Traffic models for machine-to-machine (M2M) communications: types and applications

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

Abstract: Machine-to-machine (M2M) or Machine-type Communication (MTC) is expected to have a significant traffic share in future wireless networks. It exhibits considerably different traffic patterns than human-type communication and, thus, requires new traffic models and simulation scenarios. Such models should (i) accurately capture the behaviour of a single MTC device as well as (ii) enable the concurrent simulation of massive numbers of devices (e.g., up to 30 000 devices per cell) with their potential synchronous reactions to an event. In general, source traffic models (i.e., each device is modelled as an autonomous entity) provide higher precision and flexibility. However, their complexity grows quadratically with the number of devices. Aggregated traffic models, on the contrary, are far less precise but their complexity is mainly independent of the number of devices.

In this chapter, we present several modelling strategies, namely, (i) aggregated traffic, (ii) source traffic, and (iii) a hybrid approach. The three models are explained and compared through a common use-case. It allows both to illustrate the trade-off between accuracy and complexity and to guarantee the comparability of future studies by the deployment of common models.

*Keyword:*M2M, MTC, Markov Chain, traffic modelling, traffic volume.

1 Introduction

Machine-type Communication (MTC) or Machine-to- Machine Communication (M2M) is regarded as a form of data communication that does not necessarily require human interaction (ETSI, 2010). This type of communication will play an important role in the information and communications technology (ICT) by enabling the future Internet of Things (IoT) and is expected to experience a significant growth within the next decade(3GPP, 2012). Moreover, supporting such a massive number of heterogonous connected devices, many of them serving time-critical applications, is also an important part of 5G system requirement (Ericsson, 2013).

M2M communication services, in addition to conventional voice and Internet traffic, will be an integral part of the traffic transported by LTE/LTE-Advanced network. A very large number of devices can be attached to operators' network, where the number of MTC devices could be of orders of magnitude greater than the number of cellular phones. This calls for new mechanisms to handle such a large number of devices with a low signalling and processing overhead.

While the literature focuses on enabling wide M2M deployment in LTE/LTE-A networks, the current mobile M2M traffic is in most cases conveyed through legacy networks. According to market predictions (GSMA, 2013), connections over legacy networks will remain predominant worldwide in the following years, with significant growth of M2M connections on both 3G and 2G. Wide M2M application areas imply a variety of corresponding traffic patterns and QoS requirements, which in turn impose substantial challenge to operators. While modern networks are mainly designed for Human-type Communication (HTC) and mostly downlink-dominant and bursty, M2M traffic is of generally different properties, mostly uplink-dominant, often periodic, and persistent. With the large number of connected devices, the knowledge of particular traffic patterns becomes substantial. Signalling congestion, network performance degradation affecting both voice and data services, as well as M2M services, proved to represent the worst-case scenarios (Popović, et al., 2013). The deployment of a new M2M service raises several questions for an operator: what are the traffic characteristics, what is the predicted number of connected devices, what is their spatio-temporal distribution, and what are their QoS requirements in terms of delay constraints. The operators needs to assess whether new M2M services could jeopardize traditional human type communication, and can the network provide or guarantee the required latency (Shafiq, et al., 2012). Some system solutions proposed for LTE-A could be applicable to legacy networks, but would have a deep implication in the network, making them generally unfeasible or expensive.

Understanding the properties of MTC traffic is therefore considered as the key for designing and optimizing future networks and the respective QoS schemes with the goal of provisioning adequate M2M communication services without compromising any conventional HTC services such as data, voice, and video. In particular, the success of 3GPP Rel-11 and Rel-12 Evolved Packet System (EPS) and its evolution toward 5G systems depends on the effectiveness of its class-based network-initiated QoS control scheme and the corresponding support of both MTC and HTC traffics.

Conventional HTC traffic and MTC traffic have two major differences: (i) HTC traffic is heterogeneous whereas MTC traffic is highly homogeneous (all machines running the same application behave similarly) and, further, (ii) HTC is uncoordinated on small timescales (up to minutes), while MTC may be coordinated, namely, many machines react on global events in a synchronized fashion. Some typical properties of MTC traffic may encompass:

- Massive number of devices (i.e., machines, users)
- Few short packets to be transmitted per machine
- Low duty-cycle traffic patterns (i.e., long periods between two transmission bursts)
- Traffic patterns with small statistical variation produced by single devices
- Uplink-dominant traffic (i.e., uplink volume higher than downlink volume)

- Realtime and delay-tolerant data bursts triggered by the same application
- 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)
- Spatio-temporal dependent traffic trigger

Thus, well-known traffic models designed for HTC require adaptations for their application to MTC. A fundamental question is whether it is feasible to model the traffic of a large amount of autonomous machines individually. This approach is called source traffic modelling. It is in general more accurate than its counterpart, aggregated traffic modelling (i.e., treating the accumulated data from all MTC devices as a single stream). The aim of this chapter is to provide a thorough comparison of both approaches in the context of MTC.

This chapter is organized as follows. In Section 2, we present a generic traffic modelling methodology applicable to the majority of the application scenarios including MTC. In Section 3, a detailed description of different modelling strategies and a comparable analysis by applying the same traffic recorded from a fleet management in an operational cellular network is described. Finally, in Section 4, summary and concluding remarks are presented.

2 Generic methodology for traffic modelling

There are many approaches found in literature to create traffic models for network data traffic (Adas, 1997).

Figure 1 illustrates a generic traffic modelling methodology applicable to the majority of application scenarios including M2M communication. It defines the workflow to evaluate the performance of a system operating in the desired application scenario by extracting the traffic traces to build a precise traffic model and its respective key performance indicators as the performance evaluation metrics (LOLA, 2012).

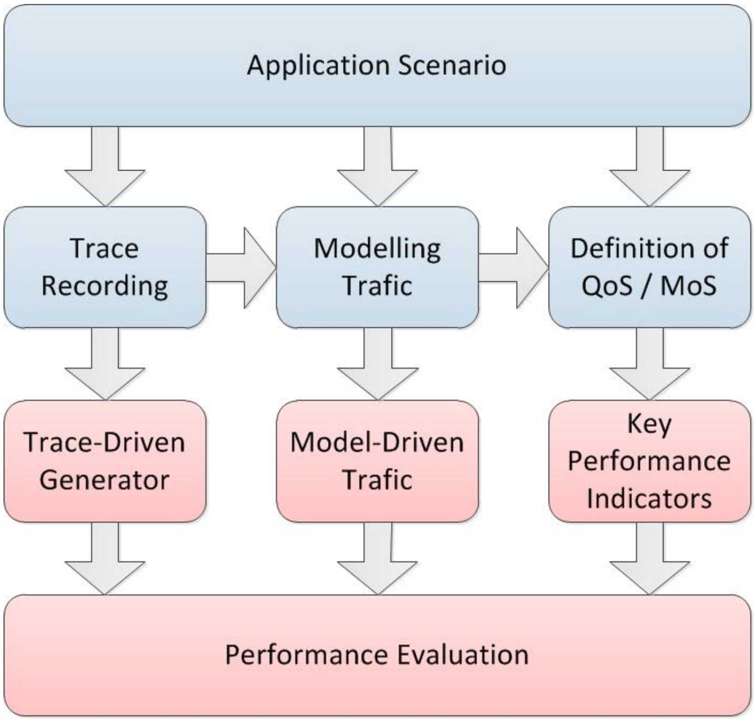

Figure 1: Generic traffic modelling methodology.

The first important step in the process of model selection is the definition of which parameters we want to model and how this should be done. Such parameters, e.g., packet loss or data rate, will then be analysed for statistical properties and an according model which generates similar statistical patterns is selected. Please note that the parameter of interest may be composed of multiples sub-parameters that have to be taken into account. If detailed statistical properties are of interest, then an adequate sample size must be ensured in order to guarantee the statistical significance of the result. In the following, we will describe these steps in more detail.

2.1 Trace recording

The traffic analyser tool generates a statistical dataset from a captured application scenario. It includes the following main building blocks: packet sniffer, parser, decoder, anonymiser, and analyser, as well as data storage, visualization, and management. Figure 2 illustrates the methodology for traffic tracing and analysis. The first step is the scenario description of an experiment. It defines the application/flow configuration, network setup, measurement links, and the required statistical datasets. Such a scenario is used to allow the experiment to be reproduced. Based on the scenario, a network is setup and synchronized to allow for measurements at different network interface/link with a common reference clock. Each measurement node needs to know the exact system time to provide time stamps on the measurements. If all links are symmetric and identical, nodes can accurately estimate their clock offset by using round-trip time measurements. When the network is asymmetric or time varying every node needs a time synchronization unit of its own. Typically, this is provided by Global Positioning System (GPS) providing Pulse-Per-Second (PPS) reference signal indicating the start of a new second with respect to the Coordinated Universal Time (UTC). The next step is to determine the measurement links associated with network interfaces, and define for each link the associated recording tools (e.g. Wireshark), packet formats (e.g. TCP, IP, MAC), and recording levels. The recording level could vary from full payload (extract all user information) to payload cut (extract header information only) and statistical information (extract traffic characteristics only). It has to be mentioned that such measurement is normally done locally.

When a packet is captured (i.e. sniffed, traced, and recorded) on the measurement links under traffic analysis, the packet parser will immediately read the recorded packet and decode the protocol header information. This procedure will iterate for all recorded packets until the duration of the experiment is reached. Relevant information may be extracted or derived from the decoded packet header during the pre-processing step (e.g. location information). Some additional control information associated to the packet (e.g. timestamp and/or processing time) may also be added to facilitate data storage and management for fast search of specific traces with particular attributes.

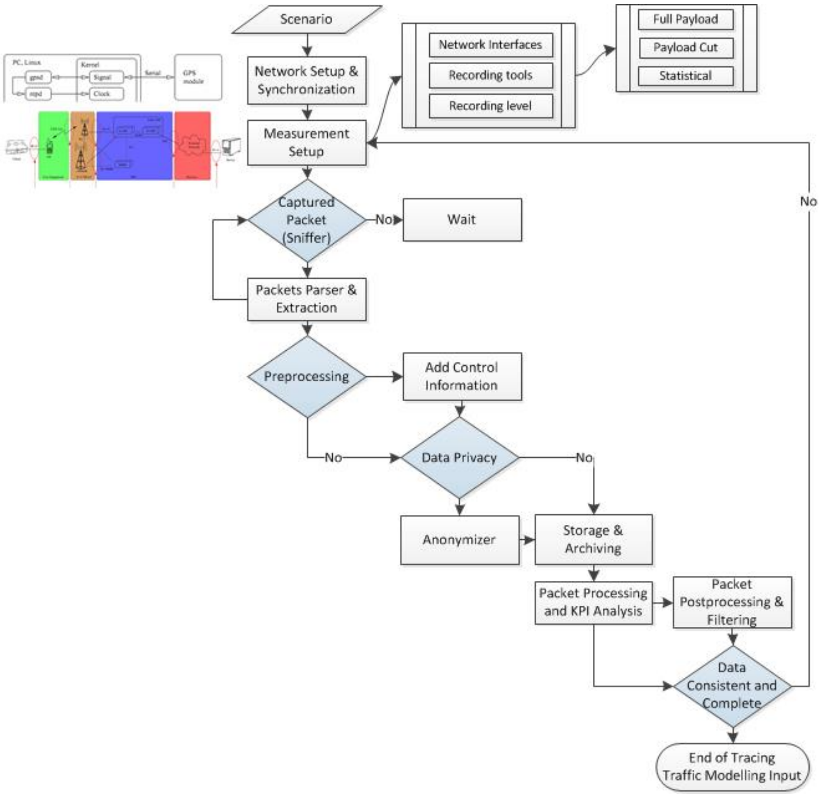

Figure 2: Packet tracing workflow.

The recorded traces are then stored and archived before being analysed. As stated above, the stored data could either be full payload, payload cut, and statistical information depending on the measurement setup for each network interface. If privacy issues apply, the traces must be anonymised before the storage and archiving. Note that the anonymisation should preserve user-packet association while hiding the user identity such that retrieving the user identify from anonymised trace becomes impossible. At this stage, the data is ready to be fully analysed in order to produce output of statistical datasets in terms of protocol operations, packet information, and KPIs. Finally, the experiment may be repeated several times to adjust the measurement links and recording tools such that the produced datasets become consistent and complete.

2.2 Traffic modelling

Figure 3 depicts the workflow of traffic modelling in order to reproduce traffic flows or data sources in a communication network. In general, there are two main approaches to reproduce traffic: traffic emulation or statistical modelling. The emulation of application traffic is possible if the application internals and its finite state machine are known or could be fully or partially derived through functional analysis (e.g. a sensor generating regularly a message within a constant time). However, if part of the application is too complex or its behaviour is unknown, a hybrid modelling approach combining traffic emulation and statistical modelling can be applied. For instance in FTP modelling, the FTP protocol part could be emulated as its behaviour is known while the requested file size and the request rate per used could be modelled based on a statistical distribution.

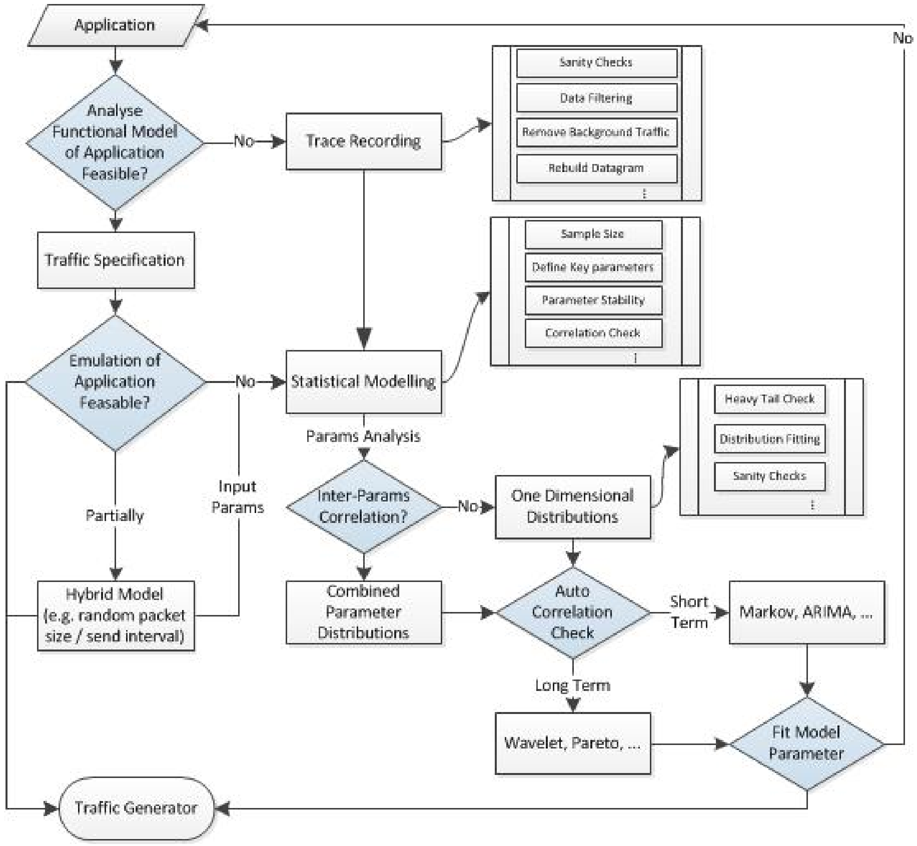

Figure 3: Traffic modelling workflow.

Statistical modelling, on the other hand, analyses application-specific recorded traffic traces by considering the underlying network under test, protocol interaction depending on at which OSI layer traffic is recorded, and other concurrent applications. Possible presence of inter-parameter correlation, e.g. between packet size and rate or data rate and delay, is checked by applying the correlation metric between parameters. Depending on the presence of correlation and required accuracy, the parameters can be fitted to one or multi-dimensional distributions.

Another important property of the traffic is the autocorrelation properties inside the time series of the data packets. In other words the question how fast the values of the times series fluctuate and if two consecutive values in the time series are dependent on the history and therefore on each other. This property is called short/long range dependency of a time series. A time series can be considered as short range dependent if the properties of the auto correlation function (ACF) have a limited lag and can be reproduced in an ARMA (Autoregressive Moving Average), Markov or similarly popular processes. Long range dependent time series on the other hand require specifically tailored models.

3 M2M traffic modelling

Traffic modelling means to design statistical processes such that they match the behaviour of physical quantities of measured data traffic (Adas, 1997). Traffic models are classified as source traffic models, e.g., video, data, voice, coming from one individual user and aggregated traffic models for backbone networks or the Internet.

The typical MTC use case includes numerous simple machines assigned to one server or medium, therefore the second modelling approach is more suitable, e.g., it is difficult to simulate thousands of discrete source traffic models in one scenario. A MTC scenario can be modelled as simple Poisson process; however, due to coordination (synchronizations) in MTC traffic, the respective arrival rate λ may be changing over time, $\lambda(t)$ (i.e., temporal modulation, (Heffes & Lucatoni, 1986)(Paiva, et al., 2011)). The more complex and individual the single MTC devices behave (e.g., video surveillance), the more questionable becomes the approach of modelling them as aggregated traffic assuming them to be all the same machine. The global data stream may exhibit high-order statistical properties which are difficult to capture (Casale, et al., 2010). We further expect this effect to be enhanced by the synchronization of sources. In such a case, traffic modelling in terms of source traffic is preferable. Source traffic models which can capture the coordinated nature of MTC traffic are available (Laner, et al., 2012). However, they are designed for a low amount of sources; thus, are too complex for MTC traffic (e.g., for N devices a N×N matrix-vector multiplication is required in each time slot).

For multiple access and capacity evaluations we do not need knowledge about the behaviour of a single node, e.g., simulate every link between a node and its base station. In this case aggregated traffic models such as homogeneous (Zhou, et al., 2012)(Ratasuk, et al., 2012), with constant arrival rate, or inhomogeneous (Paiva, et al., 2011), with time-varying arrival rate, Poisson processes, are a satisfactory description of reality and therefore largely deployed. Respective setups are defined by 3GPP (3GPP, 2012) and further discussed later. For the simulation of strongly scalable multiple access schemes in future networks (e.g., priority access, delay tolerant devices, QoS demands), mixed source traffic models have been adopted (Lien, et al., 2011)(Jou, et al., 2011)(Zhang, et al., 2011). In those studies, the case of synchronized MTC devices has not been considered or only for a limited number of MTC devices.

We observe a divergence between traffic models deployed within different studies. On one hand, higher accuracy requires source traffic models, and on the other hand, reduced complexity claims for aggregated traffic models. The following subsections present different traffic modelling strategies of both worlds.

3.1 Use case: Fleet Management

We consider one specific use case as an example for studying several modelling strategies introduced below. This use case is a “fleet management” scenario of 1000 trucks run by a transportation company in central Europe. It relies on measured M2M traffic from an operational 2G/3G network. This traffic has been captured at the Gn-interface, which is the interface with the highest data aggregation in the mobile core network. The resolution of this data set is on packet granularity. The data set does not feature payload but rather a set of parameters per packet. Those are, among others: the timestamp, the packet size, the direction, IP addresses, port numbers, pseudo identifiers of the sending devices and the Access Point Name (APN). The capturing period extended over one week, containing approximately 27 million packets originating from 1000 devices for the considered application have been observed. Due to the knowledge of the identifiers, single devices can be traced reliably over the whole observation period. The accumulated data rate was on average 1.89 kB/s, yielding a rate of 2 B/s/device. On average 4% of the MTC devices of the traced class are active. Further, they exhibit coordinated behaviour. We have observed 100 time instants, out of which more than 20% of all devices were trying to transmit data simultaneously. Consequently, this data set incorporates coordinated and uncoordinated phases and allows for (separate or joint) modelling of both cases.

3.2 Source traffic modelling

For every M2M application, there are four common basic stages of communication as follows:

1. Collection of data.
2. Transmission of selected data thought a communication network.
3. Assessment of the data.
4. Response to the available information.

As discussed subsequently, that yields different traffic patterns and associated states, which can then be modelled by specific processes.

3.2.1 M2M traffic states

Analysing the functionality of such M2M applications has revealed that MTC has three elementary traffic patterns (Nikaein, et al., 2013):

1. **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. Typically, 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).

2. **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 node acting as server to send commands to the device and control it remotely. ED is mainly 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 the distribution of a local warning message, e.g. in case of 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.
3. **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.

Real world applications are often a combination of the above-mentioned traffic types. Hence, using the three elementary types above for traffic modelling enables building models with high degree of complexity and accuracy. For example, a device may enter the power saving mode, trigger a PU pattern at regular intervals. Another example is that a PE is only triggered 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 PE may entail bursts of data traffic.

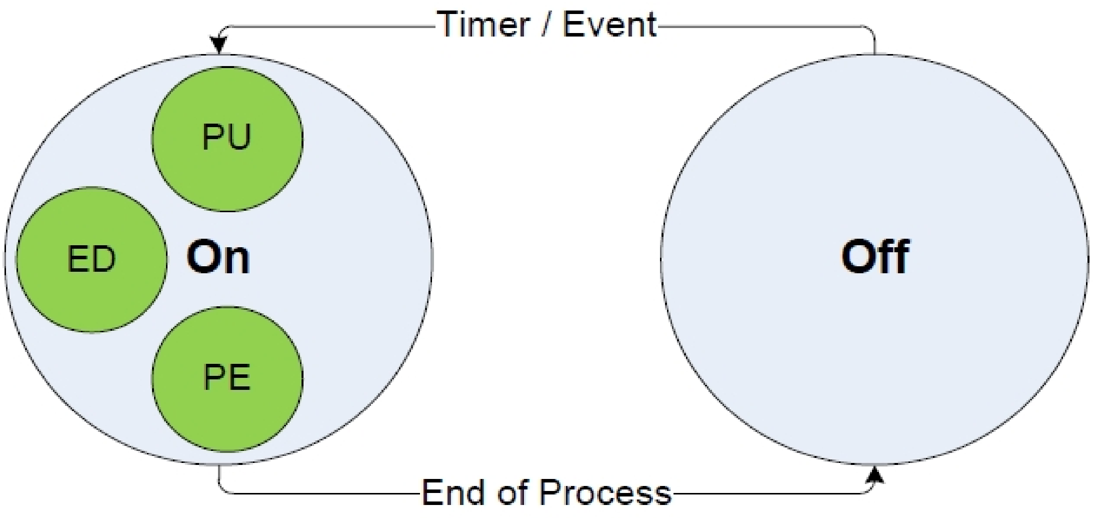

Figure 4: Elementary M2M traffic state structure.

For a convenient modelling of MTC traffic (by deploying the above described traffic types), we propose an On-Off structure, as depicted in Figure 4. 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, e.g., long periods of no data between phases of activity. 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. This enables the assignment of meaningful side-information to each state, such as respective QoS parameters. For example, the attribute “latency < 100ms” may be added to the state ED, in order to ensure fast forwarding of alarms.

3.2.2 Source modelling via Semi-Markov Models (SMM)

For modelling the data streams within single states *i*, we deploy renewal processes (Nelson., 1995, p. 254). They consist of a random packet inter-departure times (IDT) D_i and a random packet sizes (PS) Y_i.

Both random processes D_i and $Y_i$are identical and independent distributed (i.i.d.), with arbitrary marginal Probability Density Functions (PDFs) $f_{D,i}(t)$ and $f_{Y,i}(y)$. Two special cases are: periodic patterns, e.g., fixed inter-departure time, and Poisson processes, e.g., exponentially distributed IDT.

In order to model state transitions, we define a Semi-Markov Model (SMM) (Nelson., 1995, p. 352). A SMM defines transition probabilities p_{ij} between states, with p_{ii} = 0 transition probability to the current state. The transition probabilities are arranged in the transition probability matrix P. Further, a random sojourn time or holding time T_i is introduced per state, with arbitrary independent distribution $f_{T,i}(t)$. SMM models are advantageous for MTC modelling for several reasons: (i) they allow capturing a broad spectrum of traffic characteristics (S. Z. Yu, 2012), especially the almost vanishing data-rate, (ii) enable augmented modelling if side-information is available (e.g., the exact number of states are known) (Adas, 1997), and (iii) advanced fitting mechanisms are established (Yu, 2010), which allow for good fitting quality, even if nothing but raw traffic-measurements are given.

Table 1: Input parameters of the SMM approach.

State *s*	$f_{D,s}(t)$	$f_{Y,s}(y)$	$f_{T,s}(t)$	**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

The input parameters for the model are summarized in Table 1, where “$\cdot$” represents parameters to be fitted to a desired MTC traffic pattern and the completed items are state specific constants. $Deg(\cdot)$ 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.

An example use case is the modelling of a fleet management scenario, which is used as reference for the presented modelling techniques. The resulting model parameters for the outlined use case are the following (the range of distributions has been restricted to parametric distributions with at most two parameters for simplicity):

- $f_{T,\mathrm{OFF}}(t)$ = Deg(397s)
- $f_{Y,\mathrm{PU}}(t)$ = Deg(197B)
- $f_{Y,\mathrm{ED}}(t)$ = Deg(120B)
- $f_{D,\mathrm{PE}}(t)$ = Exp(6.65s)
- $f_{Y,\mathrm{PE}}(y)$ = Exp(43B)
- $f_{T,\mathrm{PE}}(t)$ = Exp(6907s)
- $P = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0.915 & 0 & 0 & 0 \\ 0.058 & 0 & 0 & 0 \\ 0.027 & 0 & 0 & 0 \end{pmatrix}$

3.3 Aggregated Traffic modelling

Because of its popularity we first provide an overview of the 3GPP model developed in (3GPP, 2012). The 3GPP model consists of two scenarios called Model 1 and Model 2. The first one treats uncoordinated events triggering data traffic and the second one coordinated events triggering data traffic. Both scenarios are defined by a distribution of packet arrivals (or, equivalently, access trials) over a given time period T, see Table 2. This is shown in Figure 5, where the Probability Density Functions (PDFs) of both distributions are depicted. The distributions $f(t)$ are both defined on the interval [0, 1], which has to be rescaled to the time interval [0, T] to yield $f_T(t)$. In order to simulate arrivals, it is sufficient to draw N samples from the given distribution and sort them in time, where N is the expected number of MTC devices, see Table 2. This number may reach up to 30 000, which is the maximum amount of smart meter devices expected to be served by one cell in a densely populated urban area (3GPP, 2012).

Table 2: Parameters of the 3GPP model.

Characteristics	Model 1	Model 2
Number of devices *N*	1000, 3000, 5000, 10000, 30000	
Distribution *f(t)* over *[0,1]*	uniform	beta(3,4)
Period *T*	60s	10s

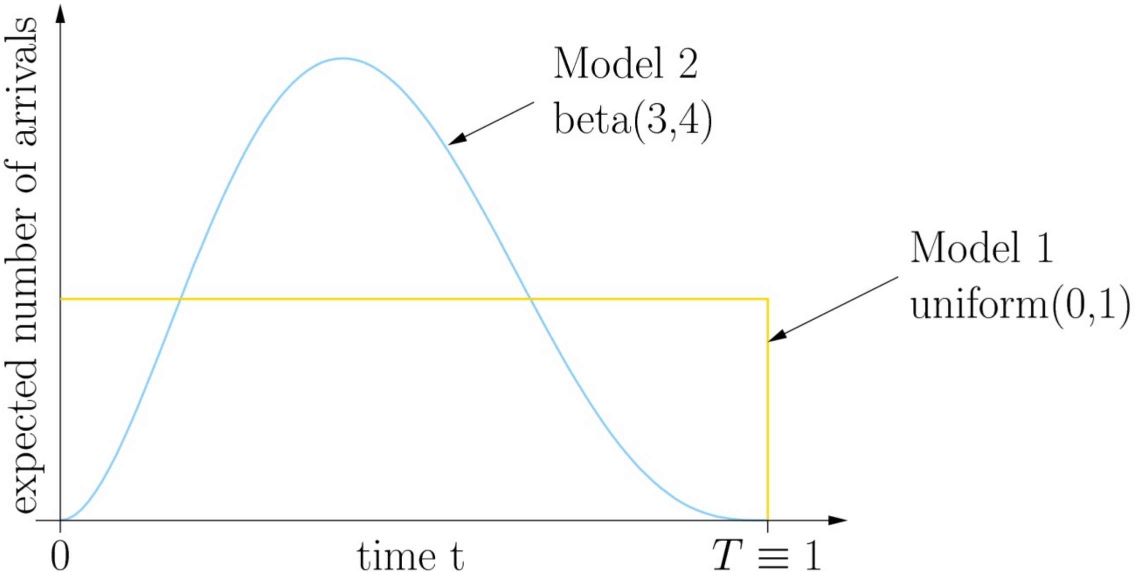

Figure 5: Access intensity of the 3GPP model.

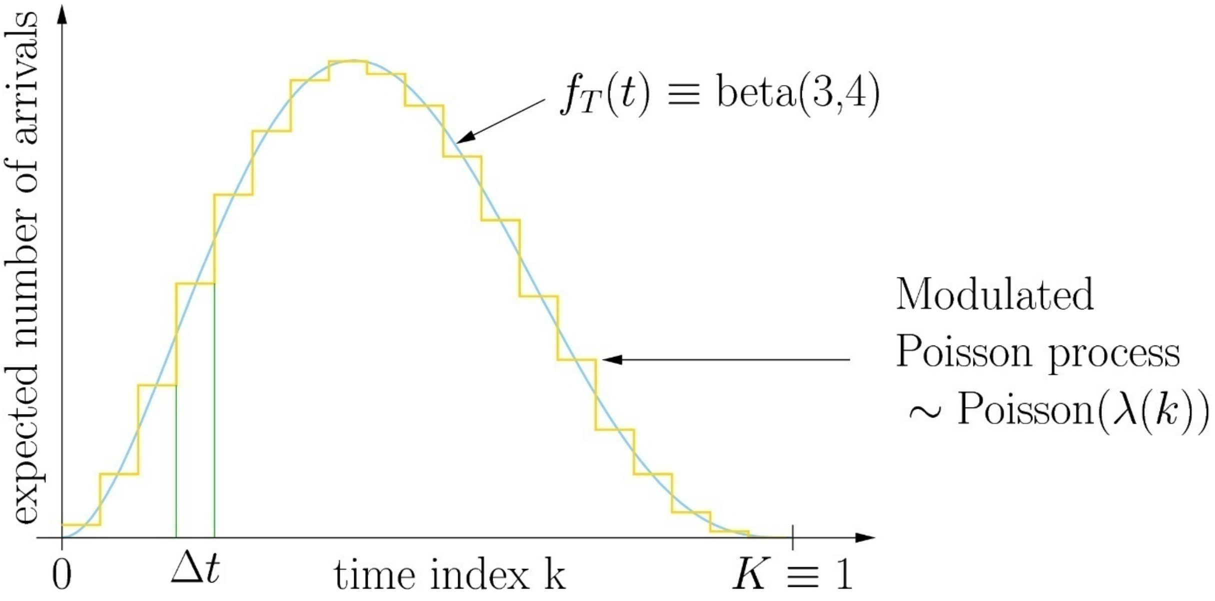

Figure 6: Synthesis of the 3GPP model from a modulated Poisson process.

In general it is undesirable to generate a full traffic pattern over T beforehand as this requires a large amount of memory on the simulation machine. In the present case this may not be an issue, however, as basic problems, such as undefined run length T or large amounts of generated data, may require a sequential drawing of samples. This issue is discussed in (Paiva, et al., 2011), where it is pointed out that the 3GPP model is equivalent to a modulated Poisson process. Thereby, the modulation is achieved by averaging the PDF of the arrival distribution $f_T(t)$ for time bins ΔT . This is depicted in Figure 6, where the mean arrival rate $\lambda(t)$ of a Poisson process is modulated in each time bin Δ_T by a beta distribution, cf. Table 2 as reference. For infinitesimal ΔT both curves coincide. Consequently, sequential sampling is performed by the generation of a Poisson distributed number of arrivals in each time bin ΔT with mean arrival rate $\lambda(t)$. In order to obtain an expected outcome of N samples within the period T (i.e., one sample per machine), the arrival rate has to be normalized according to $\lambda(t) = f_T(t) \cdot \Delta_T \cdot T \cdot N$. The two different sampling strategies are summarized in Figure 7.

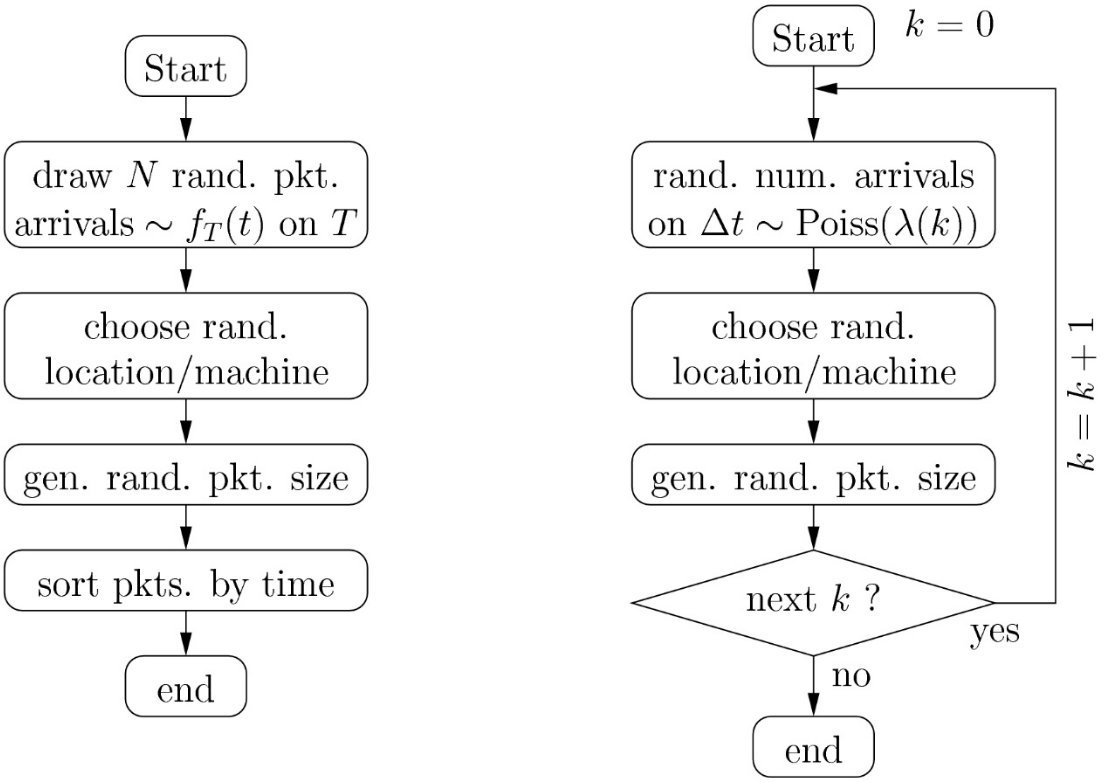
Start
draw N rand. pkt. arrivals $\sim f_T(t)$ on T
choose rand. location/machine
gen. rand. pkt. size
sort pkts. by time
end
Start
$k = 0$
rand. num. arrivals on $\Delta t \sim \text{Poiss}(\lambda(k))$
choose rand. location/machine
gen. rand. pkt. size
next k ?
yes
no
$k = k + 1$
end

Figure 7: Generation of samples from the 3GPP model. Left: non-sequential, right: sequential.

The 3GPP model however is not well suited for further M2M-specific requirements, such as:

(i) the amount of machines becomes small, so that a data source has to be associated with a fixed location,

(ii) multiple packets come from the same machine,

(iii) the synchronous traffic (Model 2) influences uncoordinated traffic (Model 1), and

(iv) the network has an influence on the traffic patterns.

Due to the small set of parameters, fitting of the 3GPP model to traced data is very simple. There are only two unknown parameters: (i) the number of MTC devices in the uncoordinated case and (ii) in the coordinated case. Applied on the use case of fleet management, presented above, both parameters are estimated to N=1000.

3.4 Source modelling for coordinated traffic via Markov Modulated Poisson Processes

To circumvent the limitations of the 3GPP model, a source modelling approach is adopted here. Each machine is thereby represented by a separate model. This approach is only feasible if a trade-off between mutual couplings among data sources (synchronization) and a tolerable complexity for large amounts of devices is found. Generic traffic models couple devices (i.e., multiple random processes) by bidirectional links between them. This is however too complex for the present purpose. Instead, one background process is proposed, modulating all MTC device entities.

Markov Modulated Poisson Processes (MMPPs) are presented in the following which model a single MTC device. Due to their simplicity, however, the operation of large amounts of device models in parallel is computationally feasible. Further, the coupling to the background process requires only low complexity.

3.4.1 Markov Modulated Poisson Processes – the basics

Markov models and Markov modulated Poisson processes are commonly deployed in traffic modelling and queuing theory. They allow for analytically tractable results for many use cases (Heffes & Lucatoni, 1986)(Nelson., 1995). MMPP models consist of a Poisson process modulated by the rate $\lambda_i[k]$, which is determined by the state of a Markov chain $s_n[k]$. Thereby k denotes the time index, obtained by migrating from continuous time t to discrete time by $k = \frac{t}{\Delta_T}$, where Δ_T denotes an arbitrary but constant time interval, constituting the “heartbeat” of the system. Further i=1...I denotes the index of Markov state and n=1...N the index of the Machine-type device. The principle of a modulated Poisson process is depicted in Figure 8, where $p_{i,j}$ are the transition probabilities between the states of the chain. In the present source modelling approach each MTC device n out of N is represented by a separate Markov chain and a corresponding Poisson process. The state transition probabilities form the state transition matrix P and the state probabilities π_i form the state probability vector π.

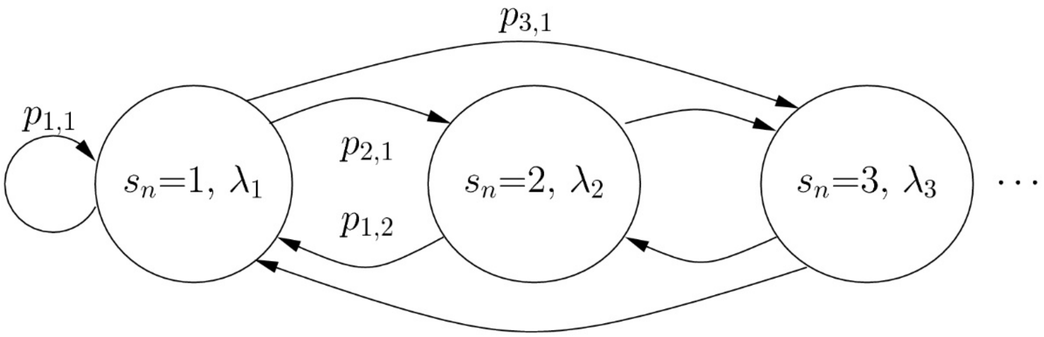

Figure 8: Markov chain driving the MMPP model.

In the stationary case both parameters are related by the balance equation $\pi = \pi P$. Hence, π is an eigenvector of P to the eigenvalue of 1. Further, the overall perceived rate of the MMPP is $\lambda_g = \sum_{i=1}^{I} \lambda_i \pi_i$, where I is the total number of states. A basic example for an MTC device modelled by a MMPP is a two state MMPP with the first state representing regular operation and the second representing the sending of an alarm. This is in analogy to the 3GPPmodel (see above), where two models (Model 1 and Model 2) are capturing coordinated and uncoordinated behavior.

3.4.2 Coupling multiple MMPP processes

The state transition matrix P has to be determined such that each device model resides a prescribed amount of time in each state. From the perspective of a single device this is straight forward; from a global perspective, the devices transit from the regular to the alarm state in a strongly correlated manner in both time and space. To imitate this behaviour for multiple MMPP models, some coupling is required.

In the context of pattern recognition Coupled Markov chains are well-known (Brand, et al., 1997)(Brand, 1997). They are realized as multiple chains which mutually influence their transition probability matrices $P_n(t)$. In terms of discrete time they are given by $P_n[k]$, corresponding to the notation deployed in the following. The matrices are influenced by the respective multiplication of weighting factors, which depend on the past states $s_m[k-1]$ of neighbouring chains m.

We only consider a unidirectional influence from a background process (master) $\Theta(t)$ to the MTC device MMPP models in the presented framework; an example of said background process is a fire in a factory triggering several fire sensors to response or a traffic jam on the road triggering telemetry sensors in trucks to transmit in a correlated fashion. This approach is named Coupled Markov Modulated Poisson Processes (CMMPP). Again, we perform the transition from continuous to discrete time by sampling at time instant k: $\theta[k] = \Theta(k\ \Delta_T)$. The separate tuning of each of the weighting parameters for each machine is avoided by defining the following framework (Laner, et al., 2013):

Let there be two transition matrices P_C and P_U globally valid for all N MMPP models, representing the strictly coordinated and strictly uncoordinated behaviour, and a background process Θ(t) producing samples θ[k] within the interval [0, 1]. Further, a parameter $\delta_n \in [0, 1]$, constant over time, is associated to each MTC device n yielding $\theta_n[k] = \delta_n \cdot \theta[k]$. Then the state transition matrix $P_n[k]$ can be calculated for machine n at time t according to the following expression:

$$P_n[k] = \theta_n[k] \cdot P_C + (1-\theta_n[k]) \cdot P_U \qquad \text{Equation 1}$$

This convex combination of both transition matrices yields again a valid transition matrix. The matrices P_C and P_U are transition matrices for the case of perfectly coordinated device behaviour and uncoordinated device behaviour, respectively. The parameter $\delta_n$corresponds to a measure of closeness (distance) to the epicentre (point in space, on which the expected coordination is maximum). The closer $\theta_n[k]$ to zero, the more uncoordinated the respective machines behaves; the closer $\theta_n[k]$ to one, the stronger the coordination. The background process Θ(t) is allowed to have an infinite number of states, yielding θ[k] to be a continuous process.

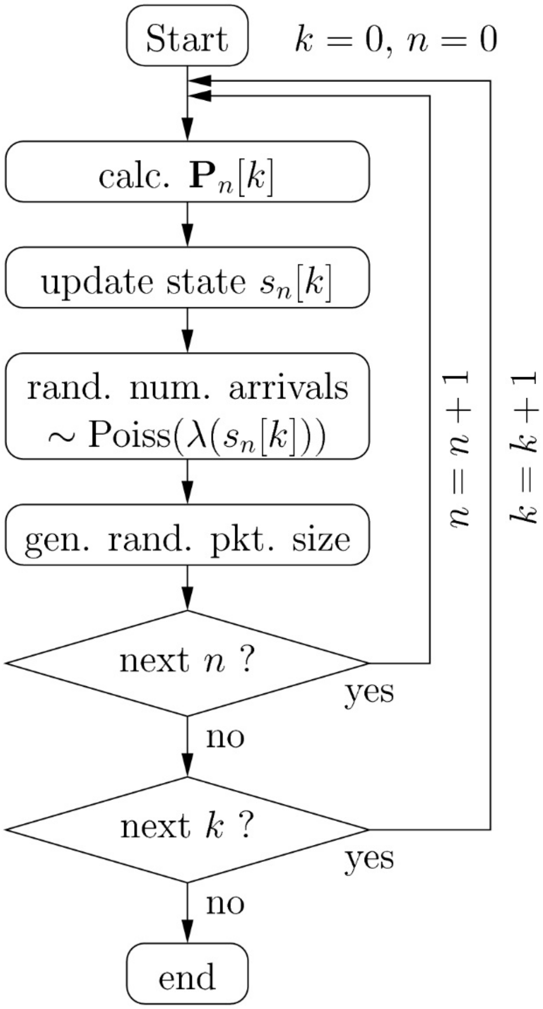

Figure 9: Generation of samples from the CMMPP model; the indices n and k stand for the device and time instant, respectively.

The synthetic generation of data traffic according to the CMMPP model is described in the flow diagram in Figure 9. Two nested loops are required, both for devices n and time index k, respectively. The transition matrix $P_n[k]$ is calculated anew for any iteration according to Equation 1. From a complexity perspective this is feasible since the number of states is usually low and the required convex combination can be computed efficiently. The random state update from $s_n[k-1]$ to $s_n[k]$ is performed afterwards. Finally, a number of arrivals and packet sizes are generated according to the current state $s_n[k]$.

The above-mentioned fleet management application for example yields the following model parameters:

- Δ_T = 1s
- λ_0 = 0.15 B/s
- λ_1 = 6.5 B/s
- λ_2 = 24.7 B/s
- θ[k] = 1 @ k=0, 0 otherwise
- δ_n = 1
- $P_U = \begin{pmatrix} 1-6.75\times10^{-5} & 1.47\times10^{-4} & 0.39 \\ 6.75\times10^{-5} & 1-1.47\times10^{-4} & 0 \\ 0 & 0 & 0.61 \end{pmatrix}$
- $P_C = \begin{pmatrix} 0.66 & 0 & 0 \\ 0 & 0.66 & 0 \\ 0.33 & 0.33 & 1 \end{pmatrix}$

4 Model fitting from recorded traffic

The question remains on how to obtain the parameters for above models. Typically, the fitting procedures of the three modelling approaches outlined above relies on the time series of the data rate produced by actual MTC devices. The properties of the traffic streams at higher granularity (e.g., packet level) are not of particular interest.

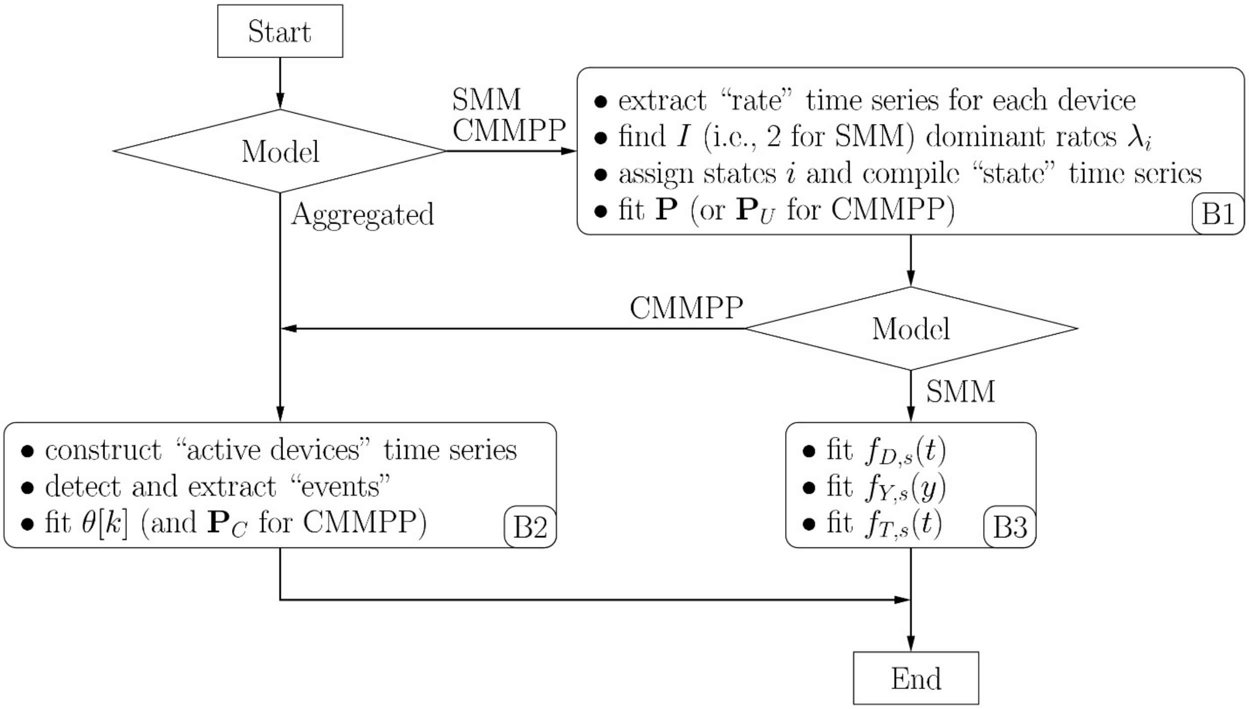
Start
Model
SMM
CMMPP
Aggregated
• extract "rate" time series for each device
• find I (i.e., 2 for SMM) dominant rates λ_i
• assign states i and compile "state" time series
• fit $\mathbf{P}$ (or $\mathbf{P}_U$ for CMMPP)
B1
CMMPP
Model
SMM
• construct "active devices" time series
• detect and extract "events"
• fit $\theta[k]$ (and $\mathbf{P}_C$ for CMMPP)
B2
• fit $f_{D,s}(t)$
• fit $f_{Y,s}(y)$
• fit $f_{T,s}(t)$
B3
End

Figure 10: Fitting of traffic models to recorded MTC traffic.

The detailed fitting procedures for the modelling approaches are outlined in the flow diagram in Figure 10. All procedures are covered by a subset of the three main building blocks, B1, B2, and B3. The SMM model requires blocks B1 and B3, the aggregated traffic model requires B2 and the CMMPP model requires B1 and B2. The functionality of the building blocks is the following:

- B1: a Markov model for the behaviour of an individual device is derived
- B2: an aggregated traffic model is built for the global data rate
- B3: each state obtained in B1 is described as general renewal processes

The three building blocks are presented in detail in the following subsections.

4.1.1 B1: Modelling individual devices as Markov chains

This block is required to fit transition probability matrices to traced traffic. In the case of SMM modelling it is the matrix P of the embedded Markov chain; in the case of CMMPP modelling this is the matrix P_U.

Further, data rates λ_i associated to the states i are obtained, either for direct use in the model (i.e., CMMPP) or for further modelling steps (i.e., SMM, B3). For the aggregated traffic model (i.e., 3GPP model) this block is not required, since the model has no notion of individual machines.

We first extract the time series of the data rate of individual MTC devices. A temporal resolution of one second is proposed, in order to capture all relevant effects. This resolution may however lead to strong variations in the data rate due to sparse arrivals of packets. In order to smooth these variations, a sliding average ought to be used. In the present case, we deployed a sliding average of 30 min and a cosine-shaped window. Figure 11 shows a corresponding time series of an MTC device over the duration of 4 days. The device is corresponding to the fleet management application described in Section 3.1. It is clearly visible that there are roughly two main rates used for communication: (i) 0.5 B/s and (ii) 7 B/s.

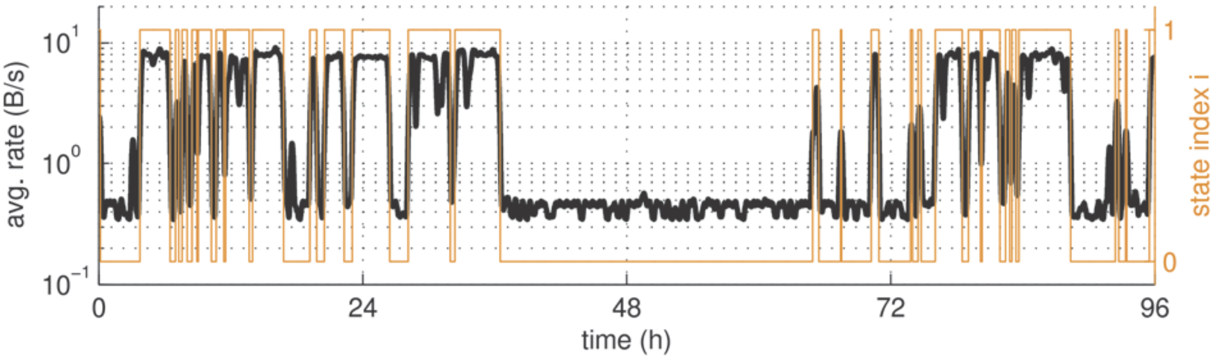

Figure 11: Rate time series of a single MTC device.

From the obtained samples of the data rate we derive the histogram (empirical distribution). This distribution should ideally consist of few regions of high density (common data rates), which constitute the Markov states of the final model. It has to be decided upon how many states shall be deployed for modelling. This is equivalent to fixing the number I for CMMPP models. For SMM models this number is always I=2, since only two different rates can be achieved by the respective approach. This is due to the restrictions imposed by definition on the four individual states, see Table 1. Namely, the states OFF, PU and ED do not exhibit the same flexibility as the PE state.

When I is fixed, one has to place boundaries between the rate regions associated to specific states. For instance in the case of SMM models, a single threshold rate is defined, yielding all instances with data rate above that threshold to be associated to state PE and all others to a combination of the states OFF, PU and ED. This task requires either human efforts or a set of elaborate rules on how to place these boundaries in an opportunistic or deep-learning manner. We achieved a satisfactory performance using the k-means algorithm on logarithmically compressed rate samples. The rates λ_i, the output of this modelling block, can be computed by averaging over all rates within one region (state i).

A time series of state sojourns can now be constructed, from which the state transition probabilities (i.e., P for SMM models or P_U for CMMPP models) are directly obtained. For SMM models the state transitions to other states are of primary interest, for CMMPP models transitions to the state itself have to be further considered. For example, Figure 11 shows a respective time series where the number of states was chosen to I=2 due to the two dominant data rates.

To evaluate one individual device, it is enough to find values for P and λ_i from a single device time series. However, to obtain statistically significant modelling parameters, several devices should be considered. Based on the obtained modelling parameters, one then has to decide whether (i) an individual model is fitted to each device and afterwards combined to an average model, or (ii) the data is combined to an average device, to which a single model is fitted.

4.1.2 B2: Modelling aggregated traffic

This building block is required to fit numerical values to the aggregated traffic stream, for both regular operation and event-based operation scenarios. For the aggregated traffic model (i.e., 3GPP model) the unknown parameter is only the number of devices N (for both kinds of operational scenarios); however, the suggested distributions (e.g., beta distribution) could also be tuned to better fit the traced traffic (e.g., adjusting the parameters of the beta distribution). For the CMMPP model, the unknown parameters are the background process $\theta[k]$, as well as the transition probability matrix P_C for the coordinated case. For the SMM model this block is not required; nevertheless, results obtained here can be used to refine the probability of occurrence of an event or, equivalently, the transition probabilities to the PE state.

For modelling aggregated traffic, it is required to construct the time series of the global number of active devices (i.e., N). Thereby, a resolution of one second is chosen, which is not critical for the present purpose, but can be adapted to specific use cases. Further, smoothing is required to reliably detect events (defined below). For this purpose we deploy the same moving average filter as outlined in the section above (i.e., cosine window with 30 min duration).

Comparing the original and the smoothed time series of the number of active devices allows for the detection of events. Thereby, the term event has to be defined by a set of rules. For the use case outlined in Section 3.1, for example, we define the occurrence of an event when:

- The number of active users in one second is ten times higher compared to the sliding average (normal operation).
- No such change in the number of active users has occurred within the last 30 sec.

With this policy, events are reliably detected and the frequency of occurrence of events can be calculated (useful for the refinement of transition probabilities of SMM models).

In the context of the 3GPP and CMMPP models, the simulation of coordinated traffic as a single isolated event is targeted. For this purpose it is convenient to define a “typical” event by averaging over all detected events. As per Figure 12, the typical event for the use case is depicted through a black solid line, where 88 events have been detected. From this curve, the average number of active users (e.g., 21.4) has been subtracted in order to avoid any bias (which is modelled separately and in parallel). During a typical event, 315 devices become active within the same second (peak second). The most extensive event during the tracing period triggered roughly 500 devices simultaneously, the lightest only 180 MTC devices. Comparing these numbers to the total device population (i.e., 961) shows that an event triggers 33% of the devices; whereas only 2.2% of the devices are active per second during uncoordinated operation.

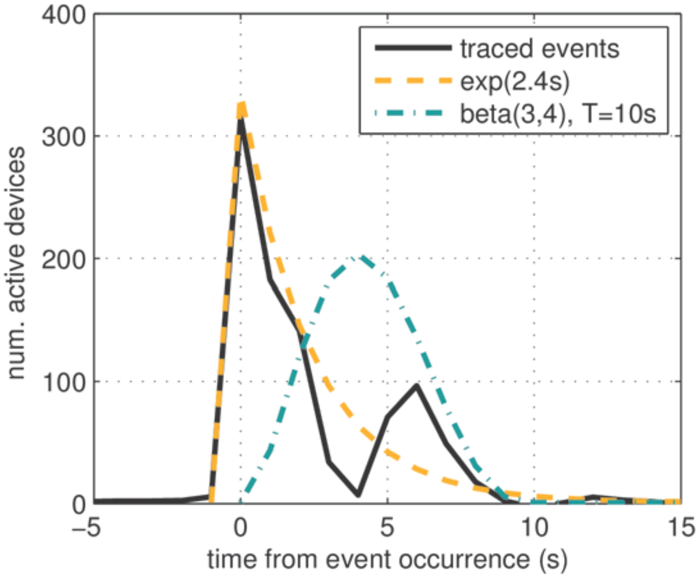

Figure 12: "Typical" event for the use case (see Section 3.1); modelled by the original 3GPP model and an exponential distribution with μ=2.4s.

Fitting the 3GPP model to the typical event requires to sum over all communication activities during the duration of an event. The typical event contains 976 activities, yielding a number of N = 1 000 MTC devices the best fit of the 3GPP model in the coordinated case. Note, however, that the 3GPP model assumes all activities to originate from different devices. This is not the case for the fleet management scenario, where each active device has on average 3 activities per event. In uncoordinated operation there are on average 21.4 activities per second, amounting to 1 284 activities per minute. The closest value adhering to the 3GPP model is here again N = 1 000 devices.

Figure 12 shows moderate accordance of the 3GPP model and the modelled use case (especially regarding the duration of the event); note that only one parameter (i.e., N) has been fitted. This confirms that the beta(3,4) distribution and the duration T=10s are decent choices for coordinated M2M behaviour for the given use case. A point of criticism is that the model assumes the event to start smoothly and reach its peak after some seconds. For the traced event this is not the case. We observe that the event starts within one second and ends smoothly. This could be modelled more accurately by an exponential distribution, as shown in Figure 12. Thereby the average duration is determined to µ=2.4s.

The determination of the parameters $\theta[k]$ and P_C of the CMMPP model allow for a more accurate representation of reality. An additional state j is introduced to the Markov chains of single devices, which represents coordinated operation; whereby the associated mean rate λ_j must be determined. In the case of the presented fleet management application, the rate calculates to λ_j = 24.7 B/s. It is reasonable to assume the background process $\theta[k]$ to resemble a unit impulse function (i.e., one at index zero, zero otherwise), since all devices involved into an event exhibit activities within the first second. Thereby $\theta[k] = 0$ corresponds to uncoordinated operation and $\theta[k] = 1$ to the coordinated case. Accordingly, P_C is constructed such that 33% of the devices fall into state j (event), the rest remains in the prior state (uncoordinated operation). Finally, the transition probabilities in P_U, which cause devices to return from state j, must be adjusted such that the exponential decay shown in Figure 12 is matched. For example, assuming a time slot duration of 1 s, this can be achieved by setting the probability of remaining in state j to$p_{U,jj} = 0.61$.

4.1.3 B3: Modelling single Markov states

This building block is required only for the SMM approach. The desired output parameters are the distributions for packet size, packet inter-arrival times and sojourn times associated to each of the four states. Thereby, several of these distributions are fixed a priory, such that only few have to be derived from traced data, see Table 1.

For this purpose the time series of state sojourns for individual devices (obtained from the building block B1) can directly be reused. Those series exhibit two states: (i) State 0 with low data rate λ_0 and (ii) State 1 with high data rate λ_1. As already mentioned, State 0 is matched by the combination of states OFF, PU and ED; State 1 is resembled by PE. Accordingly:

- $f_{T,\mathrm{OFF}}(t)$ is set to the constant value of the average inter-packet time in State 0
- $f_{Y,\mathrm{PU}}(y)$ is set to the distribution of the packet size in State 0
- $f_{Y,\mathrm{ED}}(y)$ is set to the distribution of the packet size during events
- $f_{D,\mathrm{PE}}(t)$ is set to the distribution of the packet inter-arrival time in State 1
- $f_{Y,\mathrm{PE}}(y)$ is set to the distribution of the packet size in State 1
- $f_{T,s}(t)$ is set to the distribution of the sojourn time in State 1

The extraction and fitting of these parameters is straight forward. We encourage the usage of simple distributions (e.g., unimodal distributions) and simple fitting procedures (e.g., method of moments). By fitting the fleet management scenario, for example, only degenerate (i.e., constant) and exponential distributions are adopted. Note that all introduced distributions are restricted to strictly positive support.

5 Conclusions

Recent studies have confirmed the impact of M2M traffic on network performance and accessibility through a measurement analysis in operational networks (Popović, et al., 2013). Thus, supporting the co-existence of M2M uplink-dominant traffic with conventional downlink-dominant user traffic without any service degradation coupled with the massive number of connected devices to the cellular infrastructure is of primary requirement for the next generation wireless network design.

The study of MTC application cases showed that it is possible to dissect the traffic states of an M2M node into three generic states, namely event driven, periodic update and payload exchange. These three states can be implemented as a source traffic model using an SMM. The source traffic modelling considers MTC application-specific traffic as a single stream. The parameters for such a model were extracted for different use cases leading to source traffic models for each node. The prediction for M2M nodes per cell assume numbers of up to 10.000 devices at a time. Traffic patterns of such a high number of users can only be modelled working with aggregated traffic streams in a feasible way. The aggregated approach has a lower complexity and is capable of capturing the coordinated traffic. A hybrid solution can be achieved using a CMMPP approach where the state of the nodes is coupled via a second Markov process. This allows a low complexity modelling of the traffic without neglecting the feature of correlated event-driven traffic.

Table 3 presents a comparative summary of the 3GPP aggregated modelling approach with the SMM and CMMPP source modelling approaches.

Table 3: Input parameters of the SMM approach.

Metric	Aggregated	SMM	CMMPP

Modeling device granularity		✓	✓
Modeling coordinated devices	✓		✓
Tempo-spatial coordination			✓
Modeling packet		✓	
Modeling data rate	✓	✓	✓
Random run time feasible		✓	✓
Device location		✓	✓
Modeling QoS constraint		✓	✓
Coupling of traffic states			✓
Complexity (N MTC devices)	*O(1)*	*O(N)*	*O(N)*

To provide comparable results, the same-recorded network traffic produced by a fleet management usecase has been applied to all the modelling strategies. The measurement results show that in fact the devices trigger in a correlated fashion in the time domain. While basic models like the one proposed by 3GPP fail to produce the same activity peak in the number of active users, the CMMPP model proved to be accurate under these conditions reproducing the peak of active users with a linear growing modelling complexity.

6 Bibliography

3GPP, 2012. *Service Requirements for Machine-Type communication, TR 22.368,* s.l.: s.n.

3GPP, 2012. *Study on RAN Improvements for Machine-type communications, Technical report, TR 37.868,* s.l.: s.n.

Adas, A., 1997. Traffic Models in Broadband Networks. *IEEE Communication Magazine.*

Brand, M., 1997. *Coupled hidden Markov models for modeling interacting processes,* s.l.: MIT Technical Report.

Ericsson, 2013. *5G Radio Access: Research and Vision,* s.l.: Ericsson.

ETSI, 2010. *-to-Machine communications (M2M); M2M service requirements,* s.l.: ETSI TS 102 689 .

G. Casale, E. Z. Z. a. E. S., 2010. Trace data characterization and fitting for Markov modeling. *Elsevier Performance Evaluation.*

Geng Wu, T. S. J. K. H. N., 2011. M2M: From mobile to embedded internet. *IEEE Communication Magazine.*

GSMA, 2013. *The Mobile Economy 2013,* s.l.: GSMA.

H. Heffes, D. L., 1986. A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance. *IEEE Journal on Selected Areas in Communications.*

IEEE802.16p, 2012. *Machine to Machine (M2M) System Requirements Document (SRD),* s.l.: IEEE.

K. Zhou, e. a., 2012. *Contention Based Access for Machine-Type Communications over LTE.* s.l., Vehicular Technology Conference.

LOLA, 2012. *D3.5 Traffic Models for M2M and Online Gaming Network Traffic,* s.l.: s.n.

M. Brand, N. O. P., 1997. *Coupled hidden Markov models for complex action recognition.* s.l., IEEE Computer Vision and Pattern Recognition.

M. Laner, e. a., 2012. *Users in Cells: a Data Traffic Analysis.* s.l., IEEE WCNC.

M. Laner, P. S. a. M. R., 2012. *Modeling Randomness in Network Traffic.* s.l., ACM Sigmetrics.

M. Laner, P. S. N. N. a. M. R., 2013. *Traffic Models for Machine Type Communications.* s.l., IEEE ISWCS.

M. Popović, e. a., 2013. Evaluation of the UTRAN (HSPA) performance in different configurations in the presence of M2M and online gaming traffic. *Transactions on Emerging Telecommunications Technologies.*

Morioka, Y., 2011. *LTE for Mobile Consumer Devices,* s.l.: ETSI Workshop on Machine to Machine Standardization.

N. Nikaein, e. a., 2013. *Simple Traffic Modeling Framework for Machine Type Communication.* s.l., IEEE ISWCS.

Nelson., R., 1995. *Probability, Stochastic Processes, and Queueing Theory.* s.l.:Springer.

R. C. D. Paiva, R. D. V. a. M. S., 2011. *Random access capacity evaluation with synchronized MTC users over wireless networks.* s.l., s.n.

R. Ratasuk, J. T. a. A. G., 2012. *Coverage and Capacity Analysis for Machine Type Communications in LTE.* s.l., Vehicular Technology Conference.

S. Lien, K. C. a. Y. L., 2011. Toward Ubiquitous Massive Accesses in 3GPP Machine-to-Machine Communications. *IEEE Communication MAgazine.*

S. Z. Yu, Z. L. M. S. S. C. X. L. Z., 2012. *A Hidden Semi-Markov Model for Web Workload Self-similarity.* s.l., IEEE Proceedings of the Performance, Computing, and Communications Conference.

Shafiq, M. Z., 2012. *A First Look at Cellular Machine-to-Machine Trafffic – Large Scale Measurements and Characterization.* s.l., 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems.

Y. Jou, e. a., 2011. *M2M over CDMA2000 1x Case Studies.* s.l., IEEE WCNC.

Y. Zhang, e. a., 2011. Home M2M Networks: Architectures, Standards, and QoS Improvements. *IEEE Communication Magazine.*

Yu, S. Z., 2010. Hidden semi-Markov models. *Elsevier Journal of Artificial Intelligence.*