

Performance Analysis of Multi-party Conferencing in IMS using Vacation Queues

G. Mishra, S. Dharmaraja and S. Kar

Bharti School of Telecommunication Technology
and Management, IIT Delhi,

Hauz Khas, New Delhi 110016 INDIA

garima.misra80@gmail.com, dharmar@maths.iitd.ac.in and subrat@ee.iitd.ac.in

Abstract—The IP multimedia subsystem provides a platform for group communication services using Session Initiation Protocol (SIP). SIP supports the services where participation of multiple users is involved. The centralized conferencing framework limits the ability to have large number of participants because of the overhead on the server. In this paper, an efficient solution for enhancing the capability of the system to handle multiple requests at a time is provided by presenting a queueing model for SIP message aggregation at the edge-proxies. This queueing system is analysed as a vacation queue with batch arrival and batch service to describe the arrival and processing of bulk SIP messages. The steady-state results for QoS-aware SIP signaling system are obtained from the queueing model. Further these results are verified using NS-2 simulation. In addition, the optimal value of the timer and threshold values that dynamically controls the SIP message queue associated with QoS constraints are obtained using numerical optimization technique.

Keywords: IP multimedia subsystem, Next Generation Networks, session initiation protocol, multi-party conferencing, vacation queueing, batch arrival and batch service, Markov chain

I. INTRODUCTION

The deployment of IP multimedia subsystem (IMS) provides a common multi-media service platform for IP-based network generation networks. IMS uses SIP to support communication sessions with multiple participants [1]. Initially Internet Engineering Task Force (IETF) MMUSIC working group developed session initiation protocol (SIP) for large-scale multimedia conferencing. Conference, which requires multi-party sessions are one of the specialized services supported by IMS. Multi-party sessions enable the conversational exchange of multimedia content for a wide variety of applications such as audio/ video conference, multi-player gaming, distance learning etc. [2].

SIP supports both centralized and decentralized frameworks of multi-party communication. A loosely coupled conference model is a decentralized one which makes use of multi-cast media groups. This decentralized model is well studied in the past to overcome the limited scalability issue in the centralized mode. The paper [3] proposed an architecture based on policy based distributed management for large scale enterprises. In [4], a solution is presented to support session mobility options in the full mesh conference by extending existing abstract protocol and maps them using SIP. Most of the recent researches ([5], [6], [7]) are focused on enhancing

the capacity of conferencing using distributed model only. However, the distributed framework also has some limitations such as, it does not have full security and control over the complete network due to unavailability of central server; it increases signaling delay due to multiple functional entities (FEs) introduced.

In ([8], [9]), SIP based centralized (tightly-coupled) conferencing is considered and various conferencing architectures with their performance study are presented. In centralized conferencing architecture, a single User Agent (UA) or focus has a direct relation with each participant. Therefore, it shares many problems when the number of participants in conference increases. The centralized conferencing framework does not suit for large scale conference service due to the incapability of single server and the limited capacity of bandwidth.

In the literature mentioned above, either the decentralized framework is discussed to handle the scalability issue in the present conferencing framework or the centralized frameworks are discussed without taking system overloading under consideration. This motivates us to develop a SIP based centralized conferencing framework which is capable of handling overloading in the system. The edge-proxy is modeled as batch server which describes multiple requests aggregation at the MAC layer for bulk arrivals. The analytical modeling of the MAC layer is presented by many authors [10], [11]. The results of the proposed model are compared with that of the conservative form of SIP conferencing model in which all the aforementioned improvements are not present.

The rest of this paper is organized as follows. In Section II, the conference session establishment procedure is explained. In Section III, the queueing model is developed for the edge-proxy and the core-proxy of the SIP network. The performance analysis of the proposed model is illustrated graphically in Section IV. The simulation topology and the simulation results with analytical results comparison are explained in Section V. An optimization problem is modeled in Section VI. The paper is finally concluded in Section VII.

II. MULTI-PARTY CONFERENCING ARCHITECTURE

In this section, a framework for conferencing in 3GPP IMS is presented and an application scenario for SIP signaling message aggregation is explained [1]. Conferencing is a fundamental component of enriching multimedia services by enabling

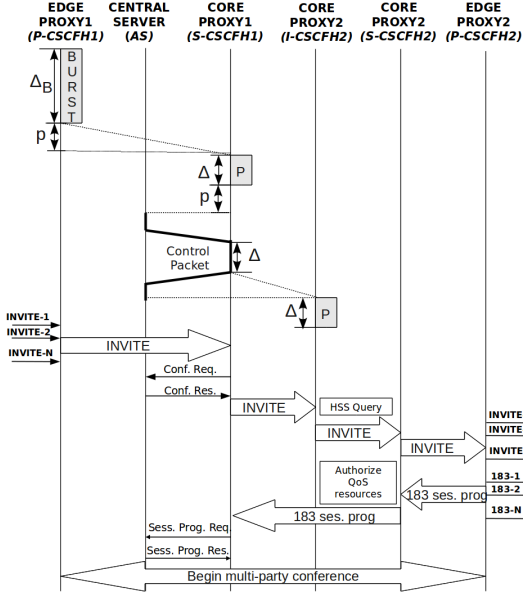


Fig. 1: Signaling flow for the conferencing session establishment

group communication. Conferencing in IMS is defined in TS 124.147 standards [12]. The primary components involved in establishing the SIP based conference sessions are the UE, the central server known as focus or application server (AS) and the SIP proxy servers (CSCFs) which includes edge-proxy (P-CSCF) and core-proxy (I-CSCF and S-CSCF).

A. Application scenario

In Fig. 1, a basic call flow and the message aggregation for creating a multi-party conference is presented.

The session establishment of a multi-party conference is also described in the following steps: Multiple initiating *UEs* try to establish a session for conference and send multiple INVITE SIP messages simultaneously to the edge-proxy of their home domain. Due to the multiple messages exchanged, which are targeted to the same receiving P-CSCF, are aggregated and formed a single INVITE request containing a list of URIs of the target *UEs*. The edge-proxy at the target home domain finally generates INVITE messages to all the target *UEs*. In response, multiple *UEs* issue multiple response messages e.g., 183(Session in Progress response). As with INVITE messages, response messages are again aggregated and forwarded as a single message to the initiating edge-proxy server.

A packet can be blocked either at an edge-proxy or core-proxy. The edge-proxy model is used to quantify the probability that a packet is blocked at an edge-buffer. In case of multi-party conferencing, the requests arrive as a batch and the destinations are in the group therefore the edge-proxy modeled as bulk queue with an exhaustive service (E) with multiple vacations (MV) can be used to provide efficient packet aggregation. A signaling timing diagram for packet aggregation and traversing from edge to core proxies is shown

in the Fig. 1.

III. EDGE-PROXY MODEL

We assume that the arrival of the SIP requests is in batches according to a time-homogeneous *Poisson* process with mean rate λ . The batch size X is a random variable and $P(X = i) = p_i, i = 1, 2, \dots, X_{max}$, mean batch size $E(X)$ and $\lambda_i = \lambda p_i$. At the edge-proxy at most B packets can be enqueued. A burst is formed by time-contiguous train of n packets where $1 \leq n \leq X_{max}$. A burst is formed at the edge-proxy as soon as n packets are enqueued. The packets marked for a common destination are aggregated into a single request and transmitted as a single entity (i.e. a batch). The length of aggregated packet transmission period is exponentially distributed with mean $\frac{1}{\mu}$. If the queue length is less than the service capacity, the server does not wait until the number of requests reaches Y_{max} , but it takes $\min(i, \text{queue length})$ customers for service. If the server does not find any request in the queue, then the system goes into the vacation phase. The time of vacation is random and exponentially distributed with rate θ . If there are at least N , or more waiting requests in the system, vacation is terminated and the server starts to serve them according to the batch service rule as explained above.

A. Steady-state Analysis

The system is modeled as a $M^X/M^Y/1/B(E, MV)$ queue and a steady state analysis of the proposed queueing model is conducted ([13], [14]). At an arbitrary time, the state of the system can be described by the following random variables:

$$\xi(t) = \begin{cases} 1 & \text{server is on vacation at any time } t, \\ 0 & \text{server is busy at any time } t, \end{cases}$$

Let $Z(t)$ denote the number of customers at any time t . The stationary distribution of the system is denoted by $\pi = (\pi_{0,0}, \pi_{1,0}, \dots, \pi_{B,0}, \pi_{0,1}, \pi_{1,1}, \dots, \pi_{B,1})$ where $\pi_{j,0} = \lim_{t \rightarrow \infty} P(Z(t) = j, \xi(t) = 0)$ and $\pi_{j,1} = \lim_{t \rightarrow \infty} P(Z(t) = j, \xi(t) = 1); j = 0, 1, \dots, B$. The steady-state equations (Eqn [1]-[2]) are constructed using state transition diagram shown in Fig. 2. The normalizing condition is given by

$$\sum_{j=0}^B \pi_{j,0} + \sum_{j=0}^B \pi_{j,1} = 1.$$

B. Core-proxy model

Packets destined for the same node are aggregated at the edge-proxy. The aggregated packets are forwarded to the next node (core-proxy or the central server). These intermediate nodes do not wait for aggregation but simply process the aggregated packets and forward them. Hence, the core-proxies and the central server of the system are considered to be as $M/M/1$ Markovian single server queueing system. The performance measures of these special cases shall be derived in the following section.

$$\begin{aligned}
(\lambda + \mu)\pi_{k,0} &= p_1\lambda\pi_{k-1,0} + \dots + p_k\lambda\pi_{1,0} + \mu\pi_{k+Y_{max},0}; k = 2, \dots, X_{max} - 1 \\
(\lambda + \mu)\pi_{k,0} &= p_1\lambda\pi_{k-1,0} + \dots + p_{X_{max}}\lambda\pi_{k-X_{max},0} + \mu\pi_{k+Y_{max},0} + \theta\pi_{k,1}; \\
&\quad k = N, \dots, B - X_{max} \\
(p_1\lambda + \dots + p_{X_{max}-1}\lambda + \mu)\pi_{B-X_{max}+1,0} &= p_1\lambda\pi_{B-X_{max},0} + \dots + p_{X_{max}}\lambda\pi_{B-2X_{max}+1,0} + \mu\pi_{B-X_{max}+1+Y_{max},0} \\
&\quad + \theta\pi_{B-X_{max}+1,1}; k = B - X_{max} + 1, \dots, B - X_{max} + Y_{max} \\
&\quad \vdots \\
(p_1\lambda + \dots + p_{Y_{max}}\lambda + \mu)\pi_{B-Y_{max},0} &= p_1\lambda\pi_{B-Y_{max}-1,0} + \dots + p_{X_{max}}\lambda\pi_{B-Y_{max}-X_{max},0} + \mu\pi_{B-2Y_{max},0} \\
&\quad + \theta\pi_{B-Y_{max},1} \\
(p_1\lambda + \dots + p_{B-k}\lambda + \mu)\pi_{k,0} &= p_1\lambda\pi_{k-1,0} + \dots + p_{X_{max}}\lambda\pi_{k-X_{max},0} + \theta\pi_{k,1}; k = B - Y_{max} + 1, \dots, B.
\end{aligned} \tag{1}$$

Equations below stand for the probability that a node goes on vacation until it finds at least X_{max} number of customers present in the system at a vacation completion instant.

$$\begin{aligned}
\lambda\pi_{0,1} &= \mu(\pi_{1,0} + \dots + \pi_{2,0} + \pi_{Y_{max},0}) \\
\pi_{k,1} &= p_1\lambda\pi_{k-1,1} + \dots + p_k\lambda\pi_{0,1}; k = 1, 2, \dots, N - 1 \\
(\lambda + \theta)\pi_{k,1} &= p_1\lambda\pi_{k-1,0} + \dots + p_{X_{max}}\lambda\pi_{k-X_{max},0}; k = N, \dots, B - X_{max} \\
(p_1\lambda + \dots + p_{X_{max}-(k-(B-X_{max}))}\lambda + \theta)\pi_{k,1} &= p_1\lambda\pi_{k-1,0} + \dots + p_{X_{max}}\lambda\pi_{k-X_{max},0}; k = B - X_{max} + 1, \dots, B
\end{aligned} \tag{2}$$

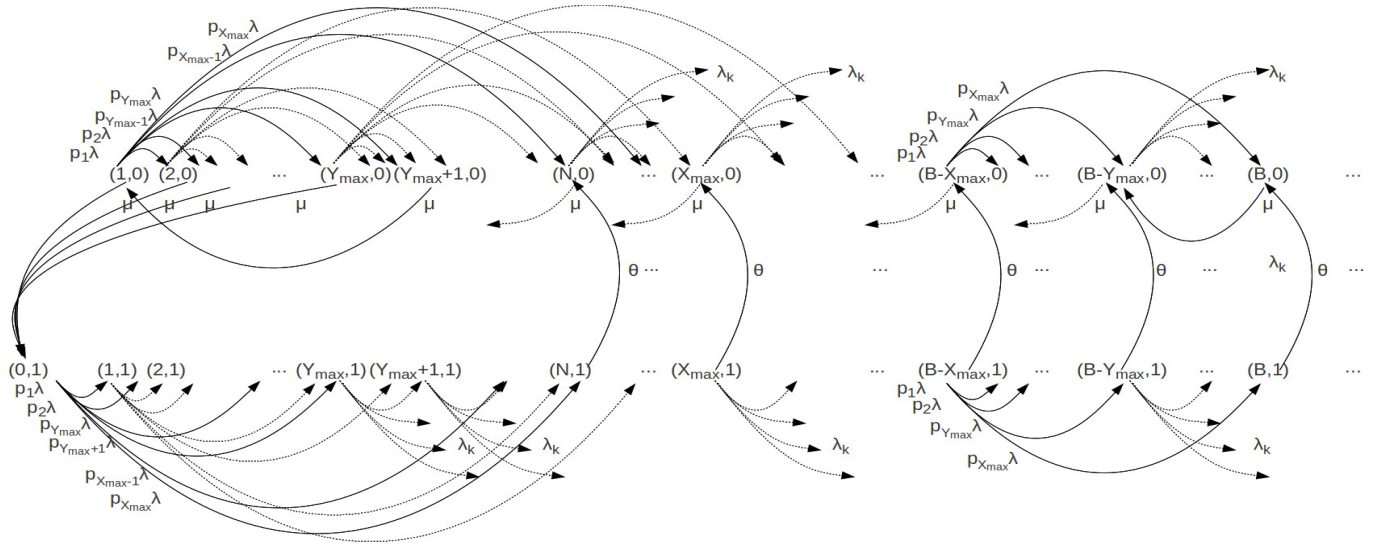


Fig. 2: State transition diagram for the $M^X/M^Y/1/B(E, MV)$ queueing model

C. Performance measures and Simulation model

Once the steady state probabilities are obtained, the packet loss rate (PLR) and the bandwidth utilization (BU) can be derived in terms of state probabilities. At anytime instant, if the available queue capacity is not sufficient for a new batch of SIP requests, the request gets blocked. Each arriving batch of requests contains k packets, $k = 1, 2, \dots, X_{max}$. The PLR of the model can be given as

$$PLR = \frac{1}{\lambda} \sum_{i=0}^{X_{max}-1} \sum_{k=i+1}^{X_{max}} \lambda_k (\pi_{B-i,0} + \pi_{B-i,1}), \tag{3}$$

where $1 \leq X_{max} \leq B + 1$. In steady-state, the BU is expressed as

$$BU = \frac{1}{Y_{max}\mu} \sum_{k=1}^{X_{max}} k\lambda_k \left(1 - \sum_{i=B+1-k}^B (\pi_{i,0} + \pi_{i,1})\right). \tag{4}$$

Thus, the average queue length becomes

$$L_q = \sum_{j=(Y_{max}+1)}^B j(\pi_{j,0} + \pi_{j,1}). \tag{5}$$

$$W_q = L_q / \sum_{k=1}^{X_{max}} k \lambda_k (1 - \sum_{i=B+1-k}^B (\pi_{i,0} + \pi_{i,1})). \quad (6)$$

Where $\sum_{k=1}^{X_{max}} k\lambda_k(1 - \sum_{i=B+1-k}^B (\pi_{i,0} + \pi_{i,1}))$ is the effective mean arrival rate (λ_{eff}) to the system. For $\rho = \frac{\lambda_{eff}}{Y_{max}\mu} < 1$, the stationary average waiting time W_q can be decomposed into the sum of two independent random variables, $W_q = W + W_v$, where W is the average waiting time of a classical $M^X/M^Y/1/B$ queueing system without vacation and W_v is the additional average waiting time due to the vacation [14].

IV. PERFORMANCE ANALYSIS

To evaluate the performance of the system, various measures are presented for a combination of different input parameters. First, the steady-state probabilities are calculated iteratively using successive over relaxation (SOR) method. Further, analysis is made for a given set of input parameters: $\lambda = 0.5$ to 4 , $\frac{1}{\theta} = 0$ to 20 , $\mu = 6$, $B = 100$, maximum batch arrival size $X_{max} = 10$, maximum batch service size $Y_{max} = 5$.

A. Analytical results

It is observed in Fig. 3(a) that as buffer size increases the blocking probability decreases and asymptotically approaches to zero. This is due the fact that the model behaves as an infinite buffer queue. Fig. 3(b) shows that as mean vacation time ($\frac{1}{\theta}$) is increased the blocking probability also increases but becomes constant after certain point when queue length reaches its maximum capacity. It is also seen that the blocking probability increases with large variation for higher arrival rates but small variation for lower arrival rates. Fig. 4(a) shows that as $\frac{1}{\theta}$ increases the average queue length L_q and mean waiting time W_q also increases. It is also seen that as we increase the average arrival rate, the W_q increases with large variation. We have also realized the effect of average batch size $E[X]$ on various traffic load on the system. It is observed in Fig. 4(b) that for larger values of ρ there is not much effect of increasing average batch size $E[X]$ but it has a significant influence on L_q for smaller values of ρ .

The total delay W_q of a user request consist of two parts: the first part W_1 is the time needed to serve the older batches and the second W_2 is the time needed starting at the end of the first part, until the batch containing the request reaches its limit i.e. X_{max} . It is observed from the Fig. 6(b) that we have different optimum points for different vacation times. It is because for larger vacation times, W_q first decreases with increase in ρ as W_1 is less for lower arrival rates. But at certain point when ρ reaches a value where W_1 starts increasing, this leads to increase in W_q . For smaller vacation times W_1 is always decreasing which leads decreasing W_q .

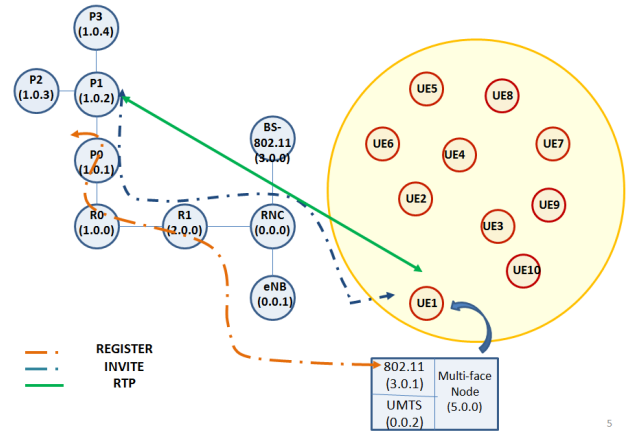


Fig. 5: Simulation topology in NS2 for multiple UEs connected to the IMS using SIP

TABLE I: Wired link parameters

Network type	Network entities	Duplex/Simplex link BW	Link delay
Access network (UMTS)	NodeB - RNC	622 Mbit/s	0.2 ms
	RNC - NodeB	622 Mbit/s	0.2 ms
	RNC - SGSN	622 Mbit/s	0.4 ms
	SGSN - GGSN	622 Mbit/s	10 ms
	GGSN - P-CSCF	10 Mbit/s	15 ms
IMS core network	P-CSCF - I-CSCF	10 Mbit/s	10 ms
	I-CSCF - S-CSCF	10 Mbit/s	10 ms
	P-CSCF - S-CSCF	10 Mbit/s	10 ms
	S-CSCF - AS	10 Mbit/s	10 ms

V. SIMULATION MODEL

We simulate the proposed multi-party conference in NS-2 (Network Simulator, version 2) [15]. The simulator model captures complete end-to-end SIP message exchange through IMS core network to establish a session.

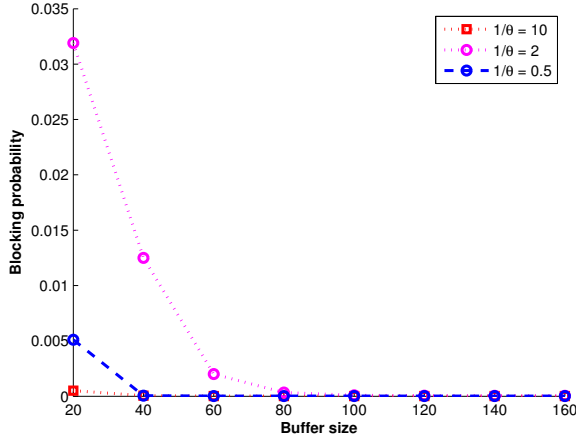
A. Simulation topology

This simulated scenario as shown in Fig. 5 describes a certain number of clients simultaneously connected to the home IMS domain through UMTS access network (AN). The home IMS domain is designed to comprise one P-CSCF, I-CSCF, S-CSCF and one conferencing AS. It is assumed that there are 10 UEs initiating a session for conference service simultaneously. Each UE sends a conference request to the 5 participants. It is assumed that all the participating clients connect to the IMS domain through the same AN and the same P-CSCF. Such assumption is made in order to ensure and investigate the performance of the network entity under maximum load condition.

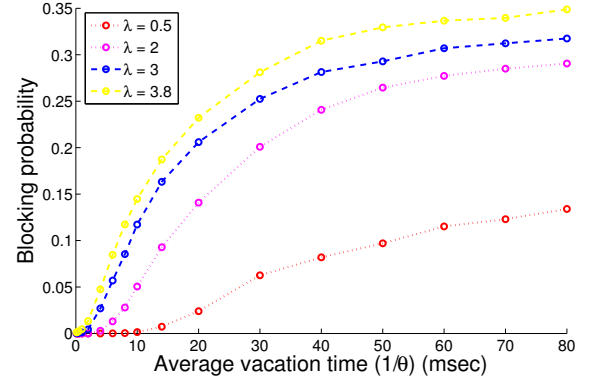
The link bandwidth and the propagation delay configured as shown in Table I. In the following section the simulated results are compared with the analytical results.

B. Simulation results

A comparative study is performed between our proposed model Fig. 6(a) and traditional model $M^X/M/1/B(E, MV)$ queue without batch service Fig. 6(b). From both the figures

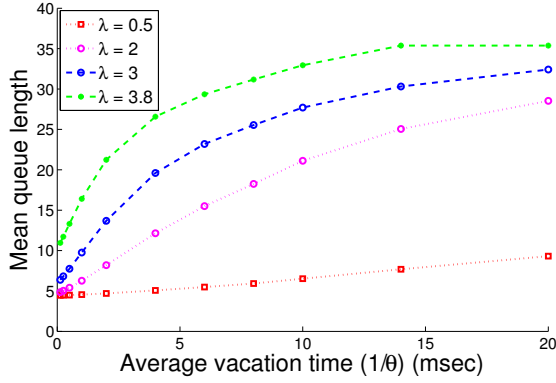


(a) Blocking Probability on varying buffer size for different mean vacation times ($1/\theta$)

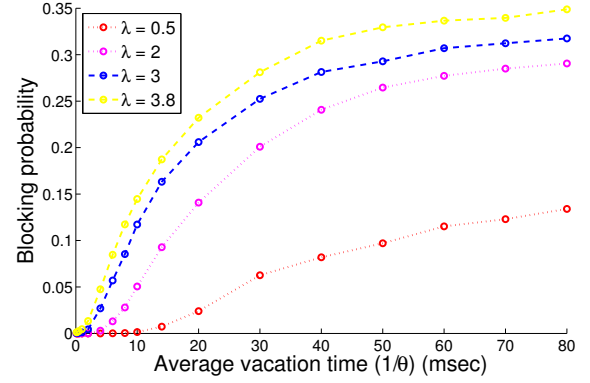


(b) Blocking Probability on varying mean vacation time for different arrival rates (λ)

Fig. 3: Blocking Probability

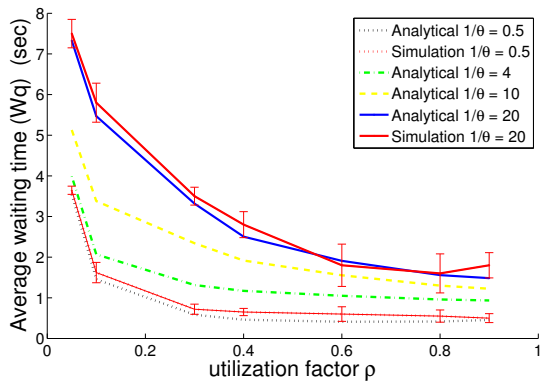


(a) Effect of mean vacation time on mean queue length for different arrival rates (λ)

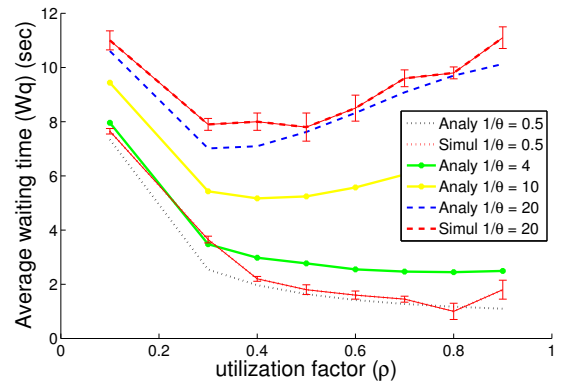


(b) Effect of ρ on mean queue length for various values of average batch size

Fig. 4: Mean Queue length



(a) Effect of traffic load on mean waiting time for the model with bulk arrival and bulk service (Proposed model) for different vacation times ($1/\theta$)



(b) Effect of traffic load on mean waiting time for the model with bulk arrival and single customer service (Traditional model) for different mean vacation times ($1/\theta$)

Fig. 6: Mean waiting time

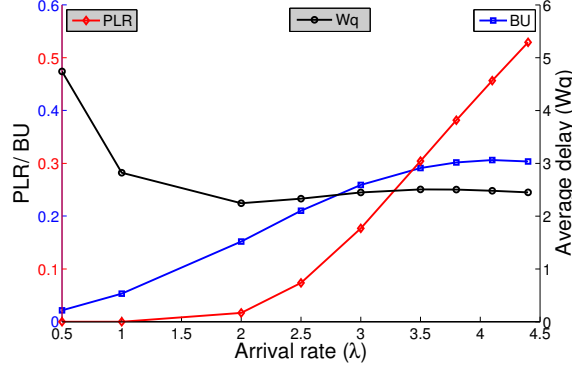


Fig. 7: Relationship among packet loss rate PLR , bandwidth utilization BU and average waiting time Wq

we observed that the mean waiting time is less in case of batch service model as compared to the model with only arrivals in batches but single customer service at a time which is the traditional model. From the above results we find that our proposed model is better for delay sensitive applications. With the bulk arrival and bulk service assumption the model works better for higher arrival rates. It is also seen in the results, when the mean vacation time ($\frac{1}{\theta}$) is increased to a certain value (according to measurement $1/\theta \geq 10\text{sec}$), the mean waiting time Wq increases rapidly.

VI. OPTIMIZATION PROBLEM

When designing a SIP edge-proxy, operator needs to determine optimal values for vacation timer ($\frac{1}{\theta}$) and queue threshold N to control the message queue for obtaining maximum bandwidth utilization for a given QoS constraints. Under constant load, a lesser θ and higher N implies higher Wq , meaning that timer value and threshold value are to be reduced to have smaller delays in the network. The performance measures, PLR , BU and Wq are numerically analysed and traced together to study their joint behavior. From the Fig. 7 it is seen that at some constant θ as the traffic intensity ρ is increased the BU also increases. The BU increases rapidly for higher ρ , but also larger Wq and PLR values.

Accordingly, given queueing delay (CWq) and packet loss rate ($CPLR$), the goal is to get optimal $1/\theta$ and N values so that the BU is maximized for various traffic loads. The objective function and the condition for the optimization problem is as follows. Given λ , μ , CWq and $CPLR$, determine optimal θ and N so that

$$\begin{aligned} \text{Maximize } BU &= \frac{1}{Y_{max}\mu} \sum_{k=1}^{X_{max}} k\lambda_k \left(1 - \sum_{i=B+1-k}^B (\pi_{i,0} + \pi_{i,1})\right), \\ \text{subject to } \frac{1}{\lambda} \sum_{i=0}^{X_{max}-1} \sum_{k=i+1}^{X_{max}} \lambda_k (\pi_{B-i,0} + \pi_{B-i,1}) &\leq CPLR; \\ L_q / \sum_{k=1}^{X_{max}} k\lambda_k \left(1 - \sum_{i=B+1-k}^B (\pi_{i,0} + \pi_{i,1})\right) &\leq CWq; \end{aligned}$$

VII. CONCLUSION

Networks, where multiple user participation is involved, traffic arrives in bulk and corresponding servers get overloaded causing very large session setup delay or even termination of the new as well as ongoing sessions. In this paper, we present a queueing model to study the system behavior in case of bulk arrivals. In a multi-party conference, the setup time is much longer than in a point-to-point application because of the multiple UEs invited. The analytical results obtained from the queueing model are verified with the results obtained from NS-2 simulations. As a conclusion, we propose that bulk arrivals are serviced in batches and if the server finds an empty system, it goes on vacation. Further the analysis shows that the session setup time is reduced and it gives us better performance when compared to the system where customers are serviced one at a time.

REFERENCES

- [1] G. Camarillo and M. A. Garcia-Martin, *The 3G IP Multimedia Sub-system (IMS): Merging the Internet and the Cellular Worlds*, 3rd ed. Wiley, 2008.
- [2] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, *RFC 3261 - SIP: Session Initiation Protocol*. IETF, 2002.
- [3] Y.-H. Cho, M.-S. Jeong, J.-W. Nah, W.-H. Lee, and J.-T. Park, "Policy-based distributed management architecture for large-scale enterprise conferencing service using SIP," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 10, pp. 1934–1949, Oct. 2005.
- [4] W. Elleuch, A. Houle, and S. Guenette, "Enabling session mobility in full mesh conferencing model," in *Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct. 2007, pp. 49–49.
- [5] A. Knauf, G. Hege, T. C. Schmidt, and M. Wählisch, "A virtual and distributed control layer with proximity awareness for group conferencing in P2PSIP," in *Principles, Systems and Applications of IP Telecommunications (IPTComm)*. ACM, Aug. 2010, pp. 122–133.
- [6] T. C. Schmidt, G. Hege, M. Wählisch, H. L. Cycon, M. Palkow, and D. Marpe, "Distributed SIP conference management with autonomously authenticated sources and its application to an H.264 videoconferencing software for mobiles," *Multimedia Tools and Applications*, vol. 53, no. 2, pp. 349–370, Jan. 2011.
- [7] X. Chen, M. Chen, B. Li, Y. Zhao, Y. Wu, and J. Li, "Celerity: A low-delay multi-party conferencing solution," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 9, pp. 155–164, 2013.
- [8] K. Singh, G. Nair, and H. Schulzrinne, "Centralized conferencing using SIP," in *Internet Telephony Workshop*, vol. 7, Apr. 2001, pp. 57–63.
- [9] T. A. Le, H. Nguyen, and N. Crespi, "IMS-based distributed multimedia conferencing service for LTE," in *IEEE Wireless Communications and Networking Conference (WCNC)*, April 2012, pp. 3210–3214.
- [10] Z. Shi, C. Beard, and K. Mitchell, "Analytical models for understanding misbehavior and mac friendliness in csma networks," *Performance Evaluation*, vol. 66, no. 9, pp. 469–487, 2009.
- [11] Z. Shi, C. C. Beard, and K. Mitchell, "Misbehavior and mac friendliness in csma networks," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2007, pp. 355–360.
- [12] J. Rosenberg, *RFC 4353 - A Framework for Conferencing with the Session Initiation Protocol (SIP)*. IETF, Feb. 2006.
- [13] G. Bolch, S. Greiner, H. de Meer and K. S. Trivedi, *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*, 2nd ed. Wiley, 2006.
- [14] N. Tian and Z. G. Zhang, *Vacation queueing models: Theory and Applications*, 2nd ed. Springer, 2006.
- [15] S. McCanne, S. Floyd, K. Fall, and Varadhan, "NS2 (network simulator 2)." [Online]. Available: <http://www.isi.edu/nsnam/ns/>