that corresponds to each MEGACO message as shown in Figure 11.

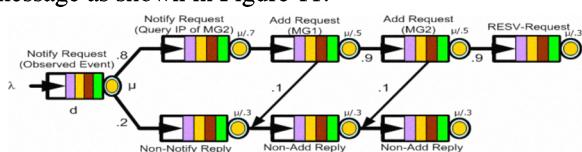


Figure 11: MGC and COPS M/M/1 Queuing Model

Only 80 percent of the Notify Request messages will be successful in getting the Notify Reply (Observed Event) and 90 percent of those Add Request (MG1, MG2) responses will get the Add Reply (MG1, MG2) responses. Remaining 20 percent of the Notify Request message will get a non- Notify Reply and 10 percent of the Add Request (MG1, MG2) responses will receive an Add Reply (MG1, MG2) responses.MG. During preparation for the next call, the MGC sends to modify request (MG1, MG2) thus MGC can setup a new call between MG1 and MG2 providing that there is a request for this propose.

VII. CALCULATED RESULTS IN MEGACO NETWORK

We consider former assumptions mentioned in this paper and assume $\mu = 0.5$, in order to calculate system's mean response time with publication delay varying between 0 - 10 ms where the distance between MG and MGC is 0 - 1000 miles (Figure 12,13). each 100 miles is assumed to be equivalent with 1ms delay. As one can see the mean response time with variation of arrival rate is approximately linear. The mean response time increases with publication delay.

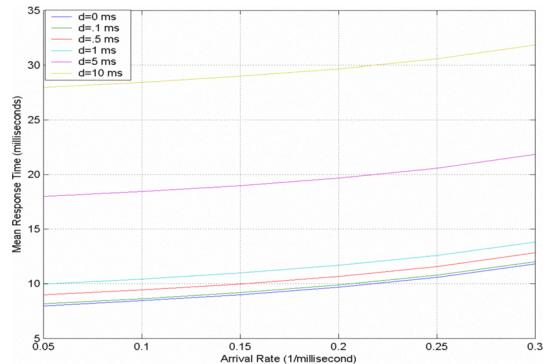


Figure 12: Mean response time base on arrival rate changes with different delays

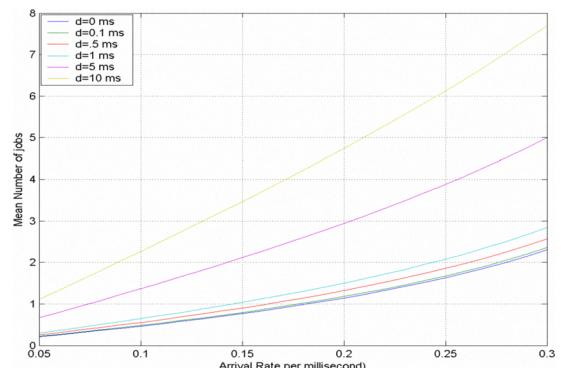


Figure 13: Mean number of jobs base on arrival rate changes with different delays

Mean number of jobs based on service rate changes with different delays are shown in Figure13. As we observe the maximum average number of jobs is less than 10 in maximum distance of 1000 miles. the results of Mean

number of jobs and mean response time are obtained by changing base service rate delay while ($\lambda = 0.3$) is supposed constant (Figure 14).

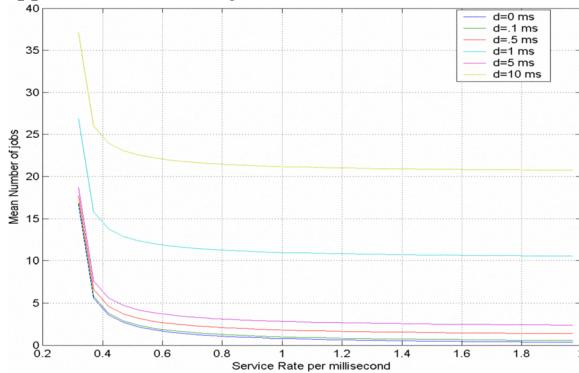


Figure 14: Mean number of jobs base on service rate changes with different delays

VIII. CONCLUSION

Based on the measurements and analysis, we have modelled Bandwidth management scenario in MEGACO with M/M/1 queuing model. With the predicted and experimental results, we has shown that the average response time, mean number of calls and server utilization factor of the M/M/1 model can produce a more predictable model with significant performance improvements and also met the ITU-T standards. In future, we intend to extend our work with considering multiple Media Gateway Controllers located in various remote locations and carrying out a comparative study of

performance effects when network delays are introduced into these models.

ACKNOWLEDGMENT

This research was support by Iran Telecommunication Research Center (ITRC). The authors would like to thank this institute for its financial support.

REFERENCES

- [1] B.S. De, P.P. Joshi, V. Sahdev and D. Callahan, "End-to-End VoIP Testing and the Effort of QoS on Signaling," IEEE Computer, Vol. 35, No. 5, Pages 80-87, May 2002.
- [2] J.M. Lennox, "Services for IP telephony," Ph.D. dissertation, Columbia University, New York, 2004.
- [3] T. Taylor, "Megaco/H.248: a new standard for media gateway control," IEEE Communications Magazine, pp. 124-132, October 2000.
- [4] "Media Gateway Control (Megaco)," Alcatel Executive Briefing, Alcatel Internetworking, December 2001.
- [5] N. Greene, M. Ramalho, and B. Rosen, "Media Gateway Control Protocol architecture and requirements," RFC 2805, April 1999; <http://www.ietf.org/rfc/rfc2805.txt> (accessed in February 2003).
- [6] V.K.Gurbani, L. Jagadeesan, V.B. Mendiritta, "Characterizing the Session Initiation Protocol (SIP) Network Performance and Reliability," ISAS 2005: LNCS 3694, pp. 196-211, April 2005.
- [7] William J. Stewart, "Probability, Queuing Models and Markov Chains: The tools of System Performance Evaluation," Lecture Notes, Department of Computer Science, North Carolina State University, Raleigh, USA, Fall2003.
- [8] I. Adan, J. Resing, "Queuing Theory," Class notes, Department of Computer Science and Mathematics, Eindhoven University of Technology, The Netherlands, February 2001.