M2M Traffic and Models

Authors: M. Laner, N. Nikaein†, D. Drajic‡, P. Svoboda*, M. Popovic**, S. Krco‡*

*Address: * Vienna University of Technology, Austria, Email:firstname.name@nt.tuwien.ac.at, †EURECOM, France, Email:navid.nikaein@eurecom.fr, **Telekom a.d, Serbia, Email: milicapop@telekom.rs,‡Ericsson d.o.o, Serbia, Email: firstname.name@ericsson.com*

1. Introduction

Different from the traditional human-type communications (HTC) for which 3G 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.

The concept of M2M, also referred to as IoT, foresees that in the close future more and more devices will have their own internet access. This access will be some kind of wireless link towards a kind of home gateway, which itself is connected to a mobile network. As this scenario starts to take off operators of wireless cellular networks have to handle an explosive growth in signaling traffic inside their cells and even the core network.

In mobile networks the wireless access is in general a shared resource [3]. Therefore the number of active users, or devices, is limited. Therefore this resource is managed at the cost of signaling protocols in parallel to the user data streams from the base station. In human to human (H2H) connections these numbers are small, e.g., no more than four users are active in the same HSDPA time slot [4] in 95% of the time, and there are less than 100 users in a cell. In M2M the design target of 3GPP in [5] for devices per cell is 10.000. This value is several orders of magnitudes larger than compared to the H2H case. The activity patterns for M2M devices are also considerably different from H2H communication. In [6] the authors show a strong correlation in the activity patterns between the devices. This is a strong contrast to the common properties of independent arrivals used for example in an Erlang traffic model [7].

The MTC is a very active area under discussion in 3GPP for integration within the LTE/LTE-Advanced framework [8], and more generally within ETSI. Regarding 3GPP, a recent study item on provision of low-cost MTC devices based on LTE and a work item on system optimizations and overload control for MTC have been approved for LTE Rel-11. 3GPP LTE with low cost enhancements is expected to be one of the key MTC enablers.

However, the most challenging problems are 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 has revealed that in majority of cases, the MTC traffic has the following specific features [1], [9] :

- Short and small number of packets
- Low duty-cycle packets (i.e. long period between two data transmissions)
- Uplink-dominant packet
- State-full traffic patterns
- Real time and non-real time packets
- Periodic and event-driven packets
- Large number of ongoing parallel traffic sources
- Homogeneous and coordinated traffic (i.e. react in a synchronized fashion on global events)
- Raw and aggregated packets (i.e. combining traffic of multiple sources into a single packet, relevant for specific nodes such as gateway)

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 such as data, voice, and video. 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 traffic. 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 treatments (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 MTC sporadic traffic.

From these first thoughts we conclude that there is a need for M2M traffic models to test, validate and improve existing networks and that these models will differ from standard H2H models. In this chapter we will present overview of existing traffic models, and then M2M Traffic Modeling Framework will be introduced and explained in details. Principles and examples of M2M applications modeling and impact of M2M Traffic on Real Networks will be shown.

2. M2M Traffic Modeling

The topic of traffic modeling is very broad. It ranges from circuit switched voice models based on Erlang formulas, over packet switched queuing models to analysis of heavy tails in TCP streams and their source in the application structure.

In general M2M is not limited to any kind of service to transport its payload, e.g., it can use voice, SMS and IP datagrams. However with the introduction of LTE, which itself does not support any circuit switched voice any more, all applications can be mapped to IP-datagrams. In the following we will focus on packet switched traffic models (but for the sake of completnes, also circuit switch model will be mentioned). We are going to discuss different traffic models for different scenarios in the network.

2.1. M2M Traffic Modelling Activities in 3GPP, ETSI and IEEE

M2M is in the focus of the mobile industry for some time now and along with the ongoing activities in the research community, efforts towards understanding the impact of M2M on the mobile network architecture and specification of the relevant standards are under way (for example ETSI M2M, 3GPP and IEEE). The following table provides an overview of ongoing standardization in 3GPP, ETSI and IEEE.

Reference	Description
3GPP TR 43.868 [11]	GERAN Improvements for Machine-type Communications
3GPP TR 37.868 [12]	Study on GERAN Improvements for MTC
3GPP TS 22.368 [8]	Network Improvements for Machine-Type Communications
3GPP TS 22368 [1]	System Improvements for MTC
IEEE802.16p [13]	Machine to Machine (M2M) System Requirements Document (SRD)
ETSI TS 103 092 [14]	Machine-to-Machine communications (M2M); OMA DM compatible Management Objects for ETSI M2M
ETSI TR 102 935 [15]	Machine-to-Machine communications (M2M); Applicability of M2M architecture to Smart Grid Networks; Impact of Smart Grids on M2M platform
ETSI TS 103 093 [16]	Machine-to-Machine communications (M2M); BBF TR-069 compatible Management Objects for ETSI M2M
ETSI TS 102 921 [17]	Machine-to-Machine communications (M2M); mIa, dIa and mId interfaces
ETSI TS 102 690 [18]	Machine-to-Machine communications (M2M); Functional architecture
ETSI TR 103 167 [19]	Machine to Machine (M2M); Threat analysis and counter measures to M2M service layer

ETSI TS 102 689 [20]	Machine-to-Machine communications (M2M); M2M service requirements
ETSI TR 102 691 [21]	Machine-to-Machine communications (M2M); Smart Metering Use Cases

Table 1. 3GPP, IEEE and ETSI references for M2M

The table depicts the common approach followed in 3GPP, ETSI and IEEE. There are work items on system improvement [11], radio access network improvement [8, 12] and service requirements [1]. However there is not a dedicated specification on traffic models for M2M devices. In fact there are various different models provided for the different tasks and optimization analysis given in [1, 8, 12].

2.1.1. M2M Activities in IEEE 802.16p

The IEEE standardization invoked a working group on M2M in the framework of CDMA. The IEEE Machine-to-Machine (M2M) Task Group was initiated in 2010 to work on the 802.16p and 802.16.1b projects. Both standards have been approved by the IEEE in 2012. IEEE 802.16's Machine-to-Machine (M2M) Task Group is a relevant resource in terms of traffic characteristics and traffic models for Smart Grids and M2M applications. The standard contains two tables providing a good overview of M2M traffic patterns. The following two tables are references from the document [13]. They depict average message size, transaction rate and data rates combined with a distribution of the arrival process in the traffic stream.

Appliances/Devices	Average Message Transaction Rate/s	Average Message size (Bytes)	Data Rate (b/s)	Distribution and arrival
Credit Machine in grocery	0.0083	24	0.2667	Poisson
Credit Machine in shop	5.5556e-4	24	0.0178	Poisson
Roadway Signs	0.0333	1	0.2664	Uniform
Traffic Lights	0.0167	1	1.3360	Uniform
Traffic Sensors	0.0167	1	1.3360	Poisson
Movie Rental Machines	1.1574e-5	152	1.4814e-3	Poisson

Table 2. City Commercial M2M Devices Traffic Parameters [13]

Scenario	Number of grocery stores/m2	Number of shops and restaurant/m2	Number of roadway signs/m2	Number of traffic lights/ m2	Number of traffic sensors/ m2	Number of Movie Renatl Machines/ m2
Urban	2.0947e-4	0.0022	3.1647e-4	1.503e-5	1.503e-5	6.9823e-5

(New York City)						
Sub-urban (Washington D.C.)	2.3122e-5	3.4988e-4	9.4325e-4	1.1442e-4	1.1442e-4	1.1561e-5

Table 3. City Commercial Facilities deployment [13]

2.1.2 M2M Activities in ETSI

The ETSI standardization body contributes in different technology. It is organized in clusters following a so called work program. There is a dedicated M2M activity, the name used in ETSI is Machine to Machine Communications. The resulting documents have been depicted in Table 1. The work so far focuses on three different layers. The main work in these documents focuses on the higher protocol layers and the management for M2M devices. From the perspective of traffic modeling approach there are two documents [20, 21] of main interest. In [20] defines the basis for M2M communication related technical specification. It presents the general and functional requirements for M2M communication services. In [21] explicit use cases are discussed. The topic is a detailed discussion on the setup of a smart meter scenario done in M2M.

Concluding the ETSI documents, there are no explicit traffic models in the current publications. However, the named documents are useful to outline simulation setup for M2M scenarios.

2.1.3 M2M Traffic Model proposed in 3GPP

The work on M2M in 3GPP specifications for cellular mobile technologies started in Rel-10. The item was generalized into the topic of machine type communications (MTC) offering not only the concept of devices but also of infrastructural elements like servers and processing units.

In Rel-10 the scope of 3GPP was to implement the support for a large number of M2M devices in the mobile networks, e.g., UMTS or LTE, and fulfill certain service requirements. In the upcoming Rel-11 the scope moved to further improvements of the mobile networks for large number of devices. Finally Rel-12 will focus on new ways to allow for cheaper and simpler devices, see [11]. In the following discussion we will focus on Rel-10 of the 3GPP standard.

The general terms M2M and MTC may be a bit misleading as they in fact are not one type of application but rather a cluster of different applications. MTC applications do not all have the same characteristics, meaning that not every system optimisation is suitable for every MTC application, so MTC Features (Requirements) are defined to provide structure for the different system optimisation possibilities.

The general requirements [1] identified as service requirement for all MTC devices are:

- Time Controlled,
- Time Tolerant,

- Small Data Transmissions,
- Mobile Originated Only,
- Infrequent Mobile Terminated,
- MTC Monitoring,
- Priority Alarm,
- Secure Connection,
- Location Specific Trigger,
- Infrequent Transmission ,
- Group Based MTC Features,
    - Group Based Policing,
    - Group Based Addressing.

MTC Requirements provided to a particular subscriber are identified in the subscription, and can be individually activated.

The technical report [12], which deals with GERAN improvements, is based on a scenario for smart meters. The designed traffic model assumes mobile traffic to be of packet switched nature only. The traffic is mobile originated, meaning that there is no polling of information from the MTC server side. Therefore the MTC device will run through a cycle of autonomous accesses to the network and there is no network based ringing. The document identifies the control channels as the main limitation in this scenario, therefore the traffic model is focused on reproducing the property of the common control channel (CCCH).

In the traffic model presented in [12], in the first step the generic traffic model for M2M devices is split into three different classes T1, T2 and T3 describing synchronous and asynchronous access to the network. MTC devices of class T1 access the network in a non-synchronized way. An example scenario for this would be a set of MTC devices of different applications in the same cell. MTC devices of the class T2 access the network in a synchronized way. An example scenario for this is a smart meter setup. Here all meters are expected to deliver synchronized reports based on a fixed time grid. Devices of class T3 are generic legacy devices generating uncoordinated background traffic in the cell.

In the second step three different traffic patterns are defined for the classes T1, T2 and T3. The following table shows the definitions found in [12]. The number of active nodes is modeled via the arrival rate λ in T1 and T3 or via the total number of nodes X. The patterns for T1 represent the pure M2M device, which is due to the expected large amount of users modeled as a Poisson arrival process.

Scenario T2 is a special case of scenario T1. Here the devices are assumed to be time synchronized within a small interval of time T, either due to misconfiguration or due to external events, e.g., power outage. Finally scenario T3 considers legacy CS (Circuit Switch) and PS (Packet Switch) devices, modeling the “normal” users in the cell. This scenario placed in parallel with either T1 or T2 can be used to show the impact of MTC on the normal traffic. Again a Poisson arrival is assumed.

Scenario	**T1**	**T2**	**T3**
Number of devices	λ / (Reporting interval)	X	λ / (Reporting interval)
Arrival process	Poisson Arrival intensity: λ [arrivals/second]	Time limited deterministic event distribution. The time-spread of the distribution is controlled by parameter T [s], which shall include T=1.	Poisson Arrival intensity: λ [arrivals/second] Case 1: λ = 5 for CS traffic (only CS traffic is present in the cell) Case 2: Like Case1 with additional λ = 15for PS traffic (combination of CS and PS traffic in the cell)
Reporting interval	• 5 seconds • 15 minutes • 1 hour • 1 day	NOTE: With this traffic model reporting interval is not defined since the number of devices are fixed and the access need to be finished by all devices before the following access can take place.	----
Report Sizes	• 10 byte • 200 byte • 1000 byte	• 10 byte • 200 byte • 1000 byte	----

Table 4: CCCH Arrival Patterns for Device Type Scenario T1, T2 and T3 [12]

In the first two steps three scenarios are defined and individual arrival process for each of them. Now, the distribution in time of the deterministic events in the M2M communication will be defined. For the given time interval of the duration t=T, the intensity of service request arrivals is given as a distribution p(t) for all the X devices in an area. There are two different distributions functions considered for *p(t)*, namely uniform distribution:

$p(t) = 1/T$, *for* $0<t<=T$; else $p(t) = 0$ (1)

and the second one, Beta distribution:

$$p(t) = \frac{t^{\alpha-1}(T-t)^{\beta-1}}{T^{\alpha+\beta-1}Beta(\alpha,\beta)} \alpha > 0, \beta > 0$$, where $Beta(\alpha,\beta)$ is the Beta function. (2)

The distribution has two tunable parameters (shape parameters alpha and beta) to allow for different peaks of intensity in the parallel active devices. 3GPP proposes alpha = 3 and beta = 4. Both functions have a well-defined support on the time axis between zero and T.

The number of devices in case of [8] is shown in the following table:

Characteristics	Traffic model 1	Traffic model 2
Number of MTC devices	1000, 3000, 5000, 10000, 30000	1000, 3000, 5000, 10000, 30000
Arrival distribution	Uniform distribution over T	Beta distribution over T, See section 6.1.1
Distribution period (T)	60 seconds	10 seconds

Table 5: Traffic Model Parameters in LTE [8].

All upper layer traffic on data channels in the network are considered to be derived from this input via simulation results. This concludes the actual 3GPP traffic model the RAN, which targets to reproduce only activity patterns at the access plane so far.

2.2. M2M Traffic Modeling Framework

The first traffic models presented so far for M2M are pure generic traffic models not considering the different type of application running in the framework of MTC. In the follow we focus on traffic models describing different forms of activity patterns driven by an application based approach. This kind of traffic modeling is called source traffic modeling as each source itself is an instance of a model itself. In the following table there is a short overview of the different categories of MTC applications.

Category	**Application**	**Traffic** **Direction / Devices / Delay / Intensity**
Health	Monitoring of vital signs Emergency support Remote telemedicine	Uplink / few / low / small
Metering / Controlling	Smart meters Smart grid Car to Car	Uplink / many / low / variable Security and Time critical
Surveillance / Security	Sensors Video Surveillance Audio Surveillance	Uplink / many / low / small Uplink / few / low / high
Tracking	Asset tracking Fleet management Team tracking	Uplink / many / low / small
Payment	Vending machines	Uplink / many / low / small

Table 6. MTC Applications and expected Traffic Patterns, [22].

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 [23]:

- **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 a 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 is 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 may further consist of a combination of the above-mentioned traffic types. 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 EDs traffic at regular intervals, thus making the traffic pattern a periodic event driven (PED). Furthermore the PE may happen after the (P)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*, *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.

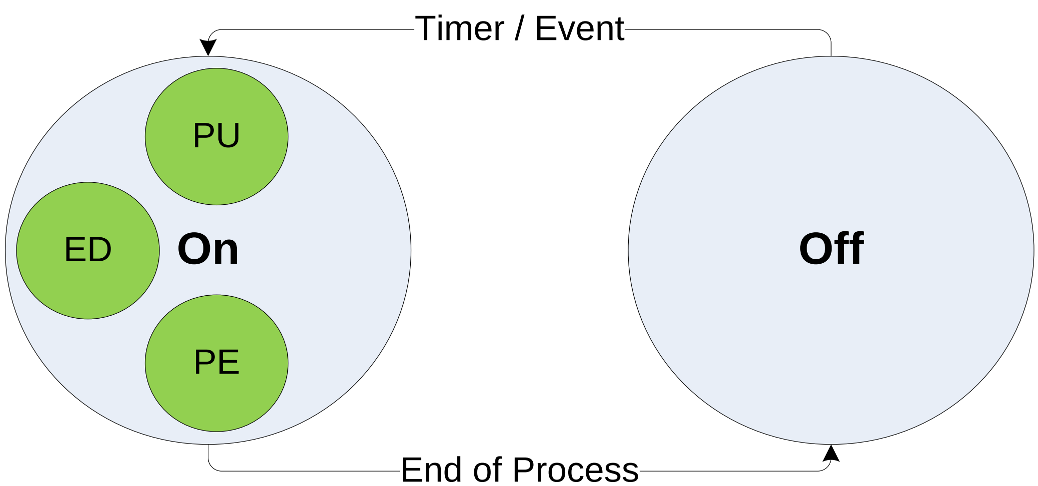

Figure 1. Generic M2M Traffic Model Entity

For modeling the data streams within single states s we deploy renewal processes [24], 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) [2], 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 $\boldsymbol{P}$. Further, a random sojourn time or holding time T_s is introduced per state, with arbitrary independent distribution $f_{T,s}(t)$ [25]. 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 [26], especially the almost vanishing data-rate, (ii) enable augmented modeling if side-information is available (e.g., the exact number of states are known) [27] and (iii) advanced fitting mechanisms are established [28], which allow for good fitting quality, even if nothing but raw traffic-measurements are given.

The input parameters for the model are summarized in Tab. 1, where "·" represents parameters to be fitted to a desired MTC traffic pattern and the completed items are state specific constants. Thereby, $\mathcal{D}$eg(·) 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.

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	$\boldsymbol{P}$			
OFF	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(0)	·	0	·	·	·
PU	$\mathcal{D}$eg(∞)	·	$\mathcal{D}$eg(ΔT)	·	0	·	·
ED	$\mathcal{D}$eg(∞)	·	$\mathcal{D}$eg(ΔT)	·	·	0	·
PE	·	·	·	·	·	·	0

Table 7. Traffic model input parameters

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 [8], 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., $p_{\mathrm{OFF,PU}}$)
- $\boldsymbol{P}$: 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

$$R_s = \frac{\overline{Y}_s}{\overline{D}_s}. \tag{3}$$

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

$$\boldsymbol{\pi}^e = \boldsymbol{\pi}^e \boldsymbol{P}, \qquad \text{under} \quad \sum_{s=1}^{S} \pi_s^e = 1. \tag{4}$$

They are further used to calculate the actual state probabilities of the SMM [24], p. 353 by

$$\pi_s = \frac{\pi_s^e \overline{T_s}}{\sum_{\sigma=1}^{S} \pi_\sigma^e \overline{T_\sigma}}. \tag{5}$$

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

$$R_{tot} = \rho_{\text{OH}} \sum_{n=1}^{N} \sum_{s=1}^{S} \pi_{s,n} R_{s,n}, \tag{6}$$

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

Note, that this model is reproducing the traffic of each single machine, which in turn does not mean that any correlations between machines can be captured. For example, assume hundreds of temperature sensors are spread over a small area, on which temperature is uniformly passing a threshold at a certain point of time. In that case all sensors would trigger simultaneously, causing strong congestion in the network. Such cases are not captured by our model, since they would require a joint modeling of all sensors.

2.2.1 Modeling M2M Applications

Although a large variety of M2M application scenarios with heterogeneous requirements and features exists, they can be classified into two main M2M communication scenarios as defined in [1], direct communications among MTC devices and/or communications from MTC devices to a set of MTC servers/users. In the following subsections, two MTC applications with different communication scenario are described and their traffic patterns are evaluated [9].

AutoPilot

As described at the beginning of this chapter there are many different M2M applications. In the following we will give one example to show how the statefull/state-aware model above could be implemented into a real world application as a source traffic model. The application selected is Auto-Pilot. This scenario includes both vehicle collision detection and avoidance (especially on highways) and how the urgency actions are taken in the case of an accident. It is based on a M2M device equipped with sensors embedded in the cars and surrounding environment and used in automatic driving systems. These M2M devices (cars, road sign units, highway cameras) send information to a backend collision avoidance system. The backend system distributes notifications to all vehicles in the vicinity of the location of the collision, together with information required for potential actuation of relevant controls in the affected cars. In all receiving cars, the automatic driving systems based on the received information take over the control fully or partially (brakes activated, driving direction changed, seating belts tightened, passengers alerted). If there is no such system in a car, the driver is notified and instructed. Also, depending on the proximity of the accident, different commands are sent to the cars, i.e. the cars which are closer to the place of the possible collision are getting immediate commands for the actuators, while the cars which are further away from this place get driver notifications only. Three main traffic patterns can be identified in this scenario:

- PU: low data rate update messages (GPS, speed, time) from the M2M devices to the backend system and notifications from the backend system to the M2M device.
- ED: short burst emergency packet from the M2M devices to the backend system.
- PE: actuation commands from the M2M backend to the M2M devices.

We assume that cars at least send information about time, position and velocity, and that it corresponds to a packet length up to 1 kB (in various tests from the M2M devices to the backend system, the packet length varied from 64 B to 1 kB, usually being 100 B; while for V2V communications it was 149 B). The frequency of the packets was usually 10 pps-packets per second (i.e. packet was send every 100 ms). For high speeds, cars should send one packet every meter (resolution of GPS). At a speed of 160 km/h (44.5 m/s) the number of packets sent from the cars will be about 45 packets per second (period 20-25 ms). So, data rates are in the range of 10kB/s for low velocities, and up to 45kB/s for high velocities. The number of cars varies, depending of the traffic intensity and the length of the surveyed track. With a small and medium number of cars, the actual throughput is not critical as the amount of traffic generated by a car will be small. In collision avoidance, acceptable values for the length of the track under surveillance are about 1 km. It is also acceptable that the observed zone is populated with up to 50 cars. In the emergency situations the frequency of packets from the cars should be higher, e.g. 100

pps (period 10 ms), and data rates be up to 100kB/s. In the case of an accident or possible collision, backend system sends event-driven, short bursts packets of 1-2 kB to the cars every 10 ms, which correspond to 100-200 kB/s per car. The number of the cars highly depends on the time of the day, and the day of the week. For the peak hour we can assume that maximum number of the cars on the 1 km track should be 50, and for that case the cell capacity limitations have to be considered. If everything is normal on the road, backend system can periodically (about every 1s) send some notification messages to the cars with packet length of 1kB. So, on the every km of the highway we have 50 terminals with sensors registered to the network, who are exchanging data with application server continuously.

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	$\boldsymbol{P}$			
OFF	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(0)	$Exp(2)$	0	0.5	0	1
PU	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(1000)	$\mathcal{D}$eg(ΔT)	0.4	0	0	0
ED	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(1000)	$\mathcal{D}$eg(ΔT)	0.6	0.5	0	0
PE	$\mathcal{D}$eg(0.1)	$\mathcal{D}$eg(1000)	$\mathcal{D}$eg (1)	0	0	1	0

Table 8. Traffic parameters for AutoPilot downlink scenario, values in seconds and bytes

Table 8. shows the analysis for the peak hour of the traffic on the highway and results need to be scaled for different time intervals of the day/week. The D_{OH} is the sum of the TCP/IPv4 (40 B), PDCP (2B), RLC-AM(4B), and MAC(4B) header size, and estimated as 50 bytes. Typical number of nodes for this scenario depends on the considered area and its density, and assumed to be 50. The distribution of the cars speeds could be the following: 10% will be low-speed drivers, 60% will be medium speed drivers and 30% will be high speed drivers.

Fig. 2 depicts the state diagram for the autopilot reference model. When everything is normal, a sensing device periodically enters to the PU state to send update messages and receive notifications from the backend collision detection system. When an accident occurs it enters to the ED state and triggers an event (i.e. collision avoidance); after that it enters to the PE state to exchange information with the backend system.

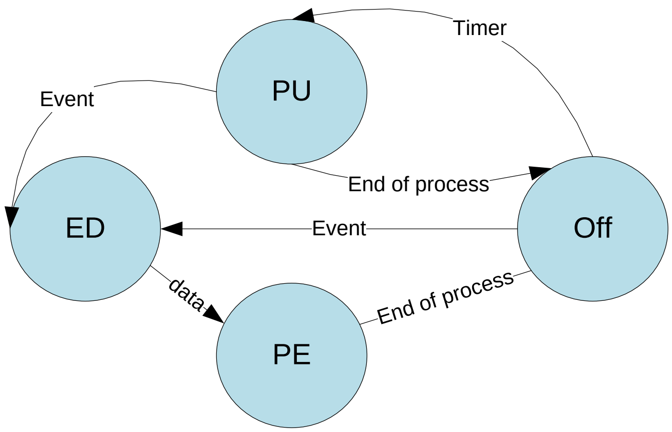

Figure 2. State Transition Diagram for Auto-Pilot Reference Model

Sensor-Based Alarm or Event Detection

Many categories of 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, stability of buildings, critical parameters in plants, etc. Of course the type of power supply (if the sensor is always on or not), density and other parameters depend on the application. Another type of application is event detection requiring fast reaction. 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 can enable interesting applications too. Examples of this type of applications can be intrusion detection sensors, or automated network of surveillance camera (with or without motion or pattern detection, mounted or not on robots, for instance), 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 also two different 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 9. presents the traffic parameters for sensor-based alarm or event detection scenario. It can be seen than the smoke detector generates PU more frequently than that of humidity and temperature sensors as this type of sensors are time-critical and requires very fast reaction time.

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	$\boldsymbol{P}$			
OFF	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(0)	$\mathcal{D}$eg(30 min)	0	0.5	1	1

PU	$\mathcal{D}eg(\infty)$	$\mathcal{D}eg(1000)$	$\mathcal{D}eg(\Delta T)$	0.5	0	0	0
ED	$\mathcal{D}eg(\infty)$	$\mathcal{D}eg(2000)$	$\mathcal{D}eg(\Delta T)$	0.5	0.5	0	0
PE	$\mathcal{D}eg(\infty)$	$\mathcal{D}eg(0)$	$\mathcal{D}eg\ (1)$	0	0	0	0

Table 9. Traffic Parameters for Uplink Sensor Based Alarm Scenario, values in seconds and bytes

A reference model is depicted in Fig. 3. The sensor enters the PU state periodically to send a keep-alive message. When an event is detected, for example pressure drop through the pipelines, the sensor transfers to the (P)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.

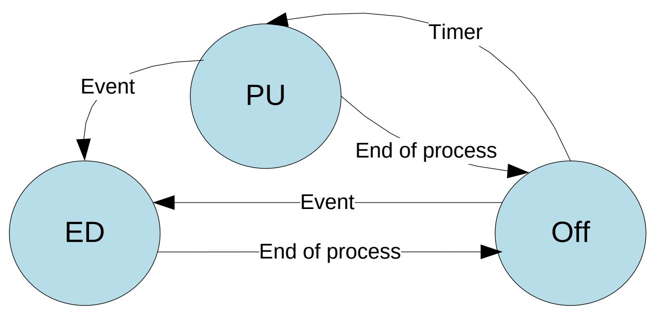

Figure 3. State Transition Diagram for Sensor based alarm and event-detection

Virtual Race

One example of the many possible MTC games is the virtual race (e.g. virtual bicycle race using real bicycles). The opponents are on different locations, possibly many kilometers away. At the beginning, the corresponding length of a race is agreed (i.e. 10 km or 20 min) between the peers. The measurements are taken by sensors (GPS, temperature, humidity, speed, terrain configuration etc.) and are exchanged between the opponents. They are used by the application to calculate the equivalent positions of the participants and to show them the corresponding state of the race (e.g. “you are leading by 10 m”). The number of competitors may be more than two, and all competitors must mutually exchange information, and the applications must present all participants the state of other competitors. For a large number of competitors (hundreds or more), a corresponding application server must be used. During the race they are informed about the place and the distances from each other (e.g. “you are the 3rd behind the 2nd by 10 m and leading before the 4th by 15 m”).

One traffic pattern can be seen here:

- PU: low data rate update message with shorter periods as the end of the race is getting closer (i.e. monotonic decreasing inter-departure time).

The packets containing GPS and sensors data are on the order of 1 kB. The D_{OH} is 50 bytes similar to the auto-pilot scenario. Taking into account the typical speeds (of bicycles) in this scenario (rarely higher than 50 km/h = 13.9 m/s), the packets should be exchanged approximately every 100 ms, which corresponds to a resolution of 1.4 m. Also we can assume that competitors have the periods of low and medium speeds during the competition which corresponds to 10 and 30 km/h. This highly depends on the road topology, but we can assume that 20% of the competition time riders will have a low speed, than 60% of the time medium speed and finally 20% they will drive very fast. If there is only two (or small number of competitors), there is no need for application server. In the case of the higher number of competitors (or team competition) there will be a need for an application server. The application should be aware of the positions of all competitors with respect to the end of the race, and, when the competitors are close to the finish, packets should be sent every 70 ms, which corresponds to a resolution of 1 m (GPS accuracy). Data rates are normally not higher than 10 kB/s (about 15 kB/s at the final stage of the competition). Typical number of competitors considered in this scenario is less than 100.

Since application is continuously sending data from the beginning of the race without any trigger, we can treat it as the PU traffic. With a small and medium number of competitors, the actual throughput is not critical as the amount of traffic generated by a user will be small.

Table 10. presents the traffic parameters for the virtual race scenario. It should be noted that the same traffic pattern could be achieved by only using the PU state, i.e., with constant inter-packet time of 100ms, packet size of 1000Byte and a sojourn time of infinity.

State s	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	$\boldsymbol{P}$			
OFF	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(0)	$\mathcal{D}$eg(100ms)	0	1	1	1
PU	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(1000)	$\mathcal{D}$eg(ΔT)	1	0	0	0
ED	$\mathcal{D}$eg(∞)	$\mathcal{D}$eg(0)	$\mathcal{D}$eg(ΔT)	0	0	0	0
PE	$\mathcal{D}$eg(∞)	$\mathcal{D}eg$(0)	$\mathcal{D}$eg(ΔT)	0	0	0	0

Table 10. Traffic Parameters for Uplink Virtual Race scenario, values in seconds and bytes

Fig. 4. depicts the state diagram for the different states of the virtual race reference model. There are two states: PU and OFF state. So, competitor periodically enters the PU state to send its data to the application server and receive ranking information from application server.

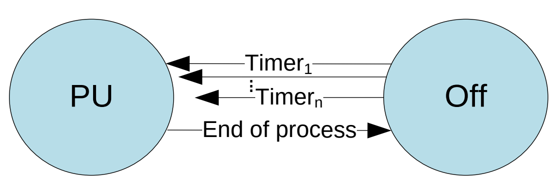

Figure 4. State Transition Diagram for Virtual-Race

3. Impact of M2M traffic on contemporary networks (HSPA)

In order to evaluate possible impacts of M2M traffic on contemporary mobile networks, in coexistence with traditional (human-originated) traffic, series of simulations have been performed in a real HSPA network. M2M traffic has been simulated through a traffic generator application installed on phones running Android OS, communicating with a remote server [9, 29]. Several traffic patterns have been chosen from scenarios depicted previously, namely:

- Bicycle Race (BR):
    - Virtual M2M game, where two or more players exchange data on position, speed etc.
    - Model chosen: 1kB packets exchanged with uniformly distributed inter-arrival time ranging from 0.1 to 0.5s
- Autopilot (AP):
    - Clients sending data on position, in time intervals depending on vehicle speed, while server performs calculations, collision detection etc, and sends back control information
    - Model chosen: 1kB packets sent towards the server with uniformly distributed inter-arrival time ranging from 0.025 to 0.1s, server responds every second with 1kB message
- GPS Keep Alive messages in Team Tracking applications (TT):
    - Clients with team members sending data on position, depending on activity
    - Model chosen: 0.5kB packets sent with uniform inter-arrival time distribution ranging from 1s to 25s

Along with 6 M2M client phones, 4 phones running online-gaming (OG) traffic models have been used (Open Arena (OA) and Team Fortress (TA)). More about these measurements can be found in [9, 29]. TCP protocol was used for transmission. The throughput of the above described applications varies a lot (Table 11), generally from less than 1kbps to 320 kbps. Application uplink (UL) and downlink (DL) traffic patterns have all been tested in network uplink.

Phone	Packet length distribution, packet inter-arrival time distribution	Application	Avg. packet size (bytes)	Avg. time between packets (s)	Max throughput (kbps)	Min throughput (kbps)
1	Gauss (0,04121;0,004497)kB, Uniform(0,069;0,103)s	OG,OA, UL	40	0.086	6.68	1.82
2	Gauss (0,07473;0,013085)kB, Uniform(0,031;0,042)s	OG,TF, UL	75	0.0365	33.27	5.21
3	Gauss (0,16836;0,08381)kB, Uniform(0,041;0,047)s	OG,OA,DL	170	0.044	94.32	0.17

4	Gauss (0,23511;0,07748)kB, Uniform(0,039;0,046)s	OG,TF, DL	240	0.0425	117.39	0.17
5	Constant(1)kB, Uniform(0,1;0,5)s	M2M, BR, UL	1024	0.3	80.00	16.00
6	Constant(1)kB, Uniform(0,1;0,5)s	M2M.BR, DL,	1024	0.3	80.00	16.00
7	Constant(1)kB, Uniform(0,025;0,1)s	M2M, AP, UL	1024	0.0625	320.00	80.00
8	Constant(1)kB, Uniform(0,999;1,001)s	M2M, AP, DL	1024	1	8.01	7.99
9	Constant(0,5)kB, Uniform(1;25)s	M2M, TT(GPS Keep Alive), UL	512	13	4.00	0.16
10	Constant(0,5)kB, Uniform(1;25)s	M2M, TT(GPS Keep Alive), UL	512	13	4.00	0.16

Table 11. Traffic characteristics

The serving Node B has been upgraded in the course of testing, and network modernized, enabling thorough insight into the effects M2M traffic might have with different network configurations, but same traffic patterns used on top of regular users' traffic. Standard network KPIs and counters related to data and voice traffic have been monitored, gathered from the network OSS. Main areas of quality of service from the end-user perspective, accessibility, retainability and integrity, as defined by (30, 31), have been analyzed through KPIs, along with the latency recorded on phones via the traffic capturing application.

The analysis has shown that the main impact is expected in the area of accessibility, i.e. the ability of a service to be obtained, within specified tolerances and other given conditions, when requested by the user. Not only the PS accessibility was affected, but the CS accessibility as well.

The serving Node B was situated in a highly urban area, with rather modest resources in first test cases, but with stable performance concerning regular users' traffic. The addition of 6 M2M and 4 online-gaming test users, with uplink-oriented traffic, led to severe KPI degradation, PS accessibility dropping to 0%, and CS accessibility below 80%. The number of active PS connections increased, as well as the number of attempts to establish the radio bearer. For these cases, the lack of processing power and the license for a small number (relative to the traffic) of simultaneous HS users were identified as the main bottlenecks.

Yet, the KPIs were showing that the lack of resources needed to establish the service was a trigger to a more serious effect – a signalling congestion. The initial lack was a reason for the Node B to reject requested PS service, but the drop of CS accessibility occurred mainly due to signalling congestion created by repeated PS requests of machine users.

Further test cases proved that with the increase of processing power and number of simultaneous HS users accessibility returned to its normal level of 100%, or nearly 100%. RAB establishment attempts also returned to their normal daily fluctuation. Yet, as the uplink is generally more critical than the downlink in modern networks designed for downlink-oriented traffic, only the further increase in capacity led to satisfactory results concerning latency. Although main KPIs returned to their normal level even with first upgrades, many users were still pushed down to common channels, offering very low throughput and consequently high latency. This may be seen through network KPIs, but is not alarming from the network performance point of view. So, for latency-critical M2M applications involving some number of clients in a cell, the cell needs to support the requested number of connections, as well as to have enough spare capacity to accommodate the throughput demands for the uplink. The stable accessibility is the necessary, but not the sufficient condition for end-user quality of service.

The deployment of a real large-scale M2M application provided opportunity to further confirm results obtained from simulations and generalize the conclusions, revealing the underlying mechanism of positive feedback. The packets sent by client applications were very sporadic, so the client modems were generally in Idle state, establishing a RRC connection only to send a packet and going back to Idle. Again the accessibility was affected, PS as well as CS, with increased number of connections and huge number of RAB establishment attempts. In this case, large number of users going from Idle state to RRC connected state created a signalling congestion, which deteriorated further as the lack of downlink channelization codes led to rejection of new connections, and repeated requests from M2M users.

Main conclusions drawn in the analysis are as follows:

- The persistence of M2M users, in a situation where the Node B lacks any of the resources necessary to assign a RAB, i.e. give service, leads to repeated attempts creating congestion on signalling channels, which then leads to further drop of accessibility and further attempts – positive feedback mechanism. In 3G network, although voice has priority, this affects the voice service, due to inherent properties of the technology. Human users do not show such persistence as devices.

- The effects depend on the number of M2M users relative to the Node B capacity.

- Traffic pattern itself has an influence on the network. Clients with sporadic traffic, with long times between packets, will reside in the Idle state, and generate signalling every time they want to send a packet i.e. get RRC connected. States allowing for the terminal to stay RRC connected for a longer period of inactivity may improve the situation.

- The massive number of M2M users creates signalling congestion in the very start, and any lack of resources just worsens the situation.

- Accessibility improvement is the necessary, but not the sufficient condition to fulfil end-user QoS requirements. For latency-critical applications, the cell needs to have enough spare resources to support uplink throughput demands.

- Traffic aggregation could solve the problem with huge number of connections and signalling congestion, but latency requirements still need to be addressed, by assessing performance in this respect and increasing resources to a satisfactory level.

4. Summary and Conclusion

In this chapter an overview of the state of the art in M2M traffic modeling was presented. Compared to the human to human interaction in the communication the M2M based applications have different properties in traffic and device numbers. The traffic is mainly directed in uplink and the number of devices is expected to be several orders of magnitudes larger than compared to the human driven devices.

The traffic models derived in the standardization bodies of 3GPP and IEEE target currently the overload scenarios in the access network. Therefore they consider pure uplink traffic and device numbers of more than 10000 per cell. While this is a good approach for link level simulations providing large samples for user traffic in a short amount of time, the actual structure of the application traffic is not considered. Recent research activities move the focus from one model for all users to a source traffic approach where each device is modeled as a traffic source based on a HSMM. These models allow different types of M2M device in the same simulation, at the increased cost of computational complexity per added node in the MTC domain.

The validation of the per source approach concludes this chapter about traffic modeling it shows that nowadays network a number as low as ten active devices with generic M2M traffic patterns are generating an impact not only to the packet domain of the network but rather more also on the circuit, e.g., voice domain of the network.

At the current state of mobile cellular networks a per-device source approach can be favored for simulation and or emulation on the IP network.

References

[1] 3GPP TS 22368, Service requirements for Machine-Type Communications, 2012

[2] A. Orrevad, "M2M Traffic Characteristics (When machines participate in communication)," KTH Information and Communication Technology, 2009.

[3] Holma, H. and A. Toskala (2010). WCDMA for UMTS: HSPA Evolution and LTE, Wiley.

[4] Laner, M. et al., Users in Cells: a Data Traffic Analysis, in Proc. Proceedings of the Wireless Communications and Networking Conference, WCNC'12, Paris, France, April, 2012.

[5] 3GPP TS 23.888, System improvements for Machine-Type Communications, 2011.

[6] Zubair Shafiq, M. et al., A First Look at Cellular Machine-to-Machine Traffic – Large Scale Measurement and Characterization, in Proc. Proceedings of the SIGMETRICS 2012, London, UK, June, 2012.

[7] ITU-T Study Group 2, Teletraffic Engineering Handbook, June 2006.

[8] 3GPP, TS 37868, Study on RAN Improvements for Machine-type, 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] 3GPP TS 36888, Study on provision of low-cost MTC UEs based on LTE, 2012.

[12] 3GPP TS 43868, GERAN Improvements for Machine-type Communications, 2012.

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

[14] ETSI TS 103 092 Machine-to-Machine communications (M2M); OMA DM compatible Management Objects for ETSI M2M, May 2012

[15] ETSI TR 102 935 Machine-to-Machine communications (M2M); Applicability of M2M architecture to Smart Grid Networks; Impact of Smart Grids on M2M platform, Sep 2012

[16] ETSI TS 103 093 Machine-to-Machine communications (M2M); BBF TR-069 compatible Management Objects for ETSI M2M, May 2012

[17] ETSI TS 102 921 Machine-to-Machine communications (M2M); mIa, dIa and mId interfaces, Feb 2012

[18] ETSI TS 102 690 Machine-to-Machine communications (M2M); Functional architecture, Oct 2011

[19] ETSI TR 103 167 Machine to Machine (M2M); Threat analysis and counter measures to M2M service layer, Aug. 2011

[20] ETSI TS 102 689 Machine-to-Machine communications (M2M); M2M service requirements, Aug 2010

[21] ETSI TR 102 691 Machine-to-Machine communications (M2M); Smart Metering Use Cases, May 2010

[22] LOLA Project (Achieving Low-Latency in Wireless Communications), "D2.2 Target System Architectures," 2010. Available on: www.ict-lola.eu/

[23] A. Hafsaoui, N. Nikaein, C. Bonnet, Analysis and Experimentation with a Realistic Traffic

Generation Tool for Emerging Application Scenarios, Emutools 2013.

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

[25] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proceedings of the IEEE, 1989.

[26] 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.

[27] A. Adas, ”Traffic models in braodband networks,” IEEE Communication Magazine, July 1997.

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

[29] D. Drajic, M. Popovic, N. Nikaein, S. Krco, P. Svoboda, I. Tomic, N. Zeljkovic, ”Impact of online games and M2M applications traffic on performance of HSPA radio access networks”, International Workshop on extending seemlessly to the Internet of Things (esIoT), 2012.

[30] ITU-T, Recommendation E.800 (1994): „Terms and definitions related to quality of service and network performance including dependability“

[31] 3GPP TS 32.450: „3rd Generation Partnership Project, Technical Specification Group Services and System Aspects; Telecommunication management; Key Performance Indicators (KPI) for E-UTRAN: Definitions (Release 8)