Proceedings of the 2015 Winter Simulation Conference
L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, eds.

MODELING AND SIMULATION APPLIED TO LINK DIMENSIONING OF STREAM IP TRAFFIC WITH INCREMENTAL VALIDATION

Edson L. Ursini
Paulo S. Martins
Varese S. Timóteo
Flávio R. Massaro Jr

School of Technology
University of Campinas – Unicamp
Limeira, SP, 13484-332, BRAZIL

ABSTRACT

Modern networks for converged services (voice, video and data) require appropriate planning and dimensioning. In this paper, we present a methodology for dimensioning the link capacity and packet delay in stream, IP multi-service networks with QoS requirements, in which discrete-event simulation is essential. The model may be used in the lack of enough reliable real-world data, since it is initially validated by an analytical model and then augmented step by step. The approach can be made more reliable if measured values are used. We show that the incremental approach allows a significant reduction in simulation time without significant loss of accuracy, by exploiting the sample variance reduction due to the large difference in the time scale between events occurring in the application (service layer) and in the packet layer. We demonstrated the applicability of this method with typical multi-service network scenarios.

1 INTRODUCTION

The increased demand for network bandwidth (and its limited availability) due to the widespread use of mobile devices has increased the importance of proper dimensioning of networks. Traditional methods for network dimensioning have relied upon simulation and/or analytical approaches. However, the correct dimensioning of multi-service networks using these methods has become a challenge for both researchers and network operators due to: 1) the integration (convergence) of services and networks, 2) complex nature of modern networks (topologies, protocols, devices), 3) lack of real-world data (physical network measurements of past and current projects), 4) inherent modeling limitations: i.e. it is not possible to analytically model complex and large networks, as it is also a challenge to build, in a single step (all-at-once), a fully-fledged validated simulation model that captures the major features of such systems.

Therefore, this paper proposes a novel approach for network dimensioning that is based on incremental validation, where discrete event simulation plays a vital role. The main features of the approach are demonstrated for the stream traffic, by dimensioning the link capacity (blocking) and packet delay (jitter). The approach was validated with a number of scenarios, as shown in the remainder of this paper. It is a general approach that can be applied to several other types of traffic and scenarios.

The key feature of the proposed simulation model lies behind the concept of incremental validation. We consider a given modern multi-service IP network (i.e. the target network) for which we wish to dimension its link capacity and packet delay. We begin by building a simple analytical model that captures the most basic features of the target network. Next, a corresponding simulation model is created and validated by the analytical model. The simulated model is evolved and augmented with several small

but well-defined increments (e.g. change of arrival distribution, addition of a new service, changing the processing server). Each newly added incremented represents one step closer to the full-fledged target network. The n-th simulation model is validated against its previous (n-1) simulation model. Each new increment also adds a relatively small change in network behavior, which is expected and validated by the analyst's previous (general) knowledge of the system behavior. For example, one given increment my change the arrival distribution of a service from exponential to Weibull. Since Weibull is a long-tailed distribution, the analyst must expect an increase in the service blocking once the incremented model is simulated. Once that behavior is observed, the incremented model may be deemed valid. The model is incremented until it matches that of the target network. The dimensioning of the target network is the final output of the approach. This result must be further validated against real data (even if it is a small dataset).

Therefore, the contribution of this paper lies in the proposal and demonstration of this novel approach, which has the advantage of being able to dimension relatively complex multi-service networks as well as the potential to significantly reduce simulation time, as demonstrated throughout the remainder of this paper. We have seen no other work reported in the literature combining discrete event simulation and incremental validation to dimension multi-service networks.

The remainder of this paper is organized as follows: In Section 2 we present a brief description of related work. Section 3 addresses the main types of scenarios and also general model characteristics. Section 4 discusses analytic and simulation traffic models. More specifically, it focuses on analytical models with call admission control (CAC) (subsection 4.1); likewise, subsection 4.2 discusses simulation models with CAC; subsections 4.3 and 4.4 present analytical and simulation models (respectively) with bandwidth control; subsection 4.5 shows simulation models for the packet layer addressing delay and jitter, and subsection 4.6 presents the incremental model regarding global sample variance reduction. Finally, in Section 5 we present the results and the main conclusions.

2 RELATED WORK

A work that is close to ours is the one by (Alves et al 2014), whereby the authors dimension a gateway. However, our work does not address the plan and design of the gateway; instead it aims at dimensioning the link capacity and packet delay for the stream traffic, such that it can be extended for several nodes. Another similar idea (Gonzalez 2007) introduces multiple user models (for 4G wireless) which are interconnected and serve different functions depending on the user model. Our models apply mainly to networks in a more general context (both wired and wireless), regarding link capacity and not specifically in a mobile context. (Kavacky et al. 2009) address discrete event simulation models of multiservice networks using admission control and priority as well as bandwidth constraints. Our paper does also apply the concept of discrete event simulation with admission control and/or bandwidth and packet delay constraints, but with the idea of model increments with different traffic characteristics. Unlike (Rani and Suganthi 2014), which discusses the specific issues of TCP-RED protocol and the optimal size of the packets using simulation with a specific free software (NS-2), our design uses the basic concepts of UDP, TCP (and simulation software) in a more suitable way for dimensioning and network planning; our approach does not rule out other models, and other protocols (including TCP-RED) may also be included in more detail if necessary. In (Perény et al. 2009), the authors used the NS-2 software but have not addressed the concept of incremental validation. Concerning stream services with the P2P context, (Walkowiak 2010) shows that packet layer statistics of P2P and non-P2P data flows are basically similar; these services can also be incrementally modeled by our method in a more flexible way. However, a more comparison analysis is outside the scope of this paper.

3 SCENARIO DESCRIPTION AND GENERAL CHARACTERISTICS

To apply the QoS concept the traffic is separated into two main kinds: stream traffic and elastic traffic. Type stream tolerates some type of loss (with some bandwidth requirements), but does not tolerate delay

and the elastic tolerate some delay, but does not tolerate loss (Trindade et al. 2003). Aiming to evaluate the behavior of a multi-service data network in terms of QoS, with respect to variations in traffic demand, the proposal is to use a scenario compatible but, in this work, focusing on traffic stream. Given the particularities of the market and of each organization, was selected a set of elements considered to comprise the typical topology of the hypothetical network communication. Particular application will have its own characteristics but certainly the approach for the design / planning may follow similar steps.

Additionally, if the link design is to be implemented, the solution can provide a reduced cost since the interconnection among the elements of access can be wireless and uses VoIP (data packets, in general) technology. Moreover, it can also use free software. Table 1 shows an example of a typical network scenario. Fig. 1 shows the scenario which is served by these components, including elastic Data files and illustrates a link in which all services are routed by a single link with arrival rate λ and service rate μ. From the point of view of telecommunications, it is sufficient for the intended purposes.

3.1 Traffic Characterization per Stream Service

Table 1 shows the simulated scenarios with five different types of stream services, including their arrival rates and duration. The elastic traffic is derived from data files, text or pictures, www pages with very varied sizes (Salvador et al. 2004). We assume that file size is a random variable with either Pareto or hyperexponential distribution. The analytic model for the elastic traffic uses a M/G/R/PS due to the TCP protocol and the models can be seen in (Tome et al. 2008; Trindade et al. 2003; Ursini et al. 2014; Riedl, Bauschert, Perske, and Probst 2000). The focus of this paper is on the stream traffic.

3.1.1 Stream Traffic Characteristics

1. Each individual VoIP service is a PABX set that corresponds to 5.0 Erl; assuming that the services generated by the VoIP and Videoconference have similar characteristics to the circuit-switched voice service, the duration has exponential probability distribution (180 and 900 s, respectively), (Sharma 1997), and the number of arrivals in a given time interval follows a Poisson distribution;

2. Video-On-Demand and Video-Clip services also have requests (i.e. arrivals) distributed exponentially. Their duration of service distributions are Gaussian: Video-On-Demand with N (7200,900) and Video Clips with N (300,100), in seconds;
3. The model assumes that the IP Camera requires a 32 kbit/s bandwidth (codec MPEG-4) to send one frame per second, in a deterministic way..

Table 1: Traffic characterization per service.

Service	Arrivals (Busy Hour)	Duration(s)	Traffic (Erl) ρ_i
Video On Demand	0.25 calls/h - Exponential	7200 - Gauss	0.50
Videoconference	1 call/h - Exponential	900 – Expon.	0.25
Video Clips	4 calls/h - Exponential	300 - Gauss	0.33
VoIP (PABX set)	100 calls/h - Exponential	180 - Expon.	5.00
IP Camera	1 frame/s - Deterministic	1- Determ.	1.00

3.1.2 General Model Characteristics

It is known that, in practice, there are scenarios (featuring a one-way link) where there are multiple nodes, various types of transmission channel and different routing schemes (Fig. 1). However, it is not difficult to assess the impact of the total delay on the QoS considering multiple hops that occur from source to destination. Taking the mean and standard deviation of the delay for each link, it is possible to approximate a Gaussian distribution for more than three hops (Karam and Tobagi, 2001). For a given

scenario (link capacity and number of network elements), we need an analytical model to assess whether the QoS requirements (e.g. 2% blocking) are met by the traffic load generated by the services shown in Table 2. Case A and B represent two scenarios and hence two different traffic loads. Based on (Trindade et al. 2003), the dimensioning is initially performed by two independent models, one for elastic traffic and another for the stream, so that one portion of the total link capacity is allocated for the elastic traffic and the other for the stream traffic.

4 TRAFFIC MODELS

In early packet network services, all the traffic was elastic and had no severe time constraints for delivery. The stream type services are the ones that are currently developing in large scale (VoIP, Videoconference, Video On Demand, etc.). Thus, it becomes important to study appropriate ways to dimension/and plan this type of traffic. We assume that the QoS requirement (blocking) for the stream traffic stream should be less than (or around) 2% for each service. The stream traffic also requires a minimum amount of available bandwidth. Considering Table 1 (traffic per service) and Table 2 (case A), the total traffic is ρ_1 = 0.5 X 10 = 5.0 Erl for Video On Demand, ρ_2 = 0.25 X 10= 2.5 Erl for Video Conference, ρ_3 = 0.33 X 10 = 3.3 Erl for Video Clips, ρ_4 = 5.0 X 14= 70.0 Erl for VoIP (each PABX set) and ρ_5 = 1.0 X 10 = 10.0 Erl for IP Camera. IP camera arrivals are set to exponential (as with the analytical model) to validate the simulation model. Due to the PASTA property (Poisson Arrival See Time Average), it is possible to evaluate the blocking for each type of traffic. The set of CODECs being considered uses 13 kbit/s to VoIP (c_4) and 32 kbit/s for the other services (c_1, c_2, c_3 and c_5). Leaving the link without any constraint, (i.e. without both CAC and bandwidth limitation), our simulation showed that to carry out the offered traffic load, more than 20 Mbit/s bandwidth is required in this case. Thus, in the following sections we address some mechanisms for limiting the offered traffic, and models to evaluate the packet delay and the jitter.

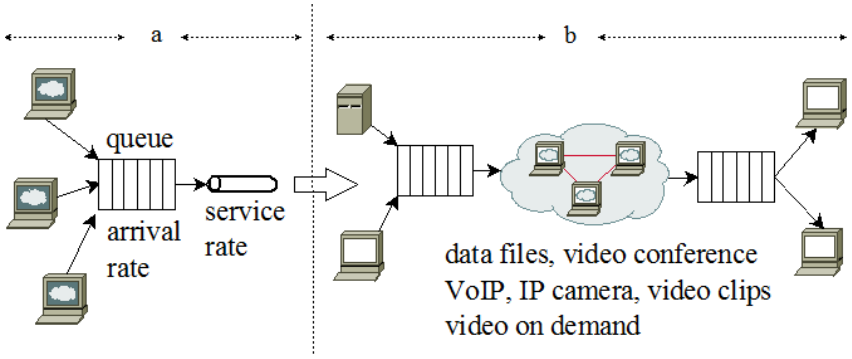

Figure 1: a) Multi-service network topology as a single link. b) Multi-service packet switch network topology.

Table 2: Stream traffic scenario description.

Network elements	# of services (case A)	# of services (case B)
Vídeo On Demand	10	10
Videoconference	10	10
Video Clips	10	10
VoIP (PABX set)	14	6
IP Camera	10	5

4.1 Analytical Model Based on Call Admission Control (CAC)

Typically, the Internet Protocol does not allow QoS and therefore it is not possible to separate the different types of traffic (in the application layer) to perform a CAC. However, a simple gateway (traffic concentrator) was implemented using the IP-ToS field in the IPv4 PDU (or Traffic Class field in IPv6 PDU) in order to separate the types of traffic (QoS Diffserv) (Diório et al. 2014). This approach assumes the use of CAC in the application layer. Therefore, Table 3 shows the maximum number of allowed services and the corresponding blocking by using the Erlang B = $E_1(N,A)$ formula. For instance, for Video On Demand, $E_1(10,5.0) = 1.8\%$. As the offered traffic means the average number of busy services, it is possible to calculate the total average bandwidth required, or M = (5.0 + 2.5 + 3.3 + 10.0).32000 bit/s + 70.13000 bit/s = 1576560 bit/s.

4.2 Simulation Model Based on CAC (Application Layer)

As mentioned earlier, the main idea of our approach is to build a simulation model and then increase it to consider other aspects, using the facilities of new simulation languages (Kelton et al. 2001; Jain 1991). In order to build the simulation model, we need to consider the first requirement, which is service loss (or blocking) for the stream traffic. This can be modeled as a binomial distribution since the focus is on the number of calls or lost images (success) related to the total calls or images generated. Thus, the proportion of successes is given by $\hat{p} = \frac{\theta}{n}$, where θ denotes the total number of lost calls or images and n is the total number of calls or images generated in the execution of the simulation model (note that $\hat{p}$ is around 2% for estimating the desired blocking value). The population variance estimator is $\frac{\hat{p}(1-\hat{p})}{n-1}$ and $\beta = z_{1-\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n-1}}$, where $z_{1-\alpha/2}$ =1.96, considering α=5%, since $n\hat{p} \geq 5$ and $n(1-\hat{p}) \geq 5$ for each replication. The simulation statistics block calculates a 95 % confidence interval (CI), including all replications (i.e. $\hat{p} \pm \beta$**)**. The simulation program was run for 320000 s and with 10 replications. The transient period was approximately 10% of the total duration (Freitas Filho 2001). The first step is to validate the analytical model, mainly regarding the total required bandwidth (without limiting the link bandwidth value). Fig. 2 shows the simulation model. Basically, the simulation model has four sub-models: **Increase Pre-CAC** increases the required input bandwidth for a given service; **Processing** serves the arriving requests according to the FIFO discipline; **Restore Bandwidth** blocks the service above a specified maximum; and **Pre Out** restore the available bandwidth in the link after the end of service. The instances use a key global variable named CHANNEL and a local variable called T_SERVICE. At every new request, the variable CHANNEL is incremented according to the CODEC used to perform the service, and the local variable T_SERVICE is implemented according to the service duration. The number of active services is described in Table 2. A maximum number of allowed dynamic active services (n_1, n_2, n_3, n_4 and n_5) is described in Table 3 according to scenarios A and B. The limits were calculated assuming a blocking value ranging from 0% to 1.8%, with the traffic values as in Tables 1 and 2 (case A). For all exponentially-arriving services, the simulated blocking is similar to the analytical one (compare Tables 3 and 4). Also, the 95% CI for global variable CHANNEL = [1536800±85297] bit/s indicates that the 1576560 bit/s mean (analytic) value falls inside the interval. When using only the CAC constraint (i.e. without limiting link size), the required bandwidth can reach peaks of 2282000 bit/s. In the next model, the arrival rate for Video Clips is changed from exponential to Weibull, e. g., if $Y \sim \text{Weibull}(\alpha,\beta) \rightarrow E[Y] = \beta\Gamma\left(1+\frac{1}{\alpha}\right)$, but the average arrival rate of 900 s is kept while α = 1/3 and β = 300. Thus, CHANNEL = [1504100±97144] bit/s, including the analytical average value. Fig. 3 shows the blocking peak due to the change of the Video Clips service arrival to Weibull. This change alone causes a significant blocking of Video-Clip service and also a small change in blocking of the other services.

For the next test (even with the CAC in application layer), we consider a constant (reserved) stream channel capacity for the link, set at C_s = 1622097 bit/s (the upper end of the range of 95% for all exponential services) and 1874896 bit/s. Therefore, in these tests we have both band (constant value) and CAC restriction. Table 5 shows the blocking values when the link capacity is constant. It can be noted that the blocking value is larger with these bandwidth limiting conditions, and as expected, as we increase the link size, it approaches the blocking caused exclusively by admission control.

Table 3: Traffic model-Erlang blocking.

Service	**Video On Dem**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Allowed # services	10	10	8	82	17
Traffic ρ_i (Erl)	5.0	2.5	3.3	70.0	10.0
Blocking b_i (%)	1.8	0.0	1.4	1.8	1.3

Therefore, in these tests we have both band (constant value) and CAC restriction. Table 5 shows the blocking values when the link capacity is constant. It can be noted that the blocking value is larger with these bandwidth limiting conditions, and as expected, as we increase the link size, it approaches the blocking caused exclusively by admission control.

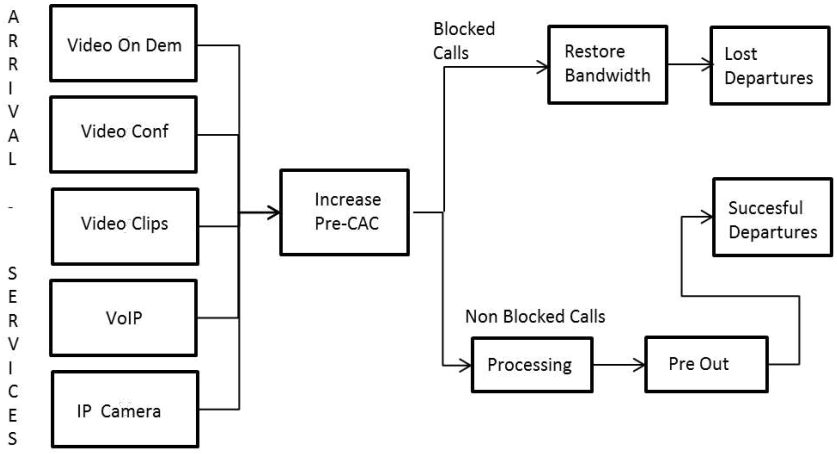

Figure 2: First simulation model for stream services.

Table 4: Traffic simulation blocking CIs (%) with CAC and unlimited channel.

Service	**Video On Dem.**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Maximum number	10	10	8	82	17
Traffic ρ_i (Erl)	5.0	2.5	3.3	70.0	10.0
All services expo	1.6±0.83	0.77±0.30	1.2±0.19	1.8±0.14	1.3±0.01
Video Clips Weibull	1.24±1.10	1.00±0.49	15.97±0.98	1.78±0.12	1.29±0.01

In most real-world cases (e.g. IP networks), there must always be a limitation on the link capacity, and they do not possess resource admission control in the application layer either.

4.3 Analytical Model with Bandwidth Control (Application Layer)

As the IP protocol does not provide admission control, in most cases we will have to calculate the blocking from both the offered traffic and the available bandwidth. When we have *m* classes of users (stream services), the general solution in closed expression for the product, according to (Labourdette and

Hart 1992), is $p(n_1, n_2, ..., n_m) = \prod_{i=1}^{m} \frac{\rho_i^{n_i}}{n_i!} \frac{1}{G}$, where G is a normalization constant. For each request (call or frame) of service i, a constant bit rate c_i is reserved (CODEC), and it is released immediately after this request. To accommodate the M traffic (bit/s) in a C_s channel (bit/s), one must set constant α, as follows:

$\sum_{i=1}^{m} c_i \rho_i \alpha^{C_i} = C_s$ or $\sum_{i=1}^{m} \frac{c_i \rho_i}{C_s} \alpha^{C_i} - 1 = 0$, where $M = \sum_{i=1}^{m} c_i \rho_i$. As there two types of CODECs, and thus two different rates, we must have an equivalent rate d (as if it were an equivalent CODEC), according to $d = \frac{\log \frac{C_S}{M}}{\log \alpha}$. The general solution to the blocking probability, allowing service i to handle traffic ρ_i, is denoted b_i and it can be approximated by $b_i \approx \frac{1-\alpha^{c_i}}{1-\frac{C_s}{M}} E_1\left(\frac{M}{d}, \frac{C_s}{d}\right)$, where $E_1\left(\frac{C_S}{d}, \frac{M}{d}\right) = \frac{\left(C_S/d\right)^{(M/d)}}{\sum_{i=0}^{(M/d)} \frac{\left(C_S/d\right)^i}{i!}}$ denotes the Erlang blocking formula, Erlang B, for a fractional number of trunks (servers). It requires a specific numerical procedure (an adequate solver) for its resolution. As an example, for M = 1576560 bit/s, C_s = 1622097 bit/s, CODECs $c_1 = c_2 = c_3 = c_5$ = 32000bit/s and c_4 = 13000 bit/s produces α = 1.00000134997, d = 21092.6 bit/s, $E_1(75.14,73.25) = 7.27.10^{-2}$, $b_1 = b_2 = b_3 = b_5$ = 11.0%, and b_4 = 4.5%. The results in Table 6 show that, for limited amounts of available bandwidth, there is no difference between using CAC or not. The blocking is larger with these bandwidth limiting conditions, but it approaches the blocking caused exclusively by admission control as we increase the link capacity.

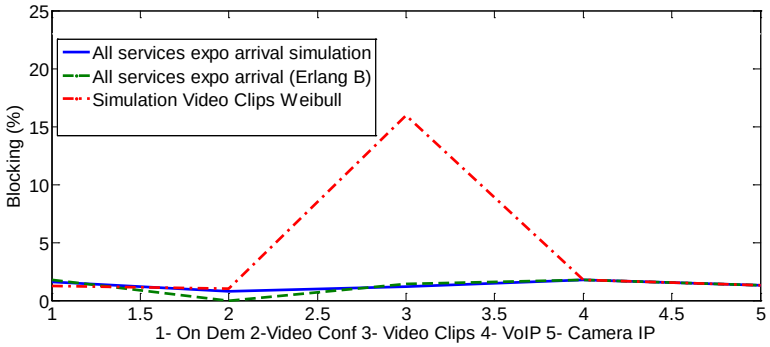

Figure 3: Simulation blocking with CAC.

4.4 Simulation Model with Bandwidth Control (Application Layer)

Using the same simulation model of Fig. 2, but with bandwidth constraints (instead of CAC), we see a significant similarity between the results of the analytical model and the simulated ones (thus validating them). Once the simulation model is deemed valid, we perform the simulation now with a constant IP Camera arrival rate of 1 sec (the PASTA property no longer holds true).

The last two lines in Table 7 show the significant difference with the new calculated values. However, in addition to bandwidth requirements, the streaming traffic has to meet the delay restrictions imposed by the receiver (de-jitter). In the next subsection, we will show a small change to the simulation model (in Processing sub-model) to estimate the one-way packet sending time (and its variation).

Table 5: Traffic simulation blocking CIs (%): using both CAC and limited channel.

Service	**Video On Dem**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Maximum number	10	10	8	82	17
Traffic ρ_i (Erl)	5.0	2.5	3.3	70.0	10.0
C_s = 1622097 bit/s	10.91±2.08	11.07±0.78	11.05±0.82	4.81±0.29	10.88±0.61
C_s = 1874896 bit/s	2.67±0.99	2.63±0.49	2.63±0.39	1.07±0.15	2.62±0.36

Table 6: Traffic analytic blockings CIs (%) and limited channel.

Service	**Video On Dem**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Traffic ρ_i (Erl)	5.0	2.5	3.3	70.0	10.0
C_s = 1622097 bit/s	11.0	11.0	11.0	4.5	11.0
C_s = 1667400 bit/s	8.5	8.5	8.5	3.4	8.5
C_s = 1874896 bit/s	2.0	2.0	2.0	0.8	2.0

Table 7: Traffic simulation blockings CIs (%) with limited channel.

Service	**Video On Dem.**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Traffic ρ_i (Erl)	5.0	2.5	3.3	70.0	10.0
C_s = 1622097 bit/s	10.78±0.87	11.32±0.84	11.02±0.76	4.5±0.30	11.19±0.66
C_s = 1667400 bit/s	8.35±1.78	8.86±0.77	8.75±0.66	3.56±0.28	8.97±0.58
C_s = 1874896 bit/s	2.18±0.78	2.18±0.43	2.08±0.37	0.79±0.12	2.17±0.30
C_s = 1874896 bit/s (*)	1.28±1.03	1.38±0.31	1.29±0.18	0.48±0.08	0.25±0.10
(*) Last line simulation with constant IP camera arrival = 1s					

4.5 Extended Simulation Model (Packet Layer)

The total delay is composed of four parts: queuing delay (*qx*), transmission delay (*tx*), processing delay (*px*) and propagation delay (*pgx*). ITU considers the network delay for voice applications in *Recommendation G. 114* and defines three one-way delay bands (Cisco, 2015). A delay less than 150 ms is acceptable to most applications. Within the 150-400 ms range, the delay becomes acceptable provided that administrators are aware of the transmission time and the impact it has on the transmission quality of user applications. It is unacceptable for general network planning purposes beyond 400 ms. We consider *Coder + Packetization + Serialization* as the processing (*px*) component. Once we have the link bandwidth calculated, we may proceed to calculate the transmission (*tx*) and queueing/buffering delay (*qx* in a G/G/1 model). The propagation delay (*pgx*) could be easily added in specific cases. The incremental simulation model is similar to that of Fi. 2, but now with 1) a single server representing the link and the buffer; 2) with the link bandwidth as a constant, and 3) with packets sent each 65 ms when the stream service is active. We consider at least three analog signal samples, one each 20 ms. This results in 1) 32000 x 20 ms = 1920 bits/packet for the CODECs with 32 kbit/s, and 2) 13000 x 20 ms = 780 bits/packet for the VoIP CODEC with 13 kbit/s. For instance, to meet QoS requirements, the combined processing, queuing, and transmission times of a packet, including de-jitter of 45 ms (*px* +*qx* + *tx*), cannot exceed 150 ms. The simulation model uses an additional functional block, called *Duplicate*, so that the instance that represents the packet be also used for the calculation of the delay budget (Fig. 4). We do not simulate packet loss (essential for elastic traffic using the TCP protocol) since, as long as it is limited to a certain level, it is not a relevant feature for the stream services using the UDP protocol. Clearly, the designer must assure that packet loss, along with delay and jitter all meet the QoS requirements. We considered, in the example, *px* = 30 ms and a 0.5 ms jitter, modeled as N (30,0.5). The *qx* and *tx* values are estimated by the model (assuming *pgx* = 0). The average link capacity should be (2.5 + 5.0 + 3.3 + 5.0) x 32000 + 30 x 13000 = 895600 bit/s (see case B in Table 2). The main increment to the previous model was to process each packet's time individually (each service is decomposed in packets), instead of

the total service time as with the former model. Assuming the link bandwidth available to be 1 Mbit/s, it is also possible to monitor service losses. Table 8 shows the values obtained for traffic values in Table 2 (case B). Fig. 5 also shows the average blocking values for the five services (1 to 5), varying at most only one arrival time distribution (while maintaining the other exponential. Note that a simple change in one service causes a disturbance in the other ones (Table 8 and Fig. 5). When arrivals are all exponential, the analytical and simulation models represent very well the average blocking (green dashed and blue solid lines). We have relatively large CIs regarding the blocking values due to relatively few observations (4 X 120000). However, due to the large number of packets in 120000 s of simulation run time, important results (and with narrow CIs for the packets delays) are obtained (Table 8). In simulation (II) the delay is 80.0 ± 0.6 ms; in (III) the delay is 82.1 ± 0.7 ms; and in (IV) the delay is 77.7 ± 1.1 ms. The run time simulation by packets is too long. For example, a simulation to evaluate blocking for the services takes approximately 588 min run time for 10 replications, with 320000 s simulation time each replication. In contrast, a simulation to evaluate blocking considering the packet layer takes approximately 5561 min run time for 4 replications, with simulation time of 120000 s for each replication. This is almost 10 times longer than the run time in the application layer (with less accuracy concerning blocking values).

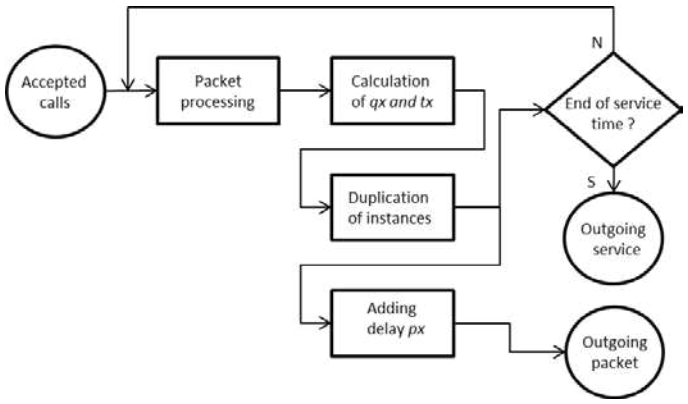

Figure 4: Processing sub-model detail.

Table 8: Traffic simulation in packet layer, C_s = 1000000 bit/s (constant).

Service	**Video On Dem.**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Traffic ρ_i (Erl)	5.0	2.5	3.3	30.0	5.0
(I) Analytic Model					
Blocking b_i (%)	8.3	8.3	8.3	3.2	8.3
(II) Simulation with all services exponential (4 X 120000 s)					
Blocking b_i (%)	9.67± 5.25	9.27 ± 0.90	9.27 ± 1.08	3.42 ± 0.33	9.20 ± 0.96
(III) Simulation with IP Camera constant = 1 s (4 X 120000 s)					
Blocking b_i (%)	8.32±8.71	12.03±3.06	11.62±3.38	4.27±1.06	2.14±0.61
(IV) Simulation with Video Clips Weibull (4 X 120000 s)					
Blocking b_i (%)	4.95±1.45	6.23±1.75	16.43±4.46	2.53±0.88	6.37±1.83

4.6 Global Sample Variance Reduction

Services dealing with application events are normally in the order of minutes or even hours, while events related to the delay of packets are in the range of milliseconds. Therefore, it is possible to consider a global sample variance reduction. To obtain the CIs for the delays (for the same simulation interval), there is a much larger number of events, which ensures a narrower CI. Table 9 shows four simulations (A - D) with variations regarding simulation (IV) in the previous Table 8 (all services exponential, except Weibull for Video Clip service). Simulation (A) is performed at the application layer and the others (B,C,D) at the

packet layer. for Video Clip service). The simulations are carried out in an Intel (R) Core (TM), i7, CPU 1.7 GHz and 4 GB RAM. To obtain a significant reduction in total simulation time without a significant loss of accuracy, we make the blocking calculation with the simulation program based on applications as in Subsections 4.2 and 4.4. Afterwards, we perform a simulation with much less run time to evaluate packet delays as in 4.5. In terms of execution time, we may combine simulations A and B (Table 9) resulting in a total run time of 6149 (i.e. 588 + 5561 min); likewise, we may have A and C = (588 + 1426 = 2014 min), or A and D (588 + 253 = 841 min). To simulate with similar accuracy as A + B, we would need to perform a simulation with 10 X 320000 s, which corresponds approximately to 37073 min. Therefore, A + B represents a 16.59% reduction in the total processing time; A + C would cause a 5.43% reduction of the total processing time, and A + D a 2.27% reduction. Fig. 6 shows the results obtained. Thus, to improve accuracy without incurring in extra run time, the best option would be to have a simulation with more replications to evaluate the blocking and then perform another simulation to evaluate the smallest delay. If the desired accuracy for the delay is not significant, it would be sufficient to simulate 32000 s (253 minutes of run time), reducing the total time to about 98%.

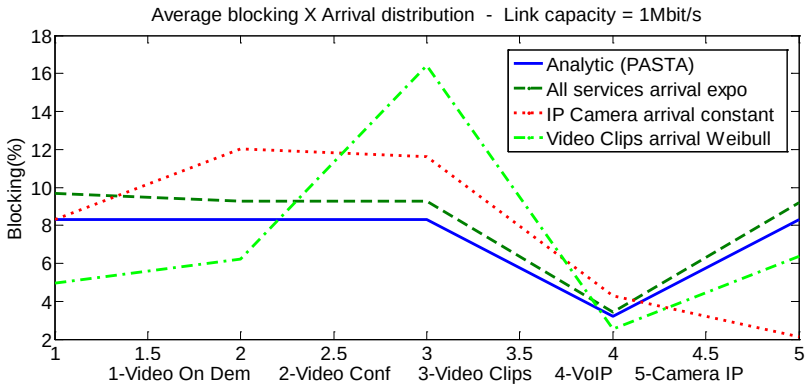

Figure 5: Mean simulation values.

5 CONCLUSIONS

As analytical models are very limited for the general characteristics of packet networks, in this work, we have proposed to increase the simulated models step-by-step. Simulation models were first validated in relation to the analytical models and then incremented with small but significant changes. For example, by just changing one type of service distribution (e. g., from exponential to constant, or from exponential to Weibull) the results changed significantly, and the analytical model could not capture these modifications. We argue that the results seem relatively proper for general use in simulation models.

The incremental simulation approach presented is sufficiently general to allow:

- The dimensioning of links with multi-service traffic (voice, data and video), i.e. the determination of the link capacity required for a given stream-traffic demand;
  A variation of the compression CODECs and the necessary re-dimensioning of the link capacity and packet delay for a given traffic demand, while still meeting QoS requirements (e.g. if we use the P2P protocol or VAD, Voice Activity Detection, as in VoIP ON-OFF service at layer 2, the only change would be simply the number of bits sent by each packet);
- Significant reduction of the total run time, by dividing the simulation model into sub-models that take into account different sample variances, each model running on a different time scale. This was only possible due to the incremental feature of the models;
- Evaluation of the impact on the fulfillment of QoS requirements (and link capacity/delay) in the case of technological upgrades or changes to network elements;

- Approximation of the delay to a Gaussian distribution in case we have the mean and standard deviation of the delay for more than two hops.

Table 9: Traffic simulation, run time and duration with C_s = 1000000 bit/s (constant).

Service	**Video On Dem.**	**Video Conf.**	**Video Clips**	**VoIP**	**IP Camera**
Traffic ρ_i (Erl)	5.0	2.5	3.3	30.0	5.0
(A) Application Layer Simulation - Video Clips Weibull (10 X 320000 s), run time = 588 min					
Blocking b_i (%)	5.98±1.07	6.88±0.75	17.70±1.52	2.55±0.18	6.56±0.43
(B) Packet Layer Simulation - Video Clips Weibull (4 X 120000 s), run time = 5561 min					
Blocking b_i (%)	4.95±1.45	6.23±1.75	16.43±4.46	2.53±0.88	6.37±1.83
(C) Packet Layer Simulation - Video Clips Weibull (1 X 96000 s), run time = 1426 min (*)					
Blocking b_i (%)	7.69	6.55	18.25	3.01	7.20
(D) Packet Layer Simulation - Video Clips Weibull (1 X 32000 s), run time = 253 min (*)					
Blocking b_i (%)	0.00	3.89	19.51	3.22	7.90
(*) CIs have not been calculated due to the small number of events					

Finally, the models considered in this work have not taken into account the behavior of the link with joint stream and elastic traffic. This is subject of future work.

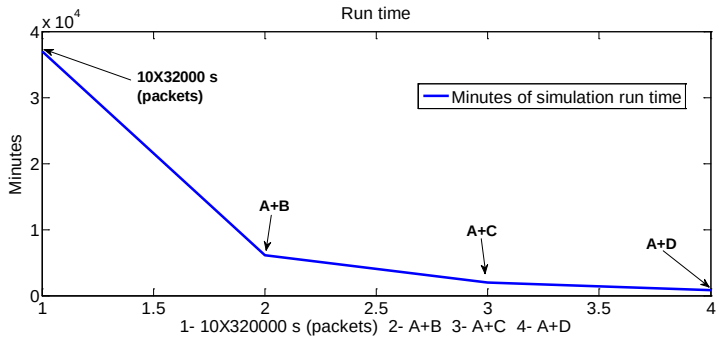

Figure 6: Reduction in simulation run time.

ACKNOWLEDGEMENTS

We would like to thank FAPESP/UNICAMP-FAEPEX 2011/17339-5 and CNPq #310980/2012-7 grants

REFERENCES

Alves, M. R., R. Matias Jr., and P. J. Freitas Filho. 2014. "Modeling and Simulation Applied to Capacity Planning of Voice Gateways: A Case Study." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, 3143–3154. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Cisco, 2006. "Understanding Delay in Packet Voice Network" http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/5125-delay-details.html [Accessed February 1st, 2015].

Diório, R. F., V. S. Timóteo, and E. L. Ursini. 2014. "Testing and IP-based Multimedia Gateway." INFOCOMP *Journal of Computer Science*, UFL, Lavras: MG, Brazil 13(1): 21-25.

Freitas Filho, P.J. 2014. *Introdução à Modelagem e Simulação de Sistemas com Aplicações em Arena*, Florianópolis, SC, Brazil: Visual Books.

Gonzalez, M. E. 2007. "A Generalized Packet Traffic Simulator for 4G Network Dimensioning Tools." *Vehicular Technology Conference* , VTC 2007-Spring. IEEE 65th.

Jain, R. 1991. *The Art of Computer Systems Performance Analysis*, Wiley.

Karam, M. J., and F. A. Tobagi. 2001. "Analysis of the Delay and Jitter of Voice Traffic Over the Internet." INFOCOM 2001.

Kavacky, M., E. Chromý , L. Krulikovská, and J. Pavlovič. 2009. "Quality of Service Issues for Multiservice IP Networks." SIGMAP 2009 – *Int. Conf.on Signal Proc. and Multimedia Applications*.

Kelton, W.D., R.P. Sadowski, and D.A. Sadowski. 2001. *Simulation with ARENA*, McGraw-Hill Higher Education.

Labourdette, J.F.P., and G.W. Hart. 1992. "Blocking Probabilities in Multitraffic Loss Systems: Insensitivity, Asymptotic Behavior and Approximations." *IEEE Transactions on Communications* 40 (8): 1355-1366.

Perényi, M.,T. D. Dang, A. G. Gefferth, and S. Molnár. 2006. "Identification and Analysis of Peer-to-Peer Traffic." *Journal of Communications* 1 (7): 36-46.

Rani, K. S. K., and J. Suganthi. 2014. "Improving Quality of Service in IP Networks for Multimedia Applications with Optimal Fragmentation." *Journal of Computer Science* 10 (8): 1336-1343.

Riedl, A., T. Bauschert, M. Perske, and A. Probst. 2000. "Investigation of the M/G/R Processor Sharing Model for Dimensioning of IP Access Networks with Elastic Traffic." *Institute of Communication Networks*, Munich University of Technology (TUM): 1-10.

Salvador, P., A. Pacheco, and R. Valadas. 2004. "Modeling IP Traffic: Joint Characterization of Packet Arrivals and Packet Sizes Using BMAPs." *Computer Networks* 44: 335-352.

Sharma, R.L.. 1997. *Network Design Using EcoNets*, Boston: International Thomson Computer Press.

Trindade, M.B., M.S. Medrano, and A.C. Lavelha. 2003. "Planejamento de Enlaces IP Multi-serviço considerando Requisitos de QoS." *Congresso Nac. de Tecn. da Inf. e Com.* – SUCESU 2003.

Ursini, E. L., Paulo S. Martins, Varese S. Timóteo, and V. F. Santos. 2014. "An IP VPN Network Design and Dimensioning Approach Using Analytical-Simulation Models with Incremental Validation." In *Proceedings of the 2014 Winter Simulation Conference*, edited by A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller. 4095–4096. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Walkowiak, K.. 2010. "Dimensioning of Overlay Networks for P2P Multicasting." *IEEE/IFIP Network Operations and Management Symposium* - NOMS 2010: Short Papers.

AUTHOR BIOGRAPHIES

EDSON L. URSINI is an Assistant Professor at Unicamp. He received a doctoral degree in Electrical Engineering from Unicamp, Brazil. His research deals with data networks, stochastic processes, simulation and performance analysis. His email is ursini@ft.unicamp.br.

PAULO S. MARTINS is an Associate Professor. He received a PhD degree in Computer Science from The University of York, UK. He works with real time systems as well as communications network design and analysis. His email is paulo@ft.unicamp.br.

VARESE S. TIMÓTEO is an Associate Professor at Unicamp. He received a doctoral degree in Physics from the Universidade de São Paulo. His work addresses simulation and performance evaluation of networks and mobile wireless systems. His email is varese@ft.unicamp.br.

FLÁVIO R. MASSARO JÚNIOR is a doctoral candidate at the School of Technology, Unicamp. He works with scheduling and simulation applied to real time systems, including communication networks. His email is frmassaro @gmail.com.