Simple Traffic Modeling Framework for Machine Type Communication

Navid Nikaein[*], Markus Laner[†], Kaijie Zhou[*], Philipp Svoboda[†], Dejan Drajic[‡], Milica Popovic[§], Srdjan Krco[‡]

[*]EURECOM, France, Email: firstname.name@eurecom.fr

[†]Vienna University of Technology, Austria, Email:firstname.name@nt.tuwien.ac.at

[‡]Ericsson d.o.o, Serbia, Email: firstname.name@ericsson.com

[§]Telekom a.d, Serbia, Email: milicapop@telekom.rs

Abstract—Understanding the machine-type communication (MTC) traffic is a key for designing and optimizing a network and the applicable QoS scheme capable of providing adequate communication services without compromising the conventional services such as data, voice, and video. In this paper, we propose a framework for traffic modeling, generation, and load estimation for machine-type communication that can be used to optimize the network to the intrinsic MTC traffic characteristic. For this purpose, key MTC traffic parameters are also derived and examined through a realtime sensor-based event detection application scenario highlighting the traffic pattern and volume generated by such an application.

Index Terms—M2M, MTC, Markov Chain, traffic modeling, traffic volume.

I. Introduction

Different from the traditional human-type communications (HTC) for which 3G and 4G wireless networks are currently designed and optimized for, machine-type communication MTC (or Machine to Machine (M2M) communications) is seen as a form of data communication, among devices and/or from devices to a set of servers, that do not necessarily require human interaction [1]. Such MTC is also about collecting and distributing the meaningful data efficiently, often in real time, managing connected devices, providing back-end connectivity anywhere and anytime and consequently enabling creation of the so-called Internet of-Things (IoT). At the present time, the most interesting applications from the commercial point of view are related to intelligent transport, smart meters (automatic electricity, water and gas meters reading) and tracking and tracing in general. However, the M2M application space is vast and includes security, health monitoring, remote management and control, distributed/mobile computing, gaming, industrial wireless automation, and ambient assisted living.

The MTC promises huge market growth with expected 50 billion connected devices by 2020 [2]. Support for such a massive number of MTC devices has deep implications on the end-to-end network architecture. Lowering both the power consumption and the deployment cost are among the primary requirements. This calls for a paradigm shift from a high data rate network to a MTC-optimized low cost network to create new revenues. Although some of the MTC use-cases are better suited for wired or short-range radio, wireless communication systems are becoming more adequate for majority of the MTC applications as they are encompassing a wide range of requirements including mobility, ease of deployment and coverage extension.

In this perspective, one of the most challenging problem is the co-habitation of M2M traffic with conventional user traffic coupled with the potential of a rapid increase in the number of machines connected to cellular infrastructure. This is because such systems are primarily designed for a continuous flow of information, at least in terms of the time-scales needed to send several IP packets (often large for user-plane data), which in turn makes the signaling overhead manageable (relative to the user-plane amount of data). Analysis of emerging M2M application scenarios such as smart metering/monitoring, e-health, and e-vehicle as well as one real application denoted as Busplus has revealed that in the majority of cases, the MTC traffic includes all or a subset of the following specific features [1], [6], [9] :

- Short and small number of packets
- Low duty-cycle packets (i.e. long period between two data transmissions)
- Uplink-dominant packets (i.e. more packets in uplink than downlink)
- Real time and non-real time packets
- Periodic and event-driven packets
- Raw and aggregated packets (i.e. combining traffic of multiple sources into a single packet, relevant for specific nodes such as gateway)
- Unsynchronized and synchronized packets (i.e. simultaneous access attempts from many devices reacting to the same/similar events)

The above-mentioned features have been also confirmed through a recent traffic measurement analyses for non-realtime applications in an operational 2G/3G cellular network [7]. Another recent study in an operational 3G network with the Busplus application has shown that even a few tens of M2M terminals can lead to a significant network accessibility degradation (up to 60%), which seriously affects the conventional services such as data, voice, and video [9]. This was mainly due to the low data rate and sporadic nature of the traffic coupled with the sensitivity of the radio access scheduling algorithm to such a traffic pattern (composed of packet inter-arrival time and size) [8], leading terminals to the RRC idle mode creating congestion on the signaling channels

(FACH/RACH).

Thus, understanding the MTC traffic characteristics is a key for designing and optimizing a network and the applicable QoS scheme capable of providing adequate communication services without necessarily compromising the conventional services. In particular, the success of 3GPP Rel-11 Evolved Packet System (EPS) depends on the effectiveness of its class-based network-initiated QoS control scheme to support MTC and HTC traffics. This is because the operators are moving from a single to a multi-service offering while the number of connected devices and their traffic volume are rapidly increasing [10]. Such a QoS control allows different packet-forwarding treatment (i.e. scheduling policy, queue management policy, resource reservation, rate-shaping policy, link-layer configuration) for different traffic using EPS bearer mapping, which is a key enabler for supporting various MTC application scenarios.

The MTC is a very active area under discussions in the 3GPP for integration within the LTE/LTE-Advanced framework [3], the IEEE 802.16 [4], and more generally within ETSI. Both the 3GPP and the IEEE 802.16 provide traffic models for M2M in [3] and [4], by considering the average packet size and the arrival distribution over a given time period. The 3GPP model further distinguishes between unsynchronized and synchronized traffic. This work extends and generalizes the existing traffic models by introducing the concept of the **traffic state**, where different packet size and inter-arrival time distributions can be applied depending on the current traffic state.

This paper is organized as follows. Section II presents the proposed framework and its modeling approach through an example application scenario. Section IV provides a discussion and presents the process of MTC packet generation obtained by the model. Finally, Section V summarizes and concludes this work.

II. M2M Traffic Modeling Framework

Nowadays, mobile networks are dimensioned using standard mobile wireless network traffic models, which are based on the typical behavior of human subscribers. It may be expressed in typical time spent using speech service, number of sent/received messages (SMS, MMS) and the amount of downloaded data. These traffic models do not take into account traffic generated by machines, thus new traffic models are required.

Some examples of (future) MTC scenarios are listed below in order to highlight the diversity in data-traffic the network designers will have to deal with. For instance, in the case of meteorological alerts or monitoring of the stability of bridges, MTC devices will infrequently deliver a small amount of data. Another type of application is event detection requiring fast reaction time to prevent potential accidents; one example is the detection of pressure drop through the pipelines (gas/oil). Moreover, in the field of surveillance and security, the sensing devices send periodic reports to the control center until a critical event happens. Once the event is triggered, *event driven*

data traffic is first sent by the sensor to a central control unit or other types of infrastructure. Subsequently more packets may be exchanged between parties to handle this event.

Analyzing the functions of the majority of the applications has revealed that the MTC has three elementary traffic patterns:

- **Periodic Update (PU):** This type of traffic occurs if devices transmit status reports of updates to a central unit on a regular basis. It can be seen as an event triggered by the device at a regular interval. PU is non-realtime and has a regular time pattern and a constant data size. The transmitting interval might be reconfigured by the server. A typical example of the PU message is smart meter reading (e.g. gas, electricity, water).
- **Event-Driven (ED):** In case an event is triggered by an MTC device and the corresponding data has to be transmitted, its traffic pattern conforms to this second class. An event may either be caused by a measurement parameter passing a certain threshold or be generated by the server to send commands to the device and control it remotely. ED is mainly a *realtime* traffic with a variable time pattern and data size in both uplink and downlink direction. An example of the realtime ED messages in the uplink is an alarm / health emergency notification and in the downlink could be a Tsunami alert. In some cases, ED traffic is non-realtime, for example when a device sends a location update to the server or receives a configuration and firmware update from the server.
- **Payload Exchange (PE):** This last type of data-traffic is issued after an event, namely following one of the previous traffic types (PU or ED). It comprises all cases where larger amount of data is exchanged between the sensing devices and a server. This traffic is more likely to be uplink-dominant and can either be of constant size as in the telemetry, or of variable size like a transmission of an image, or even of data streaming triggered by an alarm. This traffic may be real time or non-real time, depending on the sensor and the type of the event.

Real world applications are often a combination of the above-mentioned traffic types, as described in Section III. Hence, using the three elementary classes above for traffic modeling enables building models with an arbitrary degree of complexity and accuracy. For example, a device may enter the power saving mode, trigger a PU and potentially multiple ED traffic patterns at regular intervals, thus making the traffic pattern a periodic event driven. Furthermore the PE may happen after the ED to provide further details about the events. It has to be mentioned that the PU and the ED can be regarded as the short control information type of traffic (very low data rate), while the PE as the bursty traffic.

For a convenient modeling of MTC traffic (by deploying the above described traffic categories), we propose an On-Off structure, as depicted in Fig. 1. Together with the three distinct traffic patterns mentioned above, this can be integrated in a Markov structure with four different states s: *OFF*, *PU*,
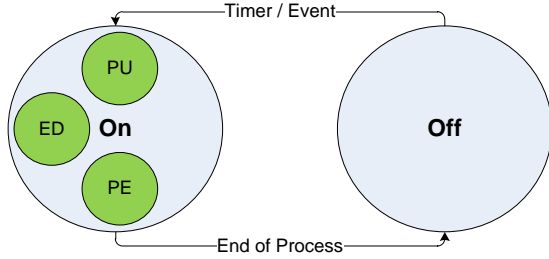
Fig. 1.   M2M Traffic Modeling Framework

ED and *PE*. The classification of the states into (several) On and (one) Off states facilitates the handling of the almost vanishing data rates, which is typical for MTC. The Off-state is thereby equivalent to an artificial traffic type, where no packets are transmitted neither from nor to the respective machine. This corresponds to situations such as the terminal being in idle/sleep mode. The predefined states shall resemble *real* functionality of MTC devices. This enables the assignment of meaningful side-information to each state, such as respective Quality of Service (QoS) parameters. For example, the attribute *latency* $< 100\,ms$ may be added to the state ED, in order to ensure fast forwarding of alarms.

For modeling the data streams within single states s we deploy renewal processes [14], pp. 254 ff. They consist of a random packet inter-departure times (IDT) D_s and a random packet sizes (PS) Y_s. Both random processes D_s and Y_s are identical and independent distributed (iid), with arbitrary marginal Probability Density Functions (PDFs) $f_{D,s}(t)$ and $f_{Y,s}(y)$. Two special cases are: periodic patterns, e.g., fixed inter-departure time, and Poisson processes, e.g., exponentially distributed IDT. Even though renewal processes are a too flexible description for the first three states (e.g., there are no packets generated in the OFF-state), we stick to this description for a coherent representation of all four states.

For interaction among the states, we define a Semi-Markov Model (SMM) [14], pp. 352 ff. Hence, we define transition probabilities $p_{s,\sigma}$ between states, with $p_{s,s} = 0$ transition probability to the current state. The transition probabilities are arranged in the transition probability matrix $\mathbf{P}$. Further, a random sojourn time or holding time T_s is introduced per state, with arbitrary independent distribution $f_{T,s}(t)$. Two special cases are: exponential, i.e., corresponding to an ordinary Markov Model, and constant, e.g., a fixed timer. Again, this description is too general for some states, e.g., the PU state is visited only for one short instant of fixed duration, but preferable for the analytic treatment. SMM models are advantageous for MTC modeling for several reasons: (i) they allow capturing a broad spectrum of traffic characteristics [11], especially the almost vanishing data-rate, (ii) enable augmented modeling if side-information is available (e.g., the exact number of states are known) [12] and (iii) advanced fitting mechanisms are established [13], which allow for good fitting quality, even if nothing but raw traffic-measurements

TABLE I
TRAFFIC MODEL INPUT PARAMETERS

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	$\mathbf{P}$			
OFF	$\mathcal{D}\mathrm{eg}(\infty)$	$\mathcal{D}\mathrm{eg}(0)$	·	0	·	·	·
PU	$\mathcal{D}\mathrm{eg}(\infty)$	·	$\mathcal{D}\mathrm{eg}(\Delta T)$	·	0	·	·
ED	$\mathcal{D}\mathrm{eg}(\infty)$	·	$\mathcal{D}\mathrm{eg}(\Delta T)$	·	·	0	·
PE	·	·	·	·	·	·	0

are given.

The input parameters for the model are summarized in Table I, where "·" represents parameters to be fitted to a desired MTC traffic pattern and the completed items are state specific constants. Thereby, $\mathcal{D}\mathrm{eg}(\cdot)$ represents the degenerate distribution, corresponding to a constant value, and ΔT represents the minimum temporal resolution of the model. Note, that the state specific constants conform two special cases, namely, (i) no traffic is generated within a state, e.g., Off-state and (ii) the sojourn time is very short and only one chunk of data is transmitted, e.g., PU and ED-state.

As already mentioned, the amount of generated traffic per machine (in terms of throughput) is vanishing small. However, future setups will involve up to hundreds or thousands of devices [4], hence, the overall data-rate R_{tot} will be of interest, in order to optimize applications and infrastructure. A simple method for estimating of R_{tot} for a number of N MTC-devices is outlined in the following. Therefore a set of parameters are required, which may be deterministic or random:

- N: Number of M2M devices/sensors
- s, σ: index of the state (e.g., OFF=1, PE, PU, ED=4)
- S: number of states, we assume $S = 4$
- $f_{D,s}$: distribution of the inter-departure time in state s
- $f_{Y,s}$: distribution of the packet size in state s
- $f_{T,s}$: distribution of the holding time in state s
- ρ_{OH}: ratio of the signaling overhead with respect to the data caused by the underlying protocols (e.g. TCP/UDP and IPv4/IPv6).
- $p_{s,\sigma}$: state transition probabilities (e.g., $p_{\mathrm{OFF,PU}}$)
- $\mathbf{P}$: state transition probability matrix
- $\overline{D}_s$: mean inter-departure time in state s
- $\overline{Y}_s$: mean packet size in state s
- $\overline{T}_s$: mean sojourn time in state s
- π_s^e: stationary state probabilities of the embedded Markov chain (i.e., the Markov model obtained by sampling the continuous SMM model at the state transition instances)
- $\boldsymbol{\pi}^e$: stationary state probability vector of the embedded Markov chain
- π_s: the stationary state probabilities of the SMM
- R_s: the mean data-rate in state s
- R_{tot}: global mean data-rate

Starting from the defined distributions $f_{D,s}$, $f_{Y,s}$ and $f_{T,s}$, the respective mean values $\overline{D}_s$, $\overline{Y}_s$ and $\overline{T}_s$ can easily be computed by integration. Further, the mean data rate for each state s is

calculated according to

$$R_s = \frac{\overline{Y}_s}{\overline{D}_s}. \tag{1}$$

From the designated matrix P the stationary state probabilities of the embedded Markov chain $\boldsymbol{\pi}^e$ can be calculate by solving the eigenvalue problem

$$\boldsymbol{\pi}^e = \boldsymbol{\pi}^e \mathbf{P}, \qquad \text{under} \quad \sum_{s=1}^{S} \pi_s^e = 1. \tag{2}$$

They are further used to calculate the actual state probabilities of the SMM [14], p. 353 by

$$\pi_s = \frac{\pi_s^e \overline{T}_s}{\sum_{\sigma=1}^{S} \pi_\sigma^e \overline{T}_\sigma}. \tag{3}$$

The total expected data rate can now be calculated by summing over all MTC devices n according to

$$R_{tot} = \rho_{\text{OH}} \sum_{n=1}^{N} \sum_{s=1}^{S} \pi_{s,n} R_{s,n}, \tag{4}$$

which becomes to a multiplication with N in case of all machines are equal.

Despite the complex representation of the traffic in terms of a SMM, the model fitting is relatively easy. This is due to many parameters of the model are fixed, cf. Table I. Further, traffic generated by the four different states is clearly distinguishable. Hence, a given input stream, the SMM should be fitted to, can a priori be split into multiple snippets, each of which assigned to one of the four states. For example, (i) *Off* phases are obviously easy to detect, (ii) *PU* phases are detectable for their periodicity. Since $p_{s,s}$ equals zero, each snippet represents a single sojourn at the given state. Consequently, the parameters $f_{D,s}$, $f_{Y,s}$, and $f_{T,s}$ can be estimated from the set of snippets associated to the state s. Further, $p_{s,\sigma}$ can be estimated from the sequence of snippets by assessing the number of transitions form s to σ. The mentioned parameters fully define the model.

In the following section, we apply the proposed modeling framework to a realtime sensor-based alarm and event detection applications and extract the elementary MTC traffic patterns and estimate the total traffic volume.

III. MODELING SENSOR-BASED ALARM AND EVENT DETECTION APPLICATIONS

Many categories of sensor-based applications exist or will be reasonably implemented in the future. In some applications, sensors infrequently deliver a small amount of data: e.g. high risk transportation, meteorological alerts, and stability of buildings. Another type of application is event detection requiring fast reaction time. An example is the detection of pressure drop through the pipelines (gas/oil); this critical information should be sent immediately to the control center in order to prevent potential accidents. In the field of surveillance and security, discrete sensors which should stay undetected enable interesting applications too. Examples of this type of applications can be intrusion detection sensors, or automated

TABLE II
TRAFFIC PARAMETERS FOR UPLINK SENSOR BASED ALARM SCENARIO, VALUES IN SECS. AND BYTES

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	**P**			
OFF	$\mathcal{D}\text{eg}(\infty)$	$\mathcal{D}\text{eg}(0)$	$\mathcal{D}\text{eg}(30\,\text{min})$	0	0.5	1	1
PU	$\mathcal{D}\text{eg}(\infty)$	$\mathcal{D}\text{eg}(1000)$	$\mathcal{D}\text{eg}(\Delta T)$	0.5	0	0	0
ED	$\mathcal{D}\text{eg}(\infty)$	$\mathcal{D}\text{eg}(2000)$	$\mathcal{D}\text{eg}(\Delta T)$	0.5	0.5	0	0
PE	$\mathcal{D}\text{eg}(\infty)$	$\mathcal{D}eg(0)$	$\mathcal{D}\text{eg}(\Delta T)$	0	0	0	0

network of surveillance camera (for instance with or without motion or pattern detection, mounted on robots), which send periodic reports to and interact with the control center, possibly in a completely automated way, until a critical event requiring the human intervention is detected. Depending on the type of applications, certain cases may require the deployment of proprietary networks, or they may be run on top of a standard LTE/LTE-A network or of a mesh network deployed for a specific need. Only the operational context may decide of the exact network architecture. The traffic for this scenario follows two different elementary patterns:

- **PU:** periodic very low data rate messages (GPS, photo, text, time) from the sensors to the control center.
- **ED:** event-driven very low data rate alarm signals from the control center to the corresponding authorities/organization.

Table II presents the traffic parameters for sensor-based alarm and event detection scenario. The corresponding graphical representation is given in Fig. 2. The sensor enters the PU state periodically to send a keep-alive message. When a event is detected, for example pressure drop through the pipelines, the sensor transfers to the ED state immediately to send the alarm message. The model could be extended to support transition from ED to PE state if larger amount of data should be sent to the server after occurrence of an event, for example transmission of a set of images or a video streaming upon detection of a movement.
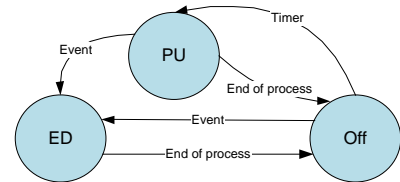

Fig. 2. Sensor based Alarm

Given the values presented in Table II and assuming $\Delta T = 1\,\text{s}$, the data rates can be calculated according to Eq. (1) to $R_s = 0, 1kB/s, 2kB/s, 0$ for the OFF, PU, ED and PE states, respectively. Solving Eq. (2) results in $\pi^e = (0.44, 0.22, 0.33, 0)^T$ for the embedded state probabilities and Eq. (3) yields $\pi = (0.99, 0.000275, 0.000412, 0)^T$ for the state probabilities of the SMM. Finally, assuming $N = 30\,000$ MTC devices of the same type and 10% signalling overhead amounts to an overall data rate of $R_{tot} = 36\,300\,\text{B/s}$, cf. Eq. (4).

IV. Discussions

The modeling approach presented above enables to emulate a broad range of MTC devices. We are able to accurately reproduce the traffic characteristics of such devices and allow for including control overhead, since this is done on the level of single machines. Fig. 3 shows the process of packet generation for MTC applications derived from the proposed traffic modeling framework, developed and validated in the context of the OpenAirInterface traffic generation (OTG) tool [15]. In the state S_i, the decision upon the transition to the next state is made when the sojourn/holding time T_i expires. During the T_i time, payloads/packets of a single node are generated according to the application-specified traffic model. If the traffic aggregation is applicable, payloads of multiple nodes are combined together to generate a single packet. Note that no traffic is generated within the off-state and during the state transition. The simplicity of the model allows for the simulation of highly populated quasi-synchronized communication scenarios (e.g. it might be used to generate traffic of 30 000 devices, which is the maximum amount of smart meter devices expected to be served by one cell in a densely populated urban area [3]).
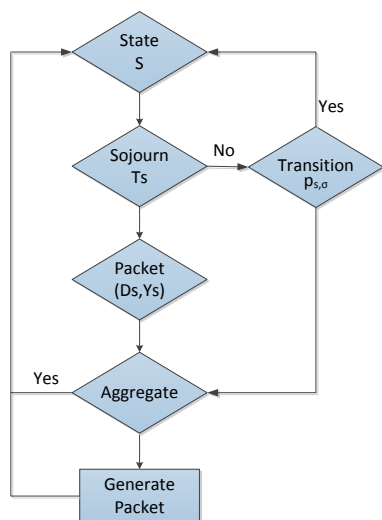

Fig. 3. Packet generation process

Different service classes and QoS requirements are inherently incorporated into the presented approach, due to the modular structure of the SMM. This allows for content-aware simulation/evaluation of the performance of next-generation networks. The motivation for such simulations is to accredit the fact that the ultimate metric for evaluating the performance of a network is user-satisfaction (not overall throughput as often assumed). For example, if the network infrastructure is able to forward the PE-type traffic with higher priority (hence, lower latency) than the PU-type traffic, the overall user-experience is improved. Future network-performance metrics will increasingly integrate diverse aspects of user-experience and therefore, traffic models equipped respective classification capabilities are needed.

V. Conclusion

To accommodate MTC over the emerging networks particularly LTE/LTE-A system, a state-full traffic modeling framework for MTC is proposed. It aims at building a complex application traffic through a combination of the elementary MTC traffic patterns, namely periodic update, (sporadic) event driven, and (bursty) payload exchange messages. With the proposed model, the traffic patterns and load for MTC can accurately be modeled and estimated, which is crucial for a better design and optimization of a system providing services for MTC applications. The model is analyzed through one real time sensor-based application scenario, where the elementary traffic patterns and load are extracted and estimated. Finally, the process of packet generation are derived from the proposed traffic modeling framework, which can be implemented by the traffic generation tool.

VI. Acknowledgments

References

[1] 3GPP, "Service Requirements for Machine-Type Communications," TS 22.368 V10.1.0, June 2010.

[2] Y. Morioka, "LTE for Mobile Consumer Devices", *ETSI Workshop on Machine to Machine Standardization*, 2011.

[3] 3GPP TR 37.868 v0.8.1 (2011-08),"Study on RAN Improvements for Machine-type communications (Release 10)".

[4] IEEE802.16p, Machine to Machine (M2M) System Requirements Document (SRD), Aug 2012.

[5] T. Taleb and A. Kunz, "Machine Type Communications in 3GPP Networks: Potential, Challenges, and Solutions," *IEEE Communication Magazine*, March 2012.

[6] S. Krco, J. Vuckovic, and S. Jokic, "ecoBus Mobile Environment Monitoring", Book Chapter in Towards a Service-Based Internet, 2010. Available on: en.wikipedia.org/wiki/BusPlus.

[7] M. Z. r Shafiq, L. Ji, A. X. Liu, J. Pang, J. Wang, "A First Look at Cellular Machine-to-Machine Traffic Large Scale Measurement and Characterization", SIGMETRICS 2012.

[8] P. Svoboda, M. Laner, J. Fabini, M. Rupp, F. Ricciato, "Packet Delay Measurements in Reactive IP Networks", IEEE Instrumentation & Measurement Magazine, 2012.

[9] LOLA Project (Achieving Low-Latency in Wireless Communications), "D3.5 Traffic Models for M2M and Online Gaming Network Traffic," 2011. Available on: www.ict-lola.eu/.

[10] H. Ekstrom, "QoS control in the 3GPP evolved packet system," *IEEE Communication Magazine*, 2009.

[11] S-Z. Yu, Z. Liu, M. S. Squillante, C. Xia, L. Zhang, "A Hidden Semi-Markov Model for Web Workload Self-similarity, " Proceedings of the Performance, Computing, and Communications Conference (PCC), 2002.

[12] A. Adas, "Traffic models in braodband networks," *IEEE Communication Magazine*, July 1997.

[13] S-Z. Yu, "Hidden semi-Markov models," Journal of Artificial Intelligence, Elsevier, 2010.

[14] R. Nelson, "Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling", Book, Springer-Verlag, 1995.

[15] A. Hafsaoui, N. Nikaein, L. Wang, "OpenAirInterface Traffic Generator (OTG): A Realistic Traffic Generation Tool for Emerging Application Scenarios", poster, MASCOTS, 2012. Available on svn.eurecom.fr/openair4G/trunk/openair2/UTIL/OTG/.