TRAFFIC MODELING AND SIMULATION FOR NGN WITH MARKOV REWARD MODEL AND NEURAL NETWORKS

Svetla Radeva and Dimitar Radev

Abstract: The paper is devoted on traffic modeling and simulation. Under consideration is traffic with compression at media gateway nodes for Next Generation Networks (NGN). The NGN uses broadband transport technologies that enable QoS management, in which service-related functions are independent from underlying transport-related technologies. The transport layer provides connection between outer NGN elements, and elements located at the NGN servers, like databases and media gateways (MGW), which present interfaces between the NGN and other networks. Modelling and simulation of traffic with compression at media gateways is developed with Markov reward model using learning vector quantification. The NGN architecture is conceived to achieve independence of applications and services from basic switching and transport technologies. The bandwidth sharing policy with partial overlapped transmission link is considered. Calls arriving to the link that belong to VBR and ABR traffic classes, are presented as independent Poisson processes and Markov processes with constant intensity or random input stream, and exponential service delay time. Service delay time is defined according to MRM. Traffic compression is calculated using neural clustering and self-organizing maps. Numerical examples and simulation results are provided.

Keywords: Modeling and Simulation, Communication Networks, Integrated Services, Markov Reward Model, Neural Networks.

ITHEA Keywords: C.2 Computer-Communication Networks; I.6 Simulation And Modeling

Introduction

Next generation networks (NGN) are packet-based networks that provide various telecommunication services. The NGN offers unrestricted access to different service providers and supports generalized mobility that allows consistent and ubiquitous provision of services to users [Ivanova et al, 2011]. On the base of broadband transport technologies service-related functions are independent from underlying transport-related technologies. The NGN can be defined with the following fundamental characteristics:
Packet-based transfer;

Service provision that is independent from the network;

Open interfaces;

Control functions that are separated from bearer capabilities, call/session, and application/service;

Broadband capabilities that provide end-to-end QoS, and transparency;

Generalized mobility support;

Unrestricted access to different service providers.

The NGN architecture is layered including transport layer and service layer, where the boundaries are strictly defined. The transport layer provides connection between outer NGN elements, like user terminals, and elements located at the NGN servers, for example, databases and media gateways [Ivanova et al, 2015]. Access depends on applied technology. For example, wireless access can be provided through WiFi, WiMAX, and CDMA. Fixed access can be provided through DSL and wired LAN [Cochennec, 2002].

The service layer provides session and other services and delivery methods. Media gateway (MGW) nodes present interfaces between the NGN and other networks. This paper is devoted to modeling and simulation of traffic with compression at media gateways, developed with Markov reward model using neural networks.

Interfaces between the NGN and other networks

The NGN concept is based on integration of currently divided voice and data networks into a simpler and more flexible IP-based network, where the transport, control and service layers are independent and interact via open interfaces. The NGNs contain both wired and wireless access networks, where all IP networks allow different access options. Important NGN requirements are support for quality of services (QoS) and simplicity to provide new services through different networks.

The most popular access to the NGN is based on media gateways with changing transfer and switching. The media gateway nodes can be implemented as independent devices, or can be integrated in another system. In traditional circuit-switched networks, the intelligence is observed in the core of the network (e.g., in central switches). In the NGN model, the intelligence for transfer and switching is expected to be decentralized at the edge of the network.

The NGN architecture has to achieve independence of applications and services from basic switching and transport technologies. The fundamental feature of the NGN architecture is independence of
applications and control mechanisms from the access and transport layers. That is possible to be achieved with migration of applications and call functions to open platforms, and introduction of common control protocols for communication between control functions and network resources. Particularly, that can be reached with the gateways providing conversion between different communication media and protocol adaptations.

The NGNs with open architecture consist of three main layers:

- Connectivity layer;
- Control layer;
- Application and service layer.

The connectivity layer consists of the following elements:

- Multi-Service Core: the IP-based transport backbone that carries multiple services over high-speed optical links. This part of the network acts as a long haul transport system providing connectivity among geographically distributed nodes. This network is shared by different services (e.g., phone calls, Web sessions, video-conferences, multi-player games, movies).

- Access Segment: consists of various different broadband access technologies (xDSL, broadband wireless, optical technologies etc.).

- Gateway Elements: they convert the information between different standards and representations.

The control layer is independent from the transport (physical) layer, which provides open and programmable interfaces towards the independent application layer that seamlessly mediates between the signaling protocols of different interconnected networks. The access layer includes both wired and wireless network technologies. The core transport network is built around Dense Wave Division Multiplexing (DWDM) transport system.

Important elements of the NGN architecture are media gateways and soft-switches. Gateways will be employed to interconnect networks based on different representation of the same signal. Common elements of the control layer are multi-service soft-switches, which are able to operate regardless of different protocols. Soft-switch is designed as software application that runs on the server or switch that directs the media gateway switching activities [Fazekas et al, 2002].
The media gateway nodes are located at the ends of the NGN network, and consist of the following elements:

- Interface with the networks with circuit switching (e.g., TDM network);
- Digital Signal Processor (DSP) for signal processing between circuit-switched networks and packet networks;
- Interface with the packet networks (e.g., LAN).

In dependence of their size, there exist three categories of the media gateway nodes:

- Small Office/Home Office (SOHO) for small peripheral networks, including voice, VoIP, data and video devices;
- Office, for medium size peripheral networks;
- Provider or carrier grade with high capacity in terms of simultaneous sessions and aggregate bandwidth.

For considered interfaces, the bandwidth sharing models with partial overlapped transmission link for media gateways are developed.

**Markov reward model for bandwidth sharing of media gateways**

Partial overlap of the bandwidth sharing model is defined in the following way: traffic of service $i$ obtains part of bandwidth equal to $r_i m_i$ bandwidth units, and the rest of traffic classes concur for sharing the rest of the link capacity $C - n_1 m_1 - n_2 m_2$ bandwidth units, where $C$ is the whole capacity. Input traffic is described as traffic of service $i$, if it has $m_i$ existing items in reserved capacity of $r_i m_i$ bandwidth units, or in sharing capacity of $C - n_1 m_1 - n_2 m_2$ bandwidth units; otherwise connection is blocking and lost [Balsamo et al, 2001]. In this way, schemes for access in full sharing and full separating of traffic flows can be introduced as particular cases of the partial overlap scheme, where for $r_1 = r_2 = 0$ full sharing is obtained, and for $r_1 m_1 + r_2 m_2 = C$ full separating is obtained. The obtained function for retranslation with partial overlapped transmission link for $(n_1, n_2) \in \Omega$ can be presented according to (1).

\[
\alpha_1(n_1, n_2) = \begin{cases} 
0, & \text{if } m_1 > C - n_1 m_1 - \max\{n_2, r_2\} m_2 \\
1, & \text{if } m_1 \leq C - n_1 m_1 - \max\{n_2, r_2\} m_2
\end{cases}
\]

\[
\alpha_2(n_1, n_2) = \begin{cases} 
0, & \text{if } m_2 > C - \max\{n_1, r_1\} m_1 - n_2 m_2 \\
1, & \text{if } m_2 \leq C - \max\{n_1, r_1\} m_1 - n_2 m_2
\end{cases}
\]  

(1)
The partial overlapped link (POL) is specific case of the partial overlap scheme. The basic idea consists in the fact that there is a traffic service class, for which part of bandwidth is reserved, e.g. \( r_1 > 0 \) and \( r_2 = 0 \). At the same time, this class is not concurring for the rest of free capacity, e.g. \( m_1 = 0 \), \( m_2 > 0 \). In this way, a hybrid approach is developed for full separating scheme, such as CBR service class, and full sharing scheme, such as variable bit rate (VBR) service class. This approach helps to determine optimal capacity of broadband connection channels shared by two or more traffic classes and services, such as, for example, CBR traffic class for voice transfer, VBR traffic class for compressed video data, and ftp transfer.

For modeling a transmission link with integrated services is used Continuous Times Markov Chain, where single channel for bandwidth sharing policy is defined as Markov Reward Model [Rácz et al, 2003]. Models of traffic classes with guarantee of bandwidth, such as conversational, adaptive and elastic traffic classes, are presented as stationary stochastic processes developed with the network of three parallel queues in the states space, as shown in Figure 1.

![Figure 1. Bandwidth Sharing in Wireless System](image)

If different traffic classes, such as conversational CBR traffic classes, adaptive VBR traffic classes, and elastic traffic classes with available bit rate (ABR) are observed at the same time, then there is no chance to obtain equivalent sharing of bandwidth capacity. Even if an assumption is made that it is
possible to separate certain bandwidth capacity for VBR class, still there is no chance to make it possible for ABR class, because of the following reasons:

- ABR class doesn’t provide the same quality of service as VBR or CBR classes works;
- Bandwidth offered for ABR calls has high variation in dependence of link overload.

When considering the bandwidth sharing policy of the link capacity, the approach of full sharing should be avoided, and an alternative method should be developed. The input parameters of the model consist of a set of arrival rates ($\lambda_1, \lambda_2, \lambda_3$) and departure rates ($\mu_1, \mu_2, \mu_3$), the bandwidths ($b_1, b_2, b_3$), and the throughput constraints ($\tilde{\theta}_{\text{min}}, \tilde{\theta}_{\text{max}}$). The state model is uniquely described by triple ($n_1, n_2, n_3$), where $n_1$ is the number of states in constant (conversational) flows, $n_2$ is the number of states in adaptive flows, and $n_3$ is the number of states in elastic flows. In order to obtain the performance measurement, the CTMC’s generator matrix $Q$ and the bandwidth sharing policy are defined, so that the link capacity $C$ is divided into two parts: a common part and a part reserved for the only two traffic classes. As is shown in Figure 1, there are two cases in the POL bandwidth sharing policy: CASE A and CASE B.

CASE A demonstrates the partial overlapped transmission link between constant and adaptive traffic flows. The only case of the partial overlapped link between CBR traffic class and VBR traffic class is considered in the paper. The number of traffic classes considered in this model of sharing policy can vary and when the number of traffic classes is more than one, it increase the system complexity, and the space of the possible states increase as well.

Let us assume that the system consists of calls of two traffic classes, which arrive in the system as independent Poisson processes with exponential service times. The service times are defined according to Markov reward model [Rácz et al, 2003]. The following assumptions are made:

- VBR calls always use maximum possible bandwidth, which is a value that is less than or equal to bandwidth of $b_2$. At the same time, this value is equal to free capacity for CBR classes;
- All VBR traffic flows share the bandwidth in equal parts, e.g. the newly arrived calls and in-progress calls are compressed to the same values, if they haven’t been assigned their peak bandwidth. If during the newly arrived call the bandwidth is less than $b_2^{\text{min}}$, then the last call is not admitted to the system, next it is blocked and lost;
- VBR call management is ideal, e.g. the time for adapting the system to new widths of the bandwidth after the newly arrived calls is infinitesimal.

The actual residency time for VBR calls depends not only on quantity of transferred data, but also on obtained bandwidth for VBR. The following parameters are used to define this balance: the moment throughput of VBR call at the moment $t$ is determined as discrete random variable
\( \theta(t) = \min[b_2, (C-n_t b_1)/n_2] \) and the throughput of VBR call during the retranslation of \( x \) quantity of data for continuous random variables is determined as \( \theta = x/T_x \).

The following additional parameters are introduced for the proposed model of partial overlapped transmission link:

- \( B_1^{\text{max}} \) – maximal allowed blocking probability of CBR class;
- \( b_2^{\text{min}} \) – minimal allowed bandwidth for VBR class;
- \( \theta^{\text{min}} \) - minimal allowed throughput for VBR class;
- \( \varepsilon \) – threshold, which determines the value of \( \theta^{\text{min}} \).

Let us develop a CTMC, which state is determined as \( i = (n_1, n_2) \), where \( n_1 \) is the number of broadband CBR calls, and \( n_2 \) is the number of VBR calls. The partial overlapped link capacity is divided into two parts: a common part capacity of \( C_{\text{COM}} \), and a separated part capacity of \( C_{\text{VBR}} \) only for VBR calls.

The constraints that define the number of calls are presented as

\[
\begin{align*}
N_{\text{VBR}} &\leq C_{\text{COM}}, \\
N_{\text{VBR}} b_2^{\text{min}} &\leq C_{\text{VBR}}, \\
n_2 &\leq N_{\text{VBR}},
\end{align*}
\]

where \( N_{\text{VBR}} \) is the maximal number of VBR calls. These constraints are guaranteed for VBR calls in difference with CBR calls, where the maximal number of \( N_{\text{VBR}} \) calls is limited, and in this way, the new arriving VBR calls get protected. If there are too many of the new arriving VBR calls in the transmission link, the throughput \( \theta \) is decreased to \( \theta^{\text{min}} \), and must be regulated via \( N_{\text{VBR}} \). The generator matrix \( Q \) is built in such a way that only transitions between the neighboring states are allowed, where \( q_{ij} \) is depicted transition from state \( i \) to state \( j \). There are four possible transitions between the states, therefore two equations describe the newly arrived calls, and other two describe their services. The compression between traffic classes can be obtained according to (2),

\[
r_i = \min(n_2, \frac{C - b_1 n_1}{b_2})
\]  

(2)

where \( r_i, b_2 \) is a common width of the bandwidth for VBR calls, when the system is at the state \( i \). The partial overlapped transmission link model is completely defined with the following two input parameters: \( N_{\text{VBR}} \) and \( C_{\text{COM}} \). The purpose is to minimize the blocking probability of CBR calls and to determine minimal throughput of VBR calls, under criteria for required quality of service.

CASE B demonstrates the partial overlap between adaptive and elastic traffic flows. This model of sharing policy is used in mobile communications as soon as modeling the media with integrated traffic
flows with variable bit rate of the real-time working sources is required. There are two basic reasons to apply the proposed model in networks with integrated services. The first reason is related to quality of elastic traffic flows with constant width of the bandwidth, as the bandwidth that is busy with elastic flow depends on current load of the transmission link, and on management and control algorithms in the network nodes. The second reason is related to the blocking of elastic flows, where the service is complete even if available bandwidth is very limited during newly arrived calls. For many services the actual residency time of elastic flows depends on the throughput that flow obtains. For instance, a file transfer protocol (ftp) session would last longer if its throughput decreased. The traffic service class that is standard for GSM mobile communications doesn’t change the bandwidth sharing policy model, because part of the link capacity $\delta$, which is obtained for this class of provider services is extracted from the link capacity dedicated to three basic service classes. In this way, the schema of full separating is used for the service class, e.g. the link capacity is left equal to $C-\delta$. The following possibilities can be considered for the proposed schema of the partial overlapped transmission link:

- If there is enough bandwidth, then all traffic flows will obtain their necessary peak bandwidth, and second and third traffic classes will obtain, respectively, $b_2$ and $b_3$ bandwidth units.

- If providing compression of the bandwidth is required, e.g. if $n_1 b_1 + n_2 b_2 + n_3 b_3 > C-\delta$, then the bandwidth compression is organized in such a manner that the bandwidth is equally shared between adaptive and elastic traffic classes up to the moment when a constraint for minimal possible value of the one of two classes is achieved.

- If providing additional compression of the bandwidth is required for the newly arriving calls, then the class that can allow additional decrease of the width of the bandwidth decreases it up to the moment when minimal constraint for that class is achieved. After that, the newly arrived calls are rejected.

The described rules demonstrate that not only adaptive but elastic flows always obtain their maximum possible bandwidth as well, which is less than requirements to the peak bandwidth for $b_2$ and $b_3$. Equal part of the bandwidth left from adaptive and elastic traffic flows for the constant traffic flows.

The compression of ABR calls is required to achieve better throughput. The particular feature of ABR classes is related with their bit rate that changes over time, e.g. vary in certain interval that makes them appropriate for compression. If the compression is applied with higher than maximum possible value, then it leads to the total loss of information. The case with VBR and ABR traffic service classes is used, with the calls arriving as independent Poisson processes and introduced to the model as Markov processes designed for pure birth with constant intensity or random input stream. The time for service delay is exponential. The time for service delay can be determined using MRM. All adaptive and elastic
traffic flows share proportionally the available bandwidth among themselves, i.e. the newly arrived flow and in-progress flows will be squeezed to the same compression values. After that, if a newly arriving flow decreases the flow bandwidth below minimal accepted value, and is not admitted to the system, then it is blocked and lost. The compression of traffic classes can be presented via a common width of the bandwidth $r_i$, for ABR calls, as shown in (3).

$$r_i = \min\left(n_i, \frac{C-b_i}{b_i}\right), \quad i = 2,3$$

(3)

The proposed model is based on MRM, in which only transitions between the neighboring states are allowed, and possible state transitions are described with nonzero transition rates according to (4),

$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i+1,n_{i2},n_{i3}) = \lambda_i$$
$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i,n_{i2}+1,n_{i3}) = \lambda_2$$
$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i,n_{i2},n_{i3}+1) = \lambda_3$$
$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i-1,n_{i2},n_{i3}) = n_i \cdot \mu_i$$
$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i,n_{i2}-1,n_{i3}) = n_2 \cdot \mu_2$$
$$q(n_{i1},n_{i2},n_{i3} \rightarrow n_i,n_{i2},n_{i3}-1) = n_3 \cdot \rho_3(n_{i1},n_{i2},n_{i3}) \cdot \mu_3$$

(4)

where the first three equations represent the state transitions due to call arrivals, while the second three equations represent the transitions due to call departures. The $n_3 \rho_3(n_{i1},n_{i2},n_{i3}) \cdot \mu_3$ quantity denotes the total bandwidth of interactive flows when the system is in state $(n_{i1},n_{i2},n_{i3})$, and their compression is denoted as $\rho_3$.

**Calculation of traffic compression with neural networks**

The calculations of the service coefficients are made through multiplying the number of flows with the coefficients of compression for the corresponding states. For the link capacity of $C=50$ units 35 feasible states are obtained, and for 16 of them the traffic flow is compressed below the peak bandwidth for adaptive and elastic traffic classes, based on the proposed rule for minimal throughput. The clustering procedure is used with steady-state distribution calculated in stochastic node network with Discrete
Time Markov Chain (DTMC) [Balsamo et al, 2001] that provides capabilities to resolve the tasks described below, in the following conditions [Gross and Harris, 1998]:

- The topology and structure of Markov chain are known in advance, e.g. the number and shape of clusters classes are determined. In that case the task of adjusting the target clusters classes size cannot be resolved with conventional stochastic methods. Neural structures with unsupervised learning and clustering algorithms such as Kohonen networks, K-means clustering and Gaussian Mixture models offer models with greatly reduced training time. These models, known collectively as Vector Quantifications (VQs), provide capability to present the winning node that represents the same class as a new training pattern [Buhmann, 2002; Webb, 1999]. With two-layered learning neural structure one can successfully estimate probability density function, occupancy distribution, rare event probability of DTMCs and MRMs.

- The topology of Markov chain is unknown, e.g. the number of clusters classes only is determined in advance. The performance of Markov chain mapping should correspond to possible classes of event that generates the data.

Minimum-squared-error algorithm is used to solve the clustering problem. Let’s assume that a dataset \( x=(x_1,\ldots,x_n) \) of points is given in some Banach space, which partitions the data into \( k \) clusters (e.g., disjoint groups), so that some minimizing empirical loss function can be written according to (5),

\[
D(x) = \frac{1}{n} \sum_{j=1}^{k} \sum_{i}^{n_j} \|x_{ij} - s_j\|^2
\]  

where \( x_{ij} \in C_j, x_{ij} = x_i I_{[x_i \in C_j]} \), \( \sum_{j=1}^{k} n_j = n \) and the dataset points belong to a \( d \)-dimensional Euclidean region \((d \geq 2)\), \( C_j \) denotes the \( j \)-th cluster, \( n_j \) denotes the number of point \( x_i \) in \( C_j \). The centroid (with the same expected value) has been partitioned into \( d \) clusters with \( n_j \) elements; and the mean vectors \( s_j \) are given as (6).

\[
s_j = \frac{1}{n_j} \sum_{x_i \in C_j} x_{ij}
\]
On the other hand, as of (5) and (6), the patterns are moved from one cluster to another only if such move improves the criterion function $D(x)$, given as (7)

$$D(x) = \sum_{j=1}^{k} \sum_{i} y_{ij} \left\| x_{ij} - S_j \right\|^2$$

(7)

where $y_{ij}$ is the indicator of $\{x_i \in C_j \}$ - $y_{ij} = I_{\{x_i \in C_j \}}$ and $n_j = \sum_{j=1}^{n} y_{ij}$.

The mean vectors and the criterion function are updated after each pattern move. Like hill-climbing algorithms in general, similar approaches can guarantee local (but yet not global) optimization. Different initial partitions and sequences of the training patterns can lead to different solutions. The goal of clustering is to partition the sample set of points into $k$ (not necessarily equal) clusters $C_j$, such that (5) is minimized. In other words, it is necessary to specify the set of centroids $S=\{s_1, \ldots, s_k\}$ and the corresponding partitions $\{ C_j \}$, which minimize (7). The definition also combines both the encoding and decoding steps in vector quantification. Clustering techniques with the loss function are called minimum-variance methods. Most well-known clustering and vector quantification methods update the set of centroids $S$, starting from some initial set $S_0$ and using iterative, typically gradient-based procedures that are multi-extreme and depend on the initial value $S_0$ in the gradient-based procedures, then they converge to a local minimum rather than global minimum [Kohonen, 1997]. We can associate with the clustering used with an $n$ dimensional discrete distribution $f(x;p)$ with independent marginal $f_m(x_m;p_m)$, $p_m = (p_{m1}, \ldots, p_{mk})$, $m=1, \ldots, n$, and so that each $f_m(x_m;p_m)$ represents a discrete $k$-parameter of Probability Density Function (PDF) with masses at points $x_m=0,1,\ldots,k-1$. Note that for $k=2$, $f_m(x_m;p_m)$ is reduced to Bernoulli PDF. It is crucial to recognize that each generation based on $f(x;p)$ partitions the set samples into $k$ clusters $C_j$. The clustering procedure consists of advanced calculations of discriminating hyperplanes $W_j$ for the pair-wise discrimination of $k$ classes $k^* = k(k-1)/2$, and later, the prediction of the dataset sample $x$, e.g. the classification procedure. The linear regression is applied that for to discriminate the $k^*$-dimensional vector $V_{n,j}$, formed for each class. For example, we can assume that $W_j(x)$ is a regression function, as given in (8),

$$W_j = w_0 + w_1 x_1 + \ldots + w_n x_n$$

(8)
then, we can determine if the pattern \( x \) belongs to pair-wise classes \( C_1 \) and \( C_2 \). In that case, the \( j \)-th component of the vector \( V_{n,j} \) is equal to 1 \( (x \in C_1) \) only if \( W_j(x) > 0 \); and it is equal to -1 \( (x \in C_2) \) only if \( W_j(x) < 0 \); and, finally, it is equal to 0 in all other cases. Using discriminating functions, vector function \( sw \) can be defined for each pattern \( x \), as it is shown in (9).

\[
sw : X_j \rightarrow \{1,0,-1\}^{k^*} \\
sw(x)_j = \text{sign}(W_j(x))
\]  

(9)

For each class \( C_j \), function \( s_j(x) \) can be presented as \( s_j(x) = \sum_{j=1}^{k^*} V_{n,j} \cdot sw(x)_i \). The pattern \( x \) is uniquely classified with discriminating hyper-planes \( W_j \) \( (j = 1, \ldots, k^*) \) into class \( C_j \) only if \( s_j(x) = k-1 \), i.e. with respect to \( k-1 \) hyper-planes, which discriminate the class \( C_j \) from the other \( k-1 \) classes. Then, the pattern \( x \) is placed in its half-space that belong to class \( C_j \) \( (V_{n,j} \text{ and } W_j(x) \text{ have the same sign for all } V_{n,j} \neq 0) \). If the pattern \( x \) is not uniquely classified, then the Euclidian distances of \( x \) to all these hyper-planes \( W_j \) are calculated, and \( x \) is assigned to the class with minimum distance. Kohonen’s network algorithm provides a tessellation of the input space into patches with corresponding code vectors [Kohonen, 1997]. It has an additional feature that the centers are arranged in a low-dimensional structure (usually a string, or a square grid), such that nearby points in the topological structure (the string or grid) map to nearby points in the attribute space. The Kohonen learning rule is used when the winning node represents the same class as a new training pattern, while a difference in class between the winning node and a training pattern causes the node to move away from training pattern by the same distance. In training, the winning node of the network which is nearest node in the input space to a given training pattern, moves towards that training pattern, while dragging with its neighboring nodes in the network topology. This leads to a smooth distribution of the network topology in a non-linear subspace of the training data. The traffic compression example represents the link capacity of \( C = 50 \) units and 35 feasible states, where only for 16 of them the traffic flow is compressed below the peak bandwidth for adaptive and elastic traffic classes based on the proposed rule for minimum throughput; and the traffic compression is calculated with learning vector quantification. The distribution of arrivals is considered, where 700 independent and identically distributed stochastic values of traffic flows are generated on both inputs of the competitive layer of the neural network. These stochastic values simulate the behavior of the products \( N_1.b_1 \) and \( N_2.b_2 \), and they are generated with Gamma and
lognormal distribution functions, respectively. Preliminary vector quantification is developed when independent arrivals are evenly distributed as 10 hits to 70 target classes (Figure 2).

The high concentration of arrivals in inner classes can explain the fact that there is no blocking obtained for the given capacities of $C_1=60$ and $C_2=50$ units. The main purpose of analysis is to achieve optimal throughput and reduce the blocking probability with compression for arrivals, which are out of the link capacity and, evidently, not admitted to the system. The developed compression technique with $(r_1, r_2)$ creates the new grouping of arrivals following rejecting the classes that don’t belong to compression.

According to received simulation results losses in arrivals after traffic compression was in classes with numbers 10, 20, 30, 40, 50, 60, 69 and 70, as soon as their probability mass function (PMF) has low values. For the link capacity of $C_1=60$ units the blocking probability after compression is about 0.4%, which can be used for the high priority data transfers; while for the link capacity of $C_2=50$ units the blocking probability after compression is about 3.6%, which is a significant increase, and this link should be used for the low priority data transfers (e.g., e-mails, or ftp files).

![Figure 2. Vector Quantification with traffic compression](image-url)
Conclusion

The paper is devoted to modeling and simulation of traffic with integrated services at media gateway nodes for next generation networks based on Markov reward models. The bandwidth sharing policy with partial overlapped transmission link is considered. Calls arriving to the link that belong to VBR and ABR traffic classes, are presented as independent Poisson processes and Markov processes with constant intensity or random input stream, and exponential service delay time. Service delay time was defined according to Markov reward model. The traffic compression was calculated using neural network with learning vector quantification. Kohonen’s network algorithm was implemented for tessellation of the input space into patches with corresponding code vectors. Numerical examples and simulation results are shown for communication networks of various sizes.

Bibliography


Authors' Information

**Svetla Radeva, DSc** – Professor at the Information Technologies Department, University of Telecommunications and Post, 1st. Acad. Stefan Mladenov Str. Sofia-1700, Bulgaria; e-mail: svetla_ktp@abv.bg

*Major Fields of Scientific Research: Neural networks, Simulation and modeling of stochastic processes, and Queuing systems.*

**Dimitar Radev, DSc, Prof.–** Rector of the University of Telecommunications and Post, 1st. Acad. Stefan Mladenov Str. Sofia-1700, Bulgaria; e-mail: dradev@abv.bg