Université de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par **l'Institut Supérieur de l'Aéronautique et de l'Espace**
Spécialité : Réseaux et Télécommunications

Présentée et soutenue par **Olivier GRÉMILLET**
le 11 décembre 2007

Analyse des modèles de trafic agrégé
pour les télécommunications multiservices par satellite

JURY

M. Cédric Baudoin, examinateur
M. Michel Bousquet, directeur de thèse
M. Axel Jahn, co-directeur de thèse
M. Otto Koudelka, rapporteur
M. Pascal Lorenz, rapporteur
M. José Radzik, examinateur

École doctorale : **Mathématiques, informatique et télécommunications de Toulouse**

Unités de recherche : **Équipe d'accueil SUPAERO SCN**
DLR Munich

Directeur de thèse : **M. Michel Bousquet**
Co-directeur de thèse : **M. Axel Jahn**

Merci - Danke - Thanks

Thanks

This thesis resulted from my work as a researcher at the Institute of Communication and Navigation of the German Aerospace Center (DLR) in Oberpfaffenhofen headed by Prof. F. Kuehne and Prof. C. Guenther. In particular, it was a pleasure to be with the department of Digital Networks of Prof. Dr. Erich Lutz: my thanks to him and all the colleagues which acquittance I made during the last six years. This work would not have been possible without this support. It was a pleasure to work there in the group of Dr. Ing. Axel Jahn and later in the group of Dr. Ing. Hermann Bischl. Meeting people from different countries was a constant pleasure there, since we had people from Germany (or Bavaria?), Slovenia, Italy, Spain (or Catalonia?), Russia, Australia, Austria.. and France.

I am very grateful to my supervisors Prof. Michel Bousquet that gave me the possibility to undertake this work and to Dr. Ing. Axel Jahn for his constant enthusiastic support and guidance. Furthermore I want to thank Prof. José Radzik for his support and precious comments on the content. To show their readiness to help, I will mention the different points where we have met: Montreal, Toulouse, Yokohama, Siena, Munich, Herrsching.

Thanks to the rapporteurs who accepted to review this work and thanks to the members of the jury for their participation to the PhD defense. Their analysis of this work has helped to increase its quality and helped to write a better organised and more complete final report answering their questions.

I want also to thank all the friends who supported me during this time in Germany and convinced me that one day, this work will reach completion. First, the DLR office mates (Oliver, Harald, Filomena, Matteo) who had to suffer *del francese qui canta* and of the constant chaos in the room. Then, the french colleagues (Nicolas, Vincent, Boubeker, SESS group) for escaping the DLR casino and organizing many mountains trips and together with the French connection in Munich (Sandra, Jessica, Caroline, Jean Philippe, Nathalie) for different outings. Then, also the astrophysicists from the MPA (Brice, Hugues, Lidia, Jonathan) for the nice time mostly spend in the Bierstube at Olympiadorf. Finally I thank the singing friends from Groebenzell and from the Münchner Domchor for the nice hours spent together. I thank also the italian (Filo and Guiseppe, Matteo, Alarico, Sergio, Sandro, Giorgio, Alessio,...) and spanish (Nuria and Cristina) groups of DLR (or shall I say mafia ?) for some very funny evenings.

Olivier Grémillet

Remerciements

En addition aux personnes mentionnées ci-dessus, je remercie mes parents et grands-parents de m'avoir soutenu pendant toute la durée de ce travail et de m'avoir donné le goût de l'effort. Je leur dédie ce travail.

Enfin, je remercie Florence qui, bien que m'ayant rejoint après la fin de ce travail, m'a soutenu pour le terminer complétement.

Olivier

Foreword - Avant propos

Foreword

This thesis resulted from my work as researcher at the German Center of Aeronautic and Space (DLR - Deutsches Zentrum für Luft- und Raumfahrt). I was a member of the institute of Communication and Navigation (KN), department of Digital Networks (headed by Prof. Dr. Erich Lutz) from 01/10/2000 and 30/11/2006 in Oberpfaffenhofen, near Munich, Germany. During these six years, in addition to the daily work devoted to the projects, I was given the opportunity to conduct a PhD in the framework of an agreement between DLR and Prof. Michel Bousquet. This gave a true dynamic to my scientific work at DLR. My activities were centered around traffic engineering and network dimensioning for satellite systems. Spread over these years, my work took seminal inputs from the projects *Euroskyway* and *WirelessCabin*. This thesis has been defended in December 2007 in Toulouse.

The topic of the thesis is the elaboration of an aggregated model for the hierarchical description of traffic generation in satellite networks. The organization of this document can be summarized as as follows. After a first chapter that introduces the topic and states the problem, three theoretical chapters (chapters 2, 3 and 4) introduces the hierarchical model of traffic, the models for traffic, and the scenario for a study case. Chapters 5, 6 and 7 are devoted to the presentation of the findings showing the interest of aggregate models. Finally, chapter 8 provides a summary, draws the conclusions and gives out an outlook for future work. French speaking readers will found in addition a *30 pages* overview in the chapter 9.

Avant propos

Cette thèse est le fruit de mon travail de recherche au centre aéronautique et spatial allemand (DLR - Deutsches Zentrum für Luft- und Raumfahrt). J'ai été membre de l'institut de Communications et Navigation au sein du département Réseaux Numériques (DN - Digital Networks) dirigé par le professeur Erich Lutz à Oberpfaffenhofen près de Munich en Allemagne d'octobre 2000 à novembre 2006. Durant ces six années, en plus des activités du DLR liés aux projets (et assurant le financement de l'institut), j'ai pu conduire un thèse coencadrée par Axel Jahn au DLR et par Michel Bousquet à Supaero (ISAE) Toulouse. Ce doctorat a été le fil conducteur de mon travail scientifique au DLR et a créé une dynamique à des activités liées à l'ingénierie du trafic et au dimensionnement de réseaux satellite. Reparti pendant ces six années, mon travail s'est nourri de ma participation aux projets *Euroskyway* et *WirelessCabin*. Cette thèse a été défendue en décembre 2007 à Toulouse.

Le thème du travail est l'élaboration d'un modèle pour la description hiérarchique de la génération de trafic au sein d'un réseau satellite. Une modélisation du trafic via un cadre hiérarchique à travers différents niveaux d'abstraction a été conduite. Des études par simulations ont été menées pour démontrer et valider l'intérêt de la modélisation hiérarchique. L'organisation du manuscrit est la suivante. Après un chapitre introductif présentant le contexte, l'état de l'art et l'organisation du travail, les chapitres 2, 3, 4 constituent le cadre théorique de l'étude en élaborant : un cadre général hiérarchique, une revue des modèles de trafic et un scénario servant de base à des investigations avec un simulateur. Les chapitres 5, 6 et 7 présentent les résultats obtenus pour des cas de complexité croissante. Enfin, le chapitre 8 résume les résultats obtenus, tire les conclusions et donne des perspectives pour des travaux futurs. Le chapitre 9 est un synthèse du travail en français.

Contents

Ces jours qui te semblent vides
Et perdus pour l'univers
Ont des racines avides
Qui travaillent les déserts.

Patience, patience,
Patience dans l'azur!
Chaque atome de silence
Est la chance d'un fruit mûr.

Paul Valery

1 Introduction

Abstract

A concept of aggregated traffic model for the hierarchical description of traffic generation in a satellite communication system is proposed in this research work. Traffic models are mathematical models developed in order to describe the streams of data conveyed in a telecommunication network. An aeronautical satellite communication system is selected as a reference simulation scenario to demonstrate the validity of the concept. The traffic behavior at the several layers of the reference scenario is analyzed and a methodology for hierarchical description of traffic generation is proposed. Simulation results show the relevance of aggregate models at the various hierarchical levels. Finally the applicability of aggregate models to support system dimensioning and performance analysis is illustrated.

As an introduction, this first chapter provides the reader with the basic definitions and discusses the scope of the work and the problem statements. The proposed methodology and the organization of the thesis report are presented.

1.1. Overview of data traffic analysis for satellite communication networks

1.1.1. Traffic engineering

Traffic engineering has three principal objectives: planning, analysis and optimization of network operation. Traffic engineering is broken down into three type of activity: traffic measurements, traffic characterization and traffic modelling ([Gog01]).

The principles of tele-traffic engineering are illustrated in figure 1.1 ([Wir97]), that shows the inputs, the actions conducted on the network (for example by the implementation of traffic shapers or the optimization of networks elements) and the monitoring of improvement obtained by measurements of the data traffic or performance indicators in the network.

The discipline of traffic engineering is based on historical network measurements to derive statistical models that are used to anticipate growth trends, to plan, design and implement network facilities.

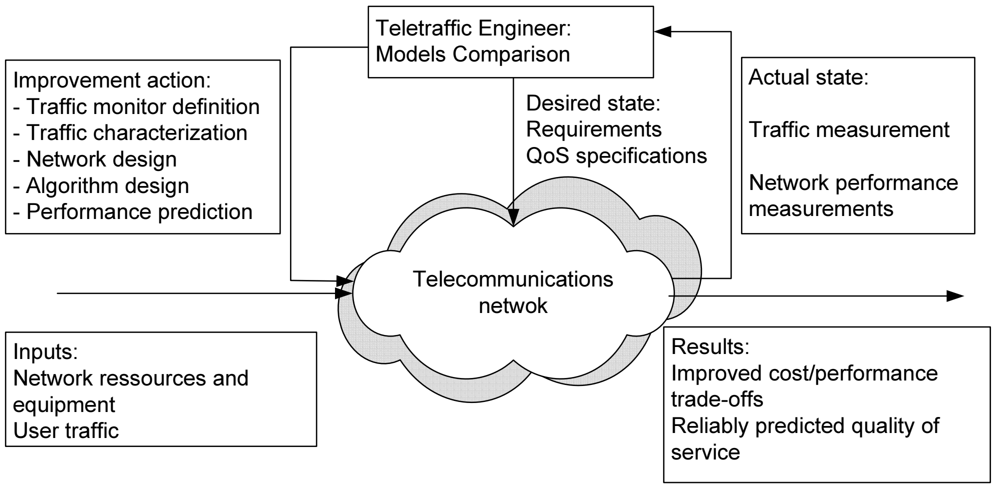


Figure 1.1.: The principles of tele-traffic engineering from [Wir97]

Traffic engineering can be applied to all kind of networks, and in particular to satellite communications networks. In this case, model accuracy deserves specific attention because of the limited satellite capacity.

For telecommunications operators, the measurements of data traffic on their networks are critical data that is not publicly available. Therefore a lot of research has been performed on university networks or research networks that made their traces publicly available (for example located at the internet traffic archive - [INT05]). This has the advantage that the trace were inve stigated by different groups with different methodologies.

As said before traffic engineering encompasses: traffic measurement, traffic characterization and traffic modelling. In this thesis, the last two are of great importance because models at different scales are derived. But the analysis of traffic measurement is of course necessary to provide the inputs for the other tasks.

1.1.2. Satellite systems

Satellite system development was started in 1960, following an idea from Arthur C. Clarke in 1945 [Cla45]. He showed that the extra terrestrial relays (or satellites) could easily be used for telecommunications, in particular for applications like intercontinental telephony. The advantages of satellite systems over classical terrestrial systems (service of large region areas communications, well suited for star topology and broadcast application, short time establishment and flexible architecture) made it a popular component of the telecommunication infrastructure. In 1965, INTELSAT I (”early bird”) was the first commercial system. New applications like TV and radio broadcasting popularized the use of satellite. In 1962, a television signal was relayed from Europe to North America over the Telstar satellite. The first domestic North American satellite to carry television was Canadas geostationary Anik 1, which was launched in 1973. From the mid seventies, satellite also played an increasing role for mobile communications with aeronautical, land-mobile and maritime terminals. With the ubiquity of TCP/IP, satellites were also thought to be used as a niche component of a global backbone. Dedicated VSAT have been used to connect multinational companies. In the late nineties, systems like Iridium, GlobalStar and ICO, have been developed to provide better performance to mobile users. Systems like Astrolink, EuroSkyWay, SpaceWay in GEO or Teledesic, SkyBrige in LEO were supposed to complement the offer in terms of multimedia communication services. The emergence of these services has led to the development of dedicated standards like Satelite DOCSIS, IPOS, DVB-RCS and DVB-S2 indicates for satellite television.

Further details on system design are provided in textbooks ([MB03] and [LWJ00]) where challenging issues like orbit selection, link design, modulation and multiple access scheme are described. For example, different orbits for single satellite and constellations can be selected to obtain the targeted communication system. Geostationary satellites (GEO) have an orbit at 36000 km and remains at a fixed position with respect to the earth. Low earth orbiting satellites (LEO) have an orbit at 700-1500 km. At last medium earth orbit satellites (MEO) are located in between (10000-20000 km). The frequencies to be used between earth and satellite are regulated by the International Telecommunication Union (ITU) World Radio Conference in order to prevent interferences with other communication systems. For mobile voice systems, frequencies in the L and S band are used (about 1.5 and 2 GHz respectively). For multimedia systems, Ku (11-14 GHz), Ka (20-30 Ghz) and V (40-50 GHz) bands are considered. In addition to the satellite, communication systems require a terrestrial infrastructure composed, on top of user terminals, of gateways and a dedicated network linking the gateways in order to provide wide area connectivity. In particular one of the gateways has to host a network control center responsible of the management of the bandwidth and of the different connections in the system.

1.2. Problem statement and scope of the work

This section introduces the framework and the objectives of the research work.

1.2.1. Scenario under consideration

A multi-services communication satellite is considered where data are exchanged between peer applications.

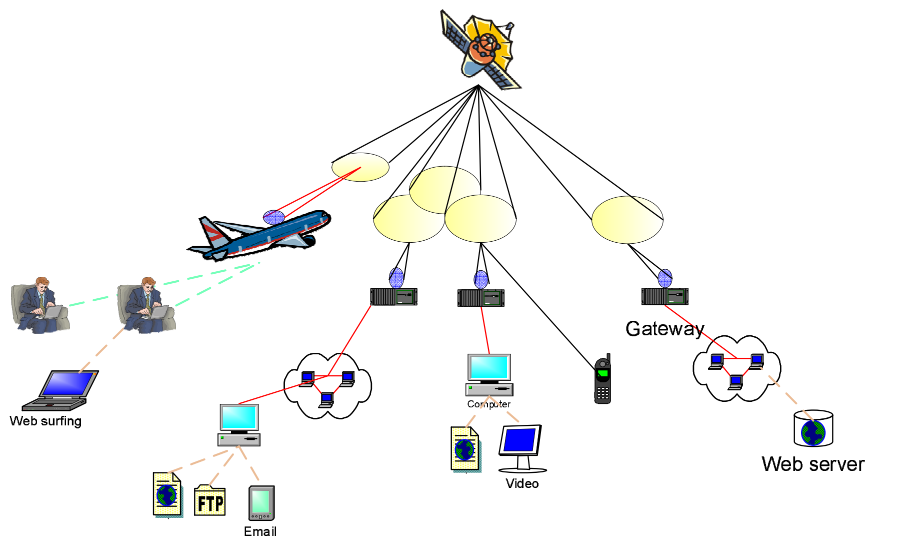


Figure 1.2.: Overview of the considered case

For example, satellite systems can be provided in different environments, such as the aeronautical, maritime or land-mobile environment (serving for example pedestrian, train and car passengers) as well as for fixed service offered as an alternative for terrestrial services (for example in remotely located areas). Such a multiplicity of environments implies a high diversity of the terminals accessing the satellite and a high variability of the applications used by the end users. Figure 1.2 represents an example of system configuration and type of user. Different kind of users are presented: passengers of an aircraft surfing the web from a on-ground server, a workstation user sharing access inside a company Local Area Network (LAN) to a satellite terminal using the system for web surfing, file exchange and email, a domestic user using the system for web browsing and video streaming, a telephony user with a satellite-enabled mobile phone.

In particular, the focus will be given to provision of aeronautical communications to passengers inside aircraft with a system similar to the one designed in the WirelessCabin project [Jah04]. The architecture is based on satellite systems to furnish connectivity inside the aircraft, refer to [JHRW02], [HJLW00], [DFG+03] for a more detailed presentation of the systems.

The general scenario gives also raise to a multiplicity of traffic profiles. After a description of each user, their individual traffic is considered and will constitute after overall aggregation the traffic load of the satellite. The process of aggregation of the individual flows into a common flow needs to be further described and modelled. Since satellite resources are costly and the satellite transmission is the bottleneck in the overall transmission, the description of the traffic transmitted across the satellite is of primary importance.

Moreover, the satellite transmission tries to guarantee quality of service. In order to achieve some acceptable performance for every connection, the most requiring connections will be limited in comparison to the less requiring ones, what will give raise to an heterogeneity in how the traffic is handled. Traffic modelling should also help to investigate if quality of service is achieved during service provision. As a consequence, the description of the overall traffic carried over a satellite system should enable to evaluate the performances of a system in particular in terms of lost and delayed packets. This description is complex because of the number and the diversity of the users (we can have groups of 1 to 1000 users) and the heterogeneity of the applications.

Therefore in this thesis, we want to characterize the overall traffic conveyed over a satellite link and in particular we will develop a model to characterize this traffic at different detail scales.

1.2.2. Objectives

The characterization of the overall traffic conveyed over the satellite can be broken down into several activities: define the traffic models best suited for the services provided by multimedia satellite system, develop appropriate traffic models for aggregated traffic flows, study the influence of limited resources on these models and predict system performances using the aggregated models. As a consequence, a framework will be created that is able to describe the aggregation of traffic streams.

So the objectives of this work can be formulated as follows

1. develop an abstract framework for traffic modelling at different level of abstraction
2. review the different traffic models
3. construct aggregated traffic models and show their equivalences
4. obtain performance evaluation with these models

An appropriate description of traffic can be of benefit:

1. for traffic dimensioning, in particular to help satellite system planners in system design
2. for investigation of quality of service (QoS) provision over such networks.

1.2.3. Methodology for problem solution

In this work, a hierarchical organization for traffic modelling is investigated. This description is close from the topological organization of the network, where the different network elements are considered at different levels. The network (data) traffic of each of the elements can be described by aggregate models for the overall level traffic description. The aggregate traffic models represent an approximation of the (more accurate) traffic models at lower levels. This approximation needs to be precisely evaluated in order to keep trace of the considered hypothesis. The models providing good approximation need to be described in details together with the procedure to estimate best fitting parameters. Afterwards the accuracy of the equivalence needs to be accessed to estimate loss of information.

1.3. State of the art

1.3.1. State of the art in teletraffic

The tele-traffic studies available in the literature are considering four different types of connections: a) circuit switched connections with homogeneous traffic b) circuit switched connections with heterogeneous traffic c) data packet connections with homogeneous traffic d) data packet connections with heterogeneous traffic. The most handled case is the case a) which can be treated with the Erlang loss formula.

1.3.1.1. Investigations on terrestrial networks

Web models and self-similarity Many investigations on traffic have been realized on terrestrial networks. The first results that triggered new research on traffic model was the discovery that Poisson models could not be applied to characterize web traffic and the collected traffic trace were exhibiting self-similarity (see for example [PF95]). This property means that the time series are bursty and these bursts are retrieved on many timescales. It has been evidenced on many kind of ”multimedia” traffic. In particular Ethernet traffic streams, video traffic and Web traffic exhibiting self-similarity was evidenced by many researchers. The Hurst parameter H ([Hur51]) is giving a grade of this property. In [BG05a], this grade was determined for different services for a single terminal, completing the investigations on backbones performed by Leland and all [LTWW93]. Since then, new estimators for self-similarity were developed for example by Veitch and Abry using wavelet analysis [VA99]. Following (re)investigations of IP traffic traces have conducted to a general rejection of the Poisson model as initiated by [Pax94]. One drawback of self similarity is that aggregation does not smooth the overall trace and has more stringent requirements on the buffer size. The formulas from Norros ([NRSV91]) provide bounds for the buffer size depending of H.

Moreover a multitude of web models have been proposed based on different measurements. These models were mainly developed to describe the total traffic on a network backbone. To be included in our study, we need a model capable to describe links of limited capacity. Since the traffic is elastic, blocking can not be described with the Erlang formula, but an equivalent to characterize web traffic would be useful. In general web models were developed independently where each group has developed its own model without performing comparisons with previous models. This doesn’t ease the selection of one particular model.

Model aggregation Considering aggregation of sources, there are not so many studies available, taking into account in particular the constraint of a (satellite) link with limited bandwidth. In his thesis [Hap98], Hapke presents an approach for successive characterization of web sources. Characterization are performed at different levels with a model used at each time. Also, matrix-analytic methods have been used to get analytic results for queueing characteristics of web traffic, with a state based modelling. For example, in her PhD [Ris02], Riska describes state aggregate methods to calculate more easily state distribution. Since the states used by these models are not related with physical elements, it is not interesting for the superposition of models.

[AMS82] was one of the first attempts to describe the superposition of sources and to analyze system performances. The superposed behavior is obtained by the numerical values of the roots of a characteristic polynomial depending of the number of sources considered.

1.3.1.2. Queueing theory

The necessity to model the amount of resource needed to provide communication was necessary from the beginning of telecommunications services. For this reason, queueing theory was always an important companion of traffic modelling. Further developments in queueing theory are of benefit for traffic modelling.

Short historical overview The pioneering work was realized by Erlang (1878-1929, refer to [BHJ99]) in a context of circuit based telephony. Erlang showed in 1909 that the Poisson distribution applies to random telephone traffic and determined the now classic formulae for loss and waiting time. As a tribute to the work of Erlang, the C.C.I.F. (Comité Consultatif International Téléphonique, a predecessor of ITU Telecommunications sector (ITU-T) and Comite Consultatif Internationale de Telegraphie et Telephonie (CCITT)) decided in 1946 that Erlang is used as the traffic unit to honor his memory even if it is dimensionless. Some of the other major pioneers in queueing theory and dates of their major works are Tore Olaus Engset (1865-1943) (first paper on queueing 1918), Felix Pollaczek (1892-1981) (first paper on queueing 1930, [Pol57] who developed transform formula for queue length distribution), Conny Palm (1907-1951)(first paper on queueing 1936,). Further work was done by Andrey Kolmogorov (1903-1987), (first paper on queueing 1931) Alexander Khinchin (1894-1959), (first paper on queueing 1932).

In 1953, David G. Kendall introduced the A/B/C type queueing notation. He also derived new settings for the integral equations describing certain queueing systems in [Ken53] which are now quite standards. In [Ken51] also generalized the usage of the embedded Markov chain. Lindley's Integral equation (from D.V. Lindley, [Lin51]) is a formula to investigate queues with more general arrival and service distributions.

The proof of Little's formula (so called because it was first proved by John Little) was published in 1961. Lajos Takacs effectively applied combinatorial methods to queueing theory. He developed an integro-differential equation for the waiting time distribution in the transient state for an ordered queue with Poisson input whose parameter depends on time and for arbitrary-holding time distribution ([Tak55]).

D. R. Cox introduced the Supplementary variables technique (1965) to analyze queues. Ronald W. Wolff named, popularized and gave the first rigorous proof of the "Poisson Arrivals See Time Average (PASTA)" principle ([Wol82]). In essence, it states that observations made of a system at time instants obeying a Poisson process, when averaged, converge to give the true value, that is to the average that an ideal observer would make when monitoring the system continuously over time.

All these researchers and many of their scientific descendants have further developed the theory of queuing and it becomes difficult to cite all of them. But at this point, it is necessary to add Marcel Neuts who introduced the matrix analytic method (1981). As editor of the journal Communications in Statistics - Stochastic Models, he promoted a large variety of queueing models. He contributed a lot to the description and investigations of the models described in the next paragraph.

To complement this historical overview, a summary of useful queueing results related to this thesis are gathered in section 4.4.2.1

Matrix geometric solution in queueing theory The development of Markov Modulated Poisson Process (MMPP)/Batch Markovian Arrival Process (BMAP) models was pushed by some research group after the advances of Neuts ([Neu81]) and the development of matrix geometric methods have given birth to methods for the derivation of methods for the calculation of queuing characteristics of BMAP/G/1 queues [Luc91]. The development of the BMAP benefitted from this approach and lost his name of N-process (N for Neuts) it had for example in [Ram80]. This model enables the calculation of the waiting time distribution that was then only obtained by transform relations. Furthermore, the group of Klemm ([KLL02],[KLL03]) derived a tool for parameters derivation from IP trace measurement. The EM-algorithm based program derived in the framework of this thesis, for estimation of parameters of MMPP and BMAP is furnishing similar estimates as their tool.

1.3.1.3. Traffic models overview

Current Models review [RMV96] gives a picture of the different models that can be used to model network traffic. In [AZN98] the focus is given to what is expected from the traffic models: 1) need of variability between mean and peak rate, 2) presence of correlation (short range dependence or long range dependence), 3) convergence of aggregated models to a gaussian model; and then simulations are realized with a selected model (M/Pareto). This first example shows that the model selected is often tributary of what the model is expected to exhibit. In the literature, a huge number of models are present, and it would be unrealistic to review them all. For example, in [HKS99] or [KM00], a long review of models is conducted with more that 60 references, each of them describing a model or a traffic generator. The parameter of each model could be then tuned to have a more realistic fit to some measurements. This review of models is completed by a classification.

The models for network traffic consist in the description of a quantity (data volume, packet size) at time instants. Therefore they are referred to as stochastic models. In this category they have then distinguished:

* Renewal models
  - Poisson process
  - Bernoulli process
* Markov models
  - Markov Modulated Traffic models
  - Markov Modulated Poisson Processes (MMPP)
  - Generalizations like Phase type renewal processes, Markov renewal processes, Batch Markovian Arrival Processes, Discrete-Time Markovian Arrival Processes
* Fluid models
* Linear stochastic models
* TES models (Transform Expand Sample)
* Self-Similar traffic models
  - Fractional Brownian Motion
  - Fractional Gaussian Noise
  - Fractional AutoRegressive Integrated Moving Average (ARIMA)
  - Wavelets
  - On/Off Processes
  - Poisson-Zeta Process
  - Deterministic Chaotic Maps
  - Aggregated heavy tailed models
  - M/G/∞ Model
  - Self-Similar Markov Modulated
  - GBAR (Gamma Beta Auto Regressive) and GBMA Processes (Gamma Beta Mobile Average)
  - Spatial Renewal Processes

The first family is coming from the Erlang models for telephony. The family of self-similar models have been developed in order to mimic some web traces that had revealed such behavior for example when they are observed at different time scale, they have a ”similar” behavior. The fluid models arise from a different approach, when instead of describing unit of packets they are assimilated to a flow of fluid. The information transfer is then similar to water running out in pipes. Continuous and discrete are suggested for example in the self-similar models: the ARIMA model is suggested as a continuous model and the Poisson-Zeta as a deterministic model. It is also difficult to describe all these models accurately. In this work, the focus is given on a few models, that are well-suited for the considered study. These models are summarized in section 3.5.

Deterministic or stochastic models In this work, only stochastic models were used even if traffic can be described either statistically or deterministically.

Investigations with deterministic models have been developed in the framework of network calculus theory. The network calculus is a theory that has developed a dedicated algebra (like the min,plus algebra) that enables to describe entities of a communication system (cf to [LBT01] for further reference). The theory provides bounds for delay, but for the provision of distribution it does not bring many answers. Every element of the network is characterized by an operator in the adequate algebra. Considering all of them enables at the end to get estimates (bounds) on the system behavior. Newer research is underway to develop a stochastic branch to increase the amount of results that could be obtained. Network calculus is becoming a self contained theory able to predict the behavior of TCP proposed in [BH00]. Bounds obtained in a simple system for web browsing were not very significant, so not so much attention was given to this theory.

On the contrary, stochastic estimates were preferred, because the output obtained are more directly usable. For example, for the investigation of sources, it was easier to get distribution of the data rate. With deterministic models only a data volume function is obtained. This does not ease comparison of models.

Multi-service models For the study of data traffic in satellite system, the approach proposed in the chapter 12 of [LWJ00], using a descriptor for every service, is providing a model for every service component has a valuable applicability. Design of networks including diverse services was done by superposing the individual models leading to the concept of a ”multiservice” traffic model. Each model is described by its session characteristics. The table 1.1 shows the parameters used for inter-arrival time and session duration proposed in [WL98]. It shows the different services that have to be considered. This was successfully applied in the ASSET and SECOMS project for the design of satellite systems, with a particular focus on system with time variable constellation. This topic is fully covered in a PhD thesis [Wer02].

Service/Application	Session Birth Rate	Mean Holding Time
Telephony, Fax	1/h	3 min
Video telephony	2/day	5 min
Shared Applications	1/day	60 min
TV broadcast	2/day	2h
Audio broadcast	2/day	2h
WWW	5/day	30 min
Email	5/day	0.25 sec
File Transfer	5/day	10 min

Table 1.1.: Statistical parameters used for application modelling in the multi-service model

This model was recently further used by [LH05] where a multi-services models was used for system planning of an aeronautical system. The models for web and ftp volume were adapted to current conditions, because older models would under-evaluate the current volume (popularity and frequency of use have certainly is increasing in time).

1.3.1.4. System performance investigation

To our knowledge, there are few studies of investigation of system performance that are combining analysis of performance together with realistic traffic models. In this work, evaluation of performance metrics will be performed with different traffic models. Some of these traffic models will be constructed to be equivalent to other ones. The performance metrics in this case shall provide similar outputs.

1.3.1.5. Level of operation and time scale of traffic models

Traffic models operate often at cell level. Other phenomena could be considered like daily, weekly, yearly variations. These variations play an important role for telecommunication operators that prefer to see a constant load that a variable load. Similarly, geographical repartition of the population can modify the traffic volume according to the amount of population in the area considered.

Moreover, traffic models could operate at different scale. The first scale is a long-term view for example if models are used in order to get hourly or daily estimation of the resource requirements of one terminal. A second scale is a connection-level that will require traffic estimates at scale equivalent to connection length, this duration will be of the order of a minutes. The third scale will be at cell level that will be necessary if resource allocation is modelled precisely. For example in DVB-RCS standard, the resource allocation is performed on "Super"frame basis i.e. every 26.5 ms.

Depending on what kind of results are required, one of the scale will need to be chosen. The first scale will give macroscopic results and should be able to run efficiently, the third scale will give very precise models

1.3.2. Satellite traffic modelling

Architecture and protocols At our level of consideration, the satellite is considered as a conveyor of data. Different standard try to federate the format of data traffic conveyed by satellite. A first tendency was to enlarge the Asynchronous Transfer Mode (ATM) protocol for the use in the satellite case. The protocol is popular for high speed networks and its usage for satellite was pushed till the late 90. Afterwards, the Digital Video Broadcasting - Satellite (DVB-S) offered a set of protocols for satellite data transfer that is preferred because of the wide availability of chipsets developed for the TV market. In the next chapter, the satellite system will be considered as an element of the transport level and the type of protocol used is not playing a major role. Both protocols will be kept in background as the possible practical realization for real systems.

In the projet Atm-Sat ([BWD+05]) a generic architecture for such system is proposed. For what concerns traffic models over satellite, a review of models is presented in [Ryu99] with the corresponding architecture in [CRD99]. They introduce the concept of effective and actual sources, that will be generalized in chapter 2.

Performance analysis ATM over satellite [GJG+99] was first investigated because the ATM was becoming a broadly used protocol in terrestrial networks. In [MSTBGE99], the terrestrial scheme of resource allocation were generalized to the satellite case. The satellite system based on ATM did not really come to reality, what would have enabled further investigations of networks performance.

The theory of equivalent bandwidth provides a measure of resource usage which adequately represents the trade-off between sources of different types, taking proper account of their varying statistical characteristics and quality of service requirements. For references on equivalent bandwidth refer to [Kel96].

In IP based networks, there are some schemes that allow the concept of Quality of service (QoS) to be considered. Mainly in such networks two different architecture could be chosen: the first one is integrated services (Int-Serv) and the second one is differentiated services (Diff-Serv). In the first case, all flows are considered in an unique main stream and in the second case, different flows are maintained in order to reach different negotiated QoS levels.

Considered services The services can be classified in real time services or non real time services. A third category would be killer applications that are requesting more that what the system is able to provide and that should be hence be identified and avoided. Inside the real-time services, a further distinction can be done between real time conversations services and real time streaming services.

In this work, the services offered by the communication system are mainly IP services. It means that the entities used to transfer the data are mainly IPv4 packets.

The services considered will be:

- voice,
- emails client,
- video client,
- web browsing.

The corresponding traffic models will be presented with further details in section 3.2.

1.4. Work organization

After this introduction, in chapter 2 the aggregate traffic model pyramid will be presented. This is the framework that has been chosen to investigate the traffic flows exchanged in the system under study. The rest of the work is strongly conditioned by the organization of the pyramid model.

In chapter 3, the basis for traffic modelling are exposed. Some models of particular interest for the remaining modelling work are further discussed. The definitions introduced in chapter 2 enable the description of some traffic models. The models from the family of MAP (Markov Arrivals processes) are described. In particular, On/Off, Phase Type (PH) renewal, MMPP and BMAP traffic models are presented because they are used for application modelling. In addition, the methods for the identification of the models parameters are listed. Since no unique method for deriving the parameters in every case is available, different methods for obtaining the model parameters are presented.

In chapter 4, the system that was chosen for traffic investigation is exposed and the operators in use are instantiated. This chapter provides the transition between the two theoretical chapters and the results obtained with a simulator. The scenario considered is hence described with the concepts and operators that were introduced in chapter 2.

In chapter 5, results of multi-levels study of voice aggregation are presented. Taking this particular example, the aggregation at different levels is presented. In particular, the match between "equivalent" models are investigated: analyzing if the generated flows have similar or different characteristics. This study is generalized in the next two chapters.

In chapters 6 and 7, the results at respectively the node and transport level that corresponds to the different levels of the pyramid are presented. A careful investigation on the modifications that occur to the traffic stream during transmission over the pyramid is performed. Chapter 6 focusses on the results obtained at node level. Gains in complexity are obtained at node level, more easily that would be obtained at subscriber level. Chapter 7 reviews previous results viewed at the transport level and demonstrates equivalencies between different models.

Within chapter 7, an investigation with purpose of system dimensioning of a particular scenario is done using the models exposed earlier. The results here are more practical results, for example estimation of future system performance metrics.

Chapter 8 summarizes the principal achievements as a conclusion and gives direction for further research.

This organization is then following the pyramid approach in the following sense: Chapter 2 introduces the pyramid traffic model, Chapter 3 populates it with the traffic models, Chapter 4 is explaining how it was practically used. Chapters 5, 6 and 7 present numerical results justifying the overall approach.

*C'est une **pyramide**, un immense caveau,*
qui contient plus de morts que la fosse commune.

Baudelaire, Spleen LXXVI in Les fleurs du mal

2
The aggregate traffic model pyramid

In this chapter, a model for the overall traffic description in a communication system is presented that is applicable to scenarios like the reference one introduced in the previous chapter. With this model, different levels of description are distinguished: the application level, the subscriber level, the node level and the transport level. First, the main principles of this model are exposed. A description of the different levels then follows with a review of the appropriate parameters for the characterization of each level. Afterwards the propagation of data packets between the different levels is analyzed and other aspects of the hierarchical modelling are introduced. Keeping this modelling approach, performance metrics are defined. Finally a scenario is introduced that will be analyzed numerically later in this study. To summarize, in this chapter, a hierarchical organization of traffic modelling is introduced in order to give the overall framework for this research work.

2.1. Principle of hierarchical modelling

In this section, we present our model that is based on the hierarchical description of the sources of information in a communication system. Depending on the required level of details, an abstract model has been constructed with the shape of a pyramid to reflect the hierarchical organization. This pyramid can for example describe the traffic that is transmitted over a satellite system as shown in next section.

The purpose of this model is to enable a description (over different levels of details) of the data traffic generated by the source of information in a telecommunication system. Indeed, a telecommunication system is concerned with the transmission of information from a source to a sink. This transmission generates a given volume of information, that we want to characterize. The transmission of information between the source and the sink is realized through different levels, that are identified in the next section. However, this transmission through these levels occurs with modification from the inter-level interfaces, that modify some characteristics of the data transmitted. Also in parallel to the description of the model, some definitions and notations will be given in order to describe the data traffic, depending on the level at which it is measured.

To simplify the description, the traffic flows will be considered from the bottom to the top. By proceeding this way, the volume of data traffic will increase with the level, therefore requiring traffic models capable to describe the aggregated flows up to the highest levels. The construction of such an aggregated model will be done level by level with the target level being the description of a traffic model over the satellite transport level. Before to proceed with the model construction, the considered levels need to be precisely defined.

Definition 2.1.1 (Aggregate traffic pyramid)**.** *The aggregate traffic pyramid is a modelling approach to*

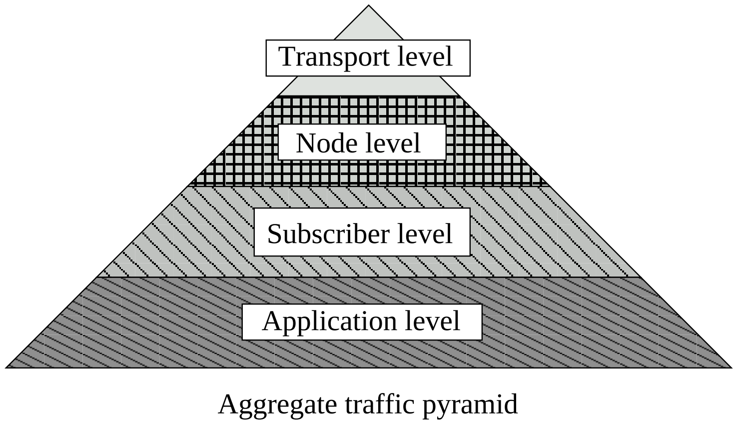


Figure 2.1.: The aggregate traffic pyramid concept

investigate traffic model over a transport level. The following knowledge is assumed:

- *all the applications used in the system are noted a_i, where a_i is a set of parameters characterizing completely the application i.*
- *all the subscribers registered in the system are noted s_j, where s_j is a set of parameters characterizing completely the subscriber j.*
- *all the access nodes of the system are noted n_k, where n_k is a set of parameters characterizing completely the node k.*

The different associations of the 3-uple (i, j, k) have a pyramidal organization.

Figure 2.1 illustrates the pyramidal shape of the traffic aggregation modelling process.

2.1.1. Example of model usage

The aggregate traffic model pyramid is used to describe the entities involved in the generation of traffic in a communication system. In the following scenario, depicted on figure 2.2, the pyramidal hierarchy is used to describe the entities involved in this typical case for multimedia application services provided by satellite. Two pyramids were used and each pyramid corresponds to either the client side or the server side. They have a common concentration point at the transport level. The basis of the pyramid, the application level is after rotation, at the left side for the client application and the right side for the web server. Our goal in this example is to get traffic models at the different levels of aggregation in the system. In particular a traffic model at the transport level is needed that could be combined with an application traffic model in order to evaluate the user satisfaction in a loaded system. The depicted scenario corresponds to the download by an user of a domestic satellite terminal of a web page located in the satellite ISP server. The ISP is offering internet content over satellite for remote areas. At the application level, a web browser client which peer is a web server located on a computer in the ISP Network is considered. The subscriber of web browser client application is located in a local LAN which share access to a satellite terminal for a collective SOHO company. On the peer side, the web server is the single subscriber of gateway terminal of the ISP providing internet access. In this case, the effective performance of the web access can be investigated, to study if this service can be provided with acceptable rate, to evaluate the requirement for the satellite and the number of acceptable customers.

In the case presented the entities are the following: at the application level, there are the voice client, the video client and the web client...; at the subscriber level, there are the computers used by the user; at the node level, there is the satellite terminal, at the transport level there is the satellite system. The pyramid organization seems able to map concrete communications system as will be shown again in the following description of the investigated test case. Afterwards, the requirements from a traffic model will be derived.

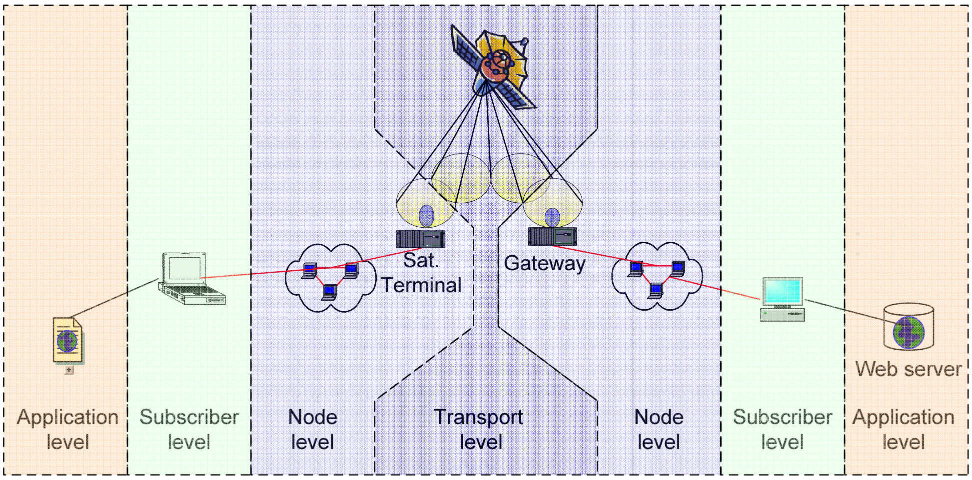


Figure 2.2.: An exemplary scenario with the different level of description

2.1.2. Investigated test case

In this thesis, the ideas developed for traffic modelling will be applied to a specific case similar to the one developed for the WirelessCabin project ([Jah04]). The aim is to provide aggregate models for a satellite system providing voice and internet access to aircraft passengers. The considered applications will be voice and internet access, the subscribers will be the aircraft passengers, the node will correspond to aircraft satellite terminal (one per aircraft), and the transport level will be assured by one satellite system with a return channel compliant with the DVB-RCS standard. In a first time, aggregated models with detailed models to prove their practical validity will be compared. In a second time, a system analysis with the help of these models will be performed. The parameters (details of applications parameters, number of subscriber per aircraft, number and type of node considered, satellite segment parameters) of the system will be detailed in the later presentation of the test case, but at this point we present the entities considered in their pyramidal organization.

2.1.3. Requirements for the modelling approach - Fundamental features for the model

The needs for traffic modelling are summarized hereafter. First, the modelling and its required level of abstraction is evaluated. Afterwards the characteristics of the model are enumerated and justified. Finally the relevance of data modelling is discussed.

The modelling of traffic is based on the description of packets exchanged inside the system over time. These packets need to be described at the different entities where they are transmitted from the source to the sink. In this approach, these entities are organized on different levels, that corresponds to their physical position in the network. The hierarchical relationship between the entities is indicated by the shape of a pyramid (as shown in figure 2.1). Another requirement is that the models shall be able to describe the diversity of services considered at the application and to provide an estimation of their respective influence on the transport level.

The models have to provide the following features:

- Comparison between two different models must be possible
- Measures for performance evaluation shall be possible as:
  - describe packet volume
  - describe free slots for insertion of other flows of traffic
  - get performance estimates (evaluation of QoS per stream, delay)

At this point, we can also review the benefits that are expected from such a modelling approach.

1) Ease of use (the use of aggregate models shall ease the complete system design)

2) Scalability (each of the levels can be easily re-scaled)
3) Modularity (One part can use aggregated models and meanwhile accurate simulation can be maintained)

Another point to tackle is the precision that is expected from a traffic model. In the following, stochastic models are proposed to model the stream of traffic. Each realization of such model has a random origin. Between the modelled stream and the original record, there will always be an inherent difference between the model and the original target. This must be remembered when comparison are performed. It will therefore be necessary to develop criterion or methods to compare models.

2.1.4. Data model: a level-centric organization

In this section we present how the data are exchanged by the system at the different levels and what are the consequences for the exchanged traffic.

2.1.4.1. General concepts

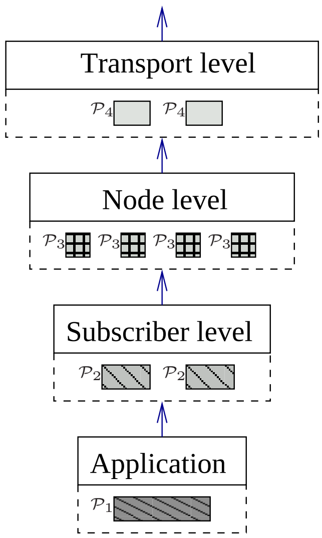


Figure 2.3.: Protocol stack in the aggregated model

Data packets Figure 2.3 represents the data that are exchanged between the different levels. Each level is using specific packets either of fixed size or of variable size. The figure shows in particular that the packets of one level have a different format than the packets from the other levels. The packets have been named $\mathcal{P}_1, ..., \mathcal{P}_4$. The differences in the packet formats used at the different levels are indicated by the different colors for each level. To handle the different type of packets, an interface between the different levels is necessary. This interface is responsible of the conversion of the packets at the different and is performed using the operation of fragmentation and header insertion on one side and its opposite operation on the other side (header extraction and reassembly of packets).

The size of the data manipulated by the different levels, the packet size and the overhead are now defined.

Definition 2.1.2 (Packet size)**.** *At a given level ℓ, the size of a packet $\mathcal{P}_\ell$ is called S_ℓ.*

This size may be bounded by a value $S_\ell^{\max}$ which is the maximum packet size. This size can be different for every packet. In the definition of the packet stream, the sequence of the size is noted $\{\mathcal{S}\}_n$.

Definition 2.1.3 (Packet payload and overhead)**.** *At a given level ℓ, the payload of the packet C_ℓ is the size of the information bits in the packet. The overhead O_ℓ is the information added on the packet for its transmission over the level. Of course $S_\ell = C_\ell + O_\ell$*

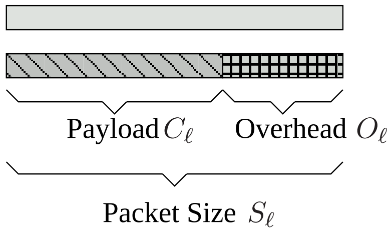


Figure 2.4.: Representation of the size of a packet

In some cases, like header compression for voice transmission, the overhead O_ℓ can be negative. For example in case of header compression for the traffic generated by a GSM base station, a scheme can be implemented that transmits a batch of packets together and removes the repetitive header of these packets when they are send together and replaces by a common (or enabling reconstruction) header. If the size of the added header is smaller than the size of the header of the suppressed header, then $\mathcal{O}_l < 0$. This reduces the transmission capacity requirements.

Figure 2.4 shows an example of a data packet with the partition between the payload part where the information bits are stored and the overhead part where the header, check sums and other information useful for the packet integrity and delivery are contained.In practice the location of the information bits in the packet can be organized differently.

The header insertion is the addition of information necessary for the transmission inside the considered level.

Level translation operations When the transmission from packets from the application level to the transport levels, some operation are necessary in order to modify the packets format for their correct transmission over the different levels.

Definition 2.1.4 (Fragmentation)**.** *When the size of a packet is not compliant with the format of the considered level, it can be splitted in d smaller packets which are compliant with the format for the considered level.*

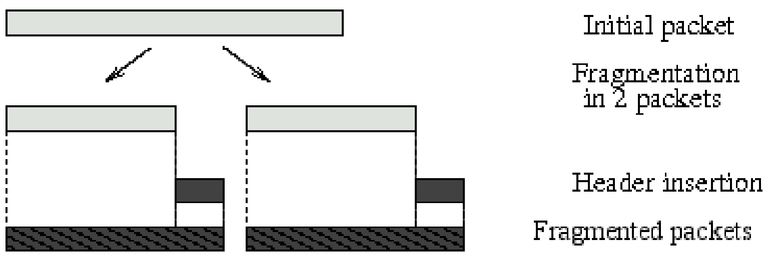


Figure 2.5.: Operation of fragmentation

Figure 2.5 represents the fragmentation of the previously described packet (with its information content and its overhead) in two packets. The fragmentation is performed on the total size of the packet and the overhead corresponding to the correct level format is added (if header compression is performed, then the header can be negative). The header insertion is the addition of data necessary for transmission management with the considered format.

Definition 2.1.5 (Level interface)**.** *To perform the transmission of one packet $\mathcal{P}_i$ from the level $\ell = i$ to the level $\ell = i+1$, level interface functions are called to translate the incoming packet into the next level format. These interface functions will include for the upstream transmission*

- *a function lower level data request sent by the lower level to the interface to request data transmission to the upper level.*

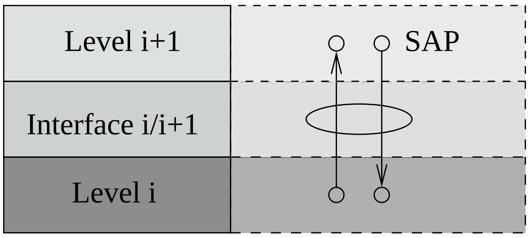


Figure 2.6.: Principle of the interface between two levels

- *a function upper level data indication sent by the interface to the upper level to indicate that data can be send*

Figure 2.6 shows the interface between two levels. The level $\ell = i$ is called the lower level and the level $\ell = i + 1$ is called the upper level. The interface functions are called via Service Access Point (SAP) which go through the levels.

Many to one data aggregation The hierarchical organization of the pyramid also implies that the ”many to one” structure will be often encountered. We have a tree structure over the levels with the trunk at an upper level (target level) and leaves in an lower level (origin level). The data flows from all leaves are aggregated to form a new data flow in the target level.

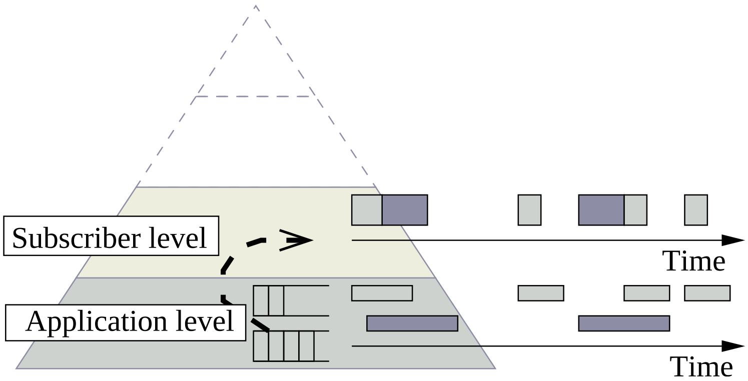


Figure 2.7.: Addition of two applications traffic streams into a subscriber traffic stream

Figure 2.7 shows the addition of two applications into a subscriber traffic stream. The target level is the subscriber level and the origin level is the application level. The interface between the two applications and the subscriber level is receiving the data packets from the applications. The output rate of the subscriber was assumed to be twice the incoming rate, so received packets are served two times quicker. For this reason, the packets are represented shorter at the subscriber level (half of the original length) but with a double height in order to keep a constant amount of information transmitted.

Data transmission in the pyramid Since we are investigating the data flows during their transmission over the pyramid, the data have to be transmitted over all levels of the pyramid. So the transmitted data suffer from translation over the different levels.

Figure 2.8 shows an expanded form of the pyramid. Packet format translation occurs at the interfaces between the different levels. Moreover, the aggregation of the packets of one level (origin level) to the next level (target level) is generating packets conforming to the format of this level. The packet format in the picture has the same representation as in figure 2.3 with the format of the aggregated output. For example the block of application of the detailed subscriber are aggregated at the subscriber level. For this reason the border of the application has the format of $\mathcal{P}_2$ packets, which is the format in the subscriber level.

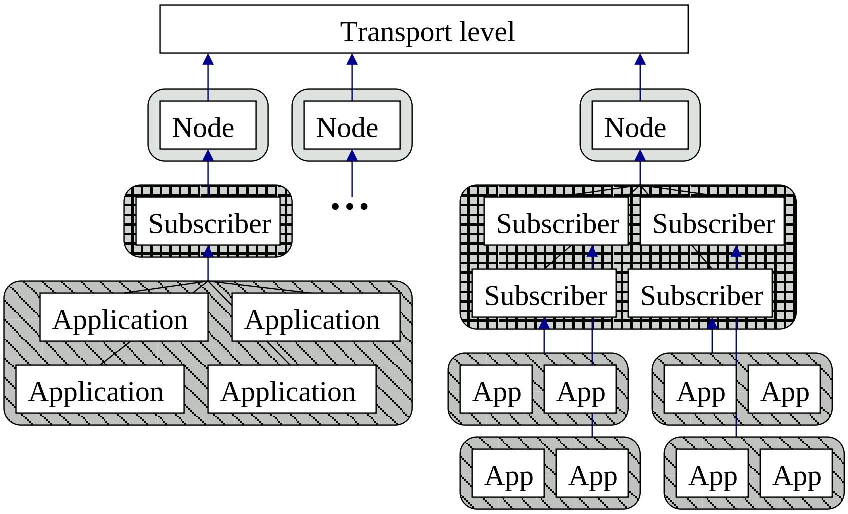


Figure 2.8.: Pyramid in its expanded form with the different packet formats

Level	Name	Packet Size	Overhead
Application	TCP	≤ 1500 Bytes	20 Bytes
Subscriber	IP		14 Bytes
Node	MPEG	188 Bytes	
Transport	Frame	3 Mpeg / TS	

Table 2.1.: Packets used in the example

Buffering If the transmission toward the next level is not instantaneously possible (or if insufficient capacity is available), the incoming flow of data packets will be stored in a local memory called buffer before transmission towards the next level. Buffer have an obvious impact of the delay of packets

Buffer is also occurring when the connection is being set-up, but in this work, established connections are considered so that the transitory effects occurring because of the connection establishment or release are neglected. A corrective factor should be considered to take this hypothesis into account. In real system, before connection establishment, a registration at the NCC is necessary, and to establish a virtual channel between source and destination.

2.1.4.2. Example

In the following example, the data packets used for the transmission of HTTP data packets from a client to a server using a satellite system are shown. This scenario is similar to the one considered in figure 2.2. The client is sending TCP packets to initiate the connection and to request a page from the server. These packets are encapsulated in IP in order to be transmitted from the subscriber computer to the node. This mechanism can be associated with the fragmentation of the data if the generated packets exceed the maximum TCP segment size. Furthermore, for the transmission from the node to the satellite, if the standard used in the satellite is DVB-RCS, the transmission will occurs in MPEG packets, so the interworking function taking place in the node will require another translation of the IP packets into the MPEG format. Here again, the content of an MPEG packet is limited to 188 bytes, so another fragmentation will take place.

The traffic flow at the transport level may have properties that are different from the ones at the previous level. In particular, fragmentation will result in a reduction of the size of the data packets but will increase the number of transmitted packets.

The description of the data units at the different levels is necessary to build a model of the data units that are transmitted in the system. The information about the packet size architecture is summarized in table 2.1.

2.2. The pyramidal architecture : a level-based aggregation

2.2.1. Levels and entities considered

The considered telecommunications networks are assumed to be representable with the following vocabulary. In particular the element of the aggregate traffic pyramid will be described accordingly. The next four abstract definitions from [Hap98] are recalled because they provide an useful framework for our study, which can be customized for the aggregated traffic pyramid. These definitions describe the flow of packets that are exchanged in the system.

Definition 2.2.1 (Network)**.** *A communication network is a set of entities and links.*

(The original definition was speaking of nodes instead of entities but this term is reserved for the name of particular level). The link convey data packets in which information can be inserted. These data packets are transmitted over the network.

Definition 2.2.2 (Link)**.** *A link is a medium able to convey data from one entity to another one.*

Definition 2.2.3 (Entities)**.** *An entity is a element of the network.*

We assume that each entity can be classified to one of the following levels. With this terminology, we have distinguished the following levels:

- Transportation level : the top-most level
- Node level : the entities that access to the transport level.
- Subscriber level : all the real users that share or have access to a node
- Application level : all the services that the subscribers are using and that are generating traffic.

Definition 2.2.4 (Levels)**.** *A level is the characterization of the type of entity. It is noted by ℓ where $1 \leq \ell \leq 4$. The application level is denoted by $\ell = 1$. The subscriber level is denoted by $\ell = 2$. The node level is denoted by $\ell = 3$. The transport level is denoted by $\ell = 4$.*

Each entity has one of the following type:

Definition 2.2.5 (Application)**.** *An application is a data communication service that is either an information source or an information sink. In the following, applications are considered as sources of the data traffic.*

Definition 2.2.6 (Subscriber)**.** *A subscriber is a computer or a mobile device (usually commanded by an human or functioning automatically) which provides data communication services. This device is connected to an access node in order get access to the transport segment. The data traffic generated by a subscriber is due to the activity of the corresponding applications.*

Definition 2.2.7 (Node)**.** *A node is an intermediate point in the data service provisioning. It is the ingoing and outgoing point towards the transport level. The interworking functions from the subscriber level into the transport level are performed in the node. The data traffic generated by a node is the traffic received from the subscriber connected to the node.*

Definition 2.2.8 (Transport)**.** *The transport mechanism enables the transmission of data between two nodes.*

The figure 2.1 represents this model. In the packet format presentation (cf. figure 2.3), the packets in the different levels were called $\mathcal{P}_i$. The number i corresponds to the level where the packet is manipulated. The hierarchical structure is then obvious: the application level is the basis and all the applications used by one subscriber "sums up" to build the subscriber traffic, the traffic of all the subscribers of a node also "sums up" to build the node activity, and finally the traffic from all the nodes "sums up" over the transport level. This construction of the traffic of level n using level $n-1$ will be called aggregation.

This operation needs to be precisely described and the equivalence of models shall be shown. Before that, traffic models for a single level will be more precisely described.

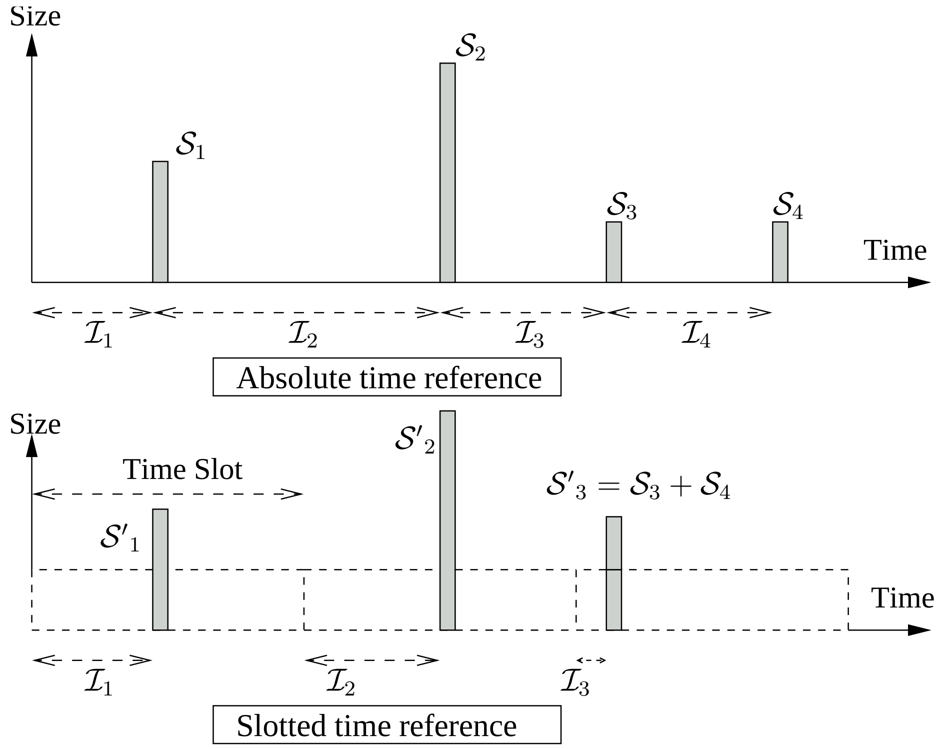


Figure 2.9.: Representation of a realization of an abstract unit stream

2.2.2. Traffic models

The aim of traffic models is to describe the traffic streams exchanged within the system. With the terminology introduced earlier, we have to characterize the sequence of packets $\mathcal{P}_\ell$ for each level ℓ of the considered pyramid.

2.2.2.1. Single level traffic model ($\mathcal{P}_1$ traffic models)

The following definitions enable to describe a traffic source. They introduce an abstract description of data container .

Definition 2.2.9 (Abstract unit)**.** *An abstract unit $\mathfrak{A} = (\mathcal{S}, \mathcal{I})$ is a pair of random variables $\mathcal{S}$ and $\mathcal{I}$.*

A companion definition is also introduced that will help the model description

Definition 2.2.10 (Absolute abstract unit)**.** *An absolute abstract unit $\mathfrak{A}^\star = (\mathcal{S}, \mathcal{T})$ is a pair of random variables $\mathcal{S}$ and $\mathcal{T}$ where $\mathcal{T}$ is strictly increasing (absolute time).*

The first interpretation of an abstract unit corresponds to the measurement of one data packet at a reference point. The random variable $\mathcal{S}$ corresponds to the packet length and $\mathcal{I}$ to the elapsed time with respect to previous packet (inter-arrival time). The second interpretation is done in a slotted system. An abstract unit is describing the time slot content: $\mathcal{S}$ is the amount of information in the time slot and $\mathcal{I}$ describes the packet timing in the time slot (relatively to the slot begin). If the time slot is empty, $\mathcal{S}$ will be nil. In [Hap98] this distinction is called ”packet semantic” and ”time slot semantic”. In the traffic model presentation, absolute and time-slotted models are distinguished and correspond to the same distinction. These two interpretations are depicted on figure 2.9 where the two approach for the same traffic source are presented. Since the packets P_3 and P_4 arrived in the same time slot, their sizes is added in the slotted-approach, because the slotted approach foresees only one traffic descriptor per slots, so some information is lost. If the time slot duration is short, the loss of information is minimal. From the slotted approach, the stream in the absolute approach can be reconstructed, assuming one packet arrival per time slots. Under this hypothesis, the description of the traffic can use each of the approach.

The absolute abstract unit complements the definition of the abstract unit. The first random variable $\mathcal{S}$ is identical to the one of the abstract unit and can be similarly assimilated to the packet size or the amount of information in the time slot. The second random variable $\mathcal{T}$ is the absolute time of the event is related to $\mathcal{I}$ in the abstract unit stream.

Definition 2.2.11 (Abstract unit in the absolute time reference)**.** *A pair $(\mathcal{S},\mathcal{I})$ is defined for every new incoming packet. $\mathcal{S}$ is the size of the packet and $\mathcal{I}$ is the inter-arrival time i.e. the difference between the arrival time of the packet and the arrival time of the previous one.*

Definition 2.2.12 (Abstract unit in the slotted time reference)**.** *Considering a given time slot duration T_s, a pair $(\mathcal{S},\mathcal{I})$ is required for every n describing the traffic seen in the time slot $[nT_s,(n+1)T_s]$. $\mathcal{S}$ is the overall size of the packets in the time slot and $\mathcal{I}$ is the position in the time slot of the first packet. If the time slot is empty, $\mathcal{S}=0$ and the value of $\mathcal{I}$ has not importance.*

Definition 2.2.13 (Abstract unit stream)**.** *A sequence of abstract units $\Phi=\{\mathfrak{A}_i, i\in\mathbb{N}\}$ is an abstract unit stream*

Definition 2.2.14 (Absolute abstract unit stream)**.** *A sequence of absolute abstract units $\Phi^\star=\{\mathfrak{A}^\star_i, i\in\mathbb{N}\}$ is an abstract unit stream*

An abstract unit stream and an absolute abstract unit stream enables to describe the output of a traffic model, because it is describing the sequence of packets over time. A finite sequence of abstract units can be collected by a measurement of the traffic over a link, for example with a traffic analyzer. The definition of the following operators enable the conversion between absolute abstract units and abstract units.

Definition 2.2.15 (Converter from abstract unit stream to absolute abstract unit stream at instant τ)**.** *Given a abstract unit stream $\Phi=\{\mathfrak{A}_i=(\mathcal{S}_i,\mathcal{I}_i, i\in\mathbb{N}^*\}$, an absolute abstract unit stream $\Phi^\star=\{\mathfrak{A}^\star_i=(\mathcal{S}_i,\mathcal{T}_i), i\in\mathbb{N}\}$ can be elaborated such as $\mathcal{S}_i=\mathcal{S}_i$ and $\mathcal{T}_0=\tau$ $\mathcal{T}_{i+1}=\mathcal{I}_{i+1}+\mathcal{T}_i$ for $i\in\mathbb{N}$.*
An operator $(.)^\star_\tau$ that transforms Φ in $\Phi^\star$ can be defined such as

$$\Phi^\star=(\Phi)^\star_\tau \tag{2.1}$$

Definition 2.2.16 (Converter from Absolute abstract unit stream to abstract unit stream at instant τ)**.** *Given a absolute abstract unit stream $\Phi^\star=\{\mathfrak{A}^\star_i=(\mathcal{S}_i,\mathcal{T}_i), i\in\mathbb{N}\}$, for $\tau<\mathcal{T}_0$ an abstract unit stream $\Phi=\{\mathfrak{A}_i=(\mathcal{S}_i,\mathcal{I}_i, i\in\mathbb{N}\}$ can be elaborated such as $\mathcal{S}_i=\mathcal{S}_i$ and $\mathcal{I}_{i+1}=\mathcal{T}_{i+1}-\mathcal{T}_i$ for $i\in\mathbb{N}^*$ and $\mathcal{I}_0=\mathcal{T}_0-\tau$*
An operator $\overline{(.)_\tau}$ that transforms $\Phi^\star$ in Φ can be defined such as

$$\Phi=\overline{(\Phi^\star)_\tau} \tag{2.2}$$

Definition 2.2.17 (Independent generator)**.** *An independent generator G is an abstract unit stream Φ parameterized by $(p_1,\ldots,p_k)$.*

A $\mathcal{P}_1$ traffic model is an independent generator.

Example For example, if we observe a TDMA satellite transport system at the node level, and assume that the slot 1 of the TDMA frame is assigned to this node (hence the node data have no delay relatively to each slot begin), and that this slot is fully loaded with data. Then, in a slotted description, the abstract stream will be $\{\mathfrak{A}\}_n=\{(\mathcal{S},\mathcal{I})\}_n=\{(S_3,0)\}_n$ where S_4 is the length of data that can be accommodated in one TDMA slot. For such traffic, a generator will have S_4 and the slot number as parameter.
On the other hand, for an example of absolute description, we can consider an application sending packets at a constant bit rate. If the packets have a size of 800 bytes and are sent every 40ms, the abstract stream will be $\{\mathfrak{A}\}_n=\{(\mathcal{S},\mathcal{I})\}_n=\{(S_1,I_1)\}_n$ where $S_1=800$ bytes and $I_1=40$ ms. These parameters will be the parameters of a generator modelling this source. This source is hence modelled by an independent generator (which is a constant bit rate generator).

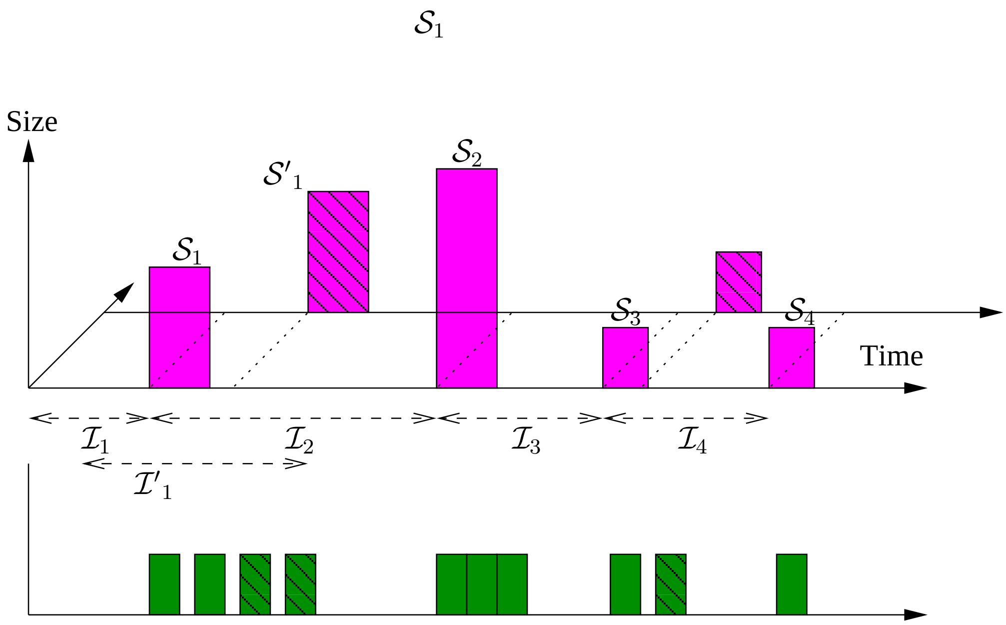


Figure 2.10.: Representation of a the aggregation of an abstract unit stream

2.2.2.2. Aggregation operation

The operation of aggregation is an important operation to describe the traffic hierarchy in our model.

Figure 2.10 shows the basic aggregation of abstract stream units. Two input traffic stream have been aggregated into one single traffic stream. The operation of aggregation has removed the single flow information. Moreover, the depicted figure shows that the size sequence and inter-arrival sequence has been deeply modified between the two streams present before the aggregation and after the aggregation. The operation can not be easily described because of the diverse process to take into consideration. It appears that the following elementary steps can be considered:

i) flow addition,
ii) level conversion,
iii) effects of entities and links on the transmission path (limited output capacity, buffer and scheduling policy implemented)

Our goal here is to obtain an operator able to characterize the modification that occurred on the traffic stream from one level to the other one.

The easiest characterization would be the aggregation of two abstract unit streams into one abstract stream.

Definition 2.2.18 (Aggregation operator)**.** *Given two abstract streams Φ_1 and Φ_2 considered at the level ℓ, the aggregated stream $\Phi = \Phi_1 \top \Phi_2$ at the level $\ell + 1$ is the result of the aggregation of flows Φ_1 and Φ_2.*

What are the properties of the operator $\top$? It is not commutative ($\Phi_1 \top \Phi_2 \neq \Phi_2 \top \Phi_1$ because one of the flows can be prioritized). The ”empty flow” which corresponds to no traffic is not such that $\Phi = \Phi_1 \top \Phi^{\emptyset} \neq \Phi_1$ because of the transformation related to the level conversion. In the following it will be written

$$\Phi = \Phi_1 \top \Phi_2 = \{(\Phi_1 \oplus \Phi_2)^{\triangle}\}^{\sqsupset} \tag{2.3}$$

where $\oplus, \triangle, \sqsupset$ are the elementary operators that are described hereafter (and stand respectively for addition of streams, level packet conversion and level conformation).

Addition of abstract unit streams If the aggregation of two traffic streams is considered, these streams have to be ”aggregable” which means that the total traffic inputs can be accommodated into the output. If storage is impossible, then the instantaneous units have to be smaller that the output capacity of the addition point in order to be added without the occurrence of packet losses; in this case the two streams are not "aggregable". It is assumed that the added flows conform to this criterion.

Definition 2.2.19 (Addition operator)**.** *Given two abstract unit streams Φ_1 and Φ_2 at the level ℓ, the summated stream $\Phi = \Phi_1 \oplus \Phi_2$ is the stream at the level ℓ sums of both traffic flow which consist the output of the level on the interface towards the next level.*

From two incoming abstract unit streams an output stream is constructed. Here, instantaneous addition is considered, buffers and priorities between streams (and other scheduling concerns or abstract unit discard cases) will be tackled later (the QoS scheme is described in 4.1.3, an opperator dealing with it is described in 4.2.3.1). Moreover, this addition is also studied at a constant level, which means no conversion of abstract units (abstract units are not fragmented or reassembled). With these strong hypothesises, the only concern is the capacity of the output.

The addition from two abstract unit streams $\Phi^{(1)} = \{\mathfrak{A}_i^{(1)}\} = \{(\mathcal{S}_i^{(1)}, \mathcal{I}_i^{(1)})\}$ and $\Phi^{(2)} = \{\mathfrak{A}_i^{(2)}\} = \{(\mathcal{S}_i^{(2)}\mathcal{I}_i^{(2)}\}$ into the abstract unit stream $\Phi = \Phi^{(1)} \oplus \Phi^{(2)} = \{\mathfrak{A}\}$ will be characterized for each following case:

Slotted addition It is assumed that for every time slot the output has a maximal slot capacity $S_{\max}$. For all i, $\mathcal{S}_i^{(1)} + \mathcal{S}_i^{(2)} \leq S_{\max}$ with

$$\mathfrak{A}_i = (\mathcal{S}_i, \mathcal{I}_i) \quad \text{where} \quad \mathcal{S}_i = \mathcal{S}_i^{(1)} + \mathcal{S}_i^{(2)}, \quad \mathcal{I}_i = \min(\mathcal{I}_i^{(1)}, \mathcal{I}_i^{(2)}) \tag{2.4}$$

At this level, information about the original streams has been lost. However for the performance evaluation, questions like where are the abstract units of flow $\Phi^{[(1)}$ located in the flow Φ will need to be answered.

Absolute case: In this case, it is more difficult to determine whether the two input streams can be added. First the procedure for the construction of the output is explained and then, its compatibility with the output link is examined.

Hereafter the absolute time sequence is needed which is build from the absolute time sequence from the streams Φ_1 and Φ_2 by $\{\mathcal{T}_i^{(1)} = \sum_{j=1}^{i} \mathcal{I}_j^{(1)}\}$ and $\{\mathcal{T}_i^{(2)} = \sum_{j=1}^{i} \mathcal{I}_j^{(2)}\}$, $\mathcal{T}_i$ is the ordered sequence of the union of the two sequences. The type of arrival at $\mathcal{T}_i$ is noted $\tau(i)$. Then for all i,

$$\mathfrak{A}_i = (\mathcal{S}_i, \mathcal{I}_i) \quad \text{where} \quad \mathcal{S}_i = \mathcal{S}_{i_{\tau(i)}}^{(\tau(i))} \quad \mathcal{I}_i = \mathcal{T}_i - \mathcal{I}_{i-1} \quad i_{\tau(i)} = \max_j(\mathcal{T}_j^{(\tau(i))} \leq \mathcal{T}_i) \tag{2.5}$$

If the addition is not performed instantaneously, more delicate addition operations have to be considered. If a buffer is implemented, some packets can be stored (if the output capacity is too small). Moreover a selection of the packets can also be performed in order to assure some quality of service constraints.

Level packet conversion

Definition 2.2.20 (Level conversion operator)**.** *Given one abstract unit stream $\Phi^{(\ell)}$ at the level ℓ, the stream at the level $\ell+1$ $\Phi^{(\ell+1)}$ such that*

$$\Phi^{(\ell+1)} = \Phi^{(\ell)^{\triangle_{\ell \to \ell+1}}}$$

is the converted stream of the input stream into the level $\ell+1$. The output stream is conveyed via $S_{\ell+1}$ packets.

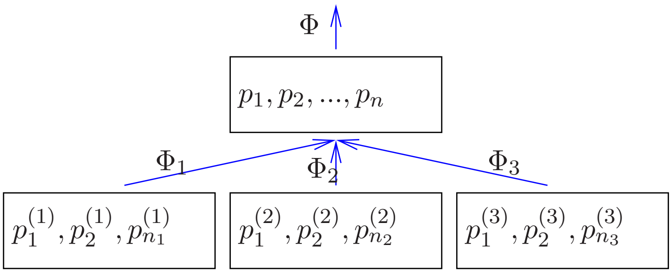


Figure 2.11.: Representation of a dependant generator (dependent on three independent generators)

Here we can say

$$(\Phi_1 \oplus \Phi_2)^{\triangle} = (\Phi_2 \oplus \Phi_1)^{\triangle} \tag{2.6}$$

because the level conversion operation is only related to the generation of packets in the upper level and independent of the inputs streams.

The conversion operator can follow different policies, that will be summarized hereafter in order to show what the operator is practically doing on the input flow

Inventory of conversion policies

- Fixed sized with fragmentation: the output packets have the same fixed size and are composed of an header and a fragment of the input packet. This results in an increased number of packets, but the packet size is constant.
- Header adddition without fragmentation: the output packets are composed of the complete input packet and an header.

Level conformation

Definition 2.2.21 (operator $\sqsupset$)**.** *The operator $\sqsupset$ is converting an input stream in an output stream compliant with the format required on the output.*

The format requirements can be for example a maximum consecutive number of packets on the output (following a parameter like the maximum burst size), a time spacing between the packets, a priority scheme on the packets.

The difficulty for modelling comes from the operator $\sqsupset$ which action on the traffic stream is deeply related to the level characteristics and that its modifications on the output stream are not easily traceable.

Another question arises when priorities between packets are considered to decide whether one or many outputs shall be considered. Considering one single output kept the further description (at least abstractly) simple and considering many output eases the handling of differentiated streams.

In order to approximate the action of the operator $\sqsupset$, a series of operations will be suggested for each level that describe the more important action on the streams due to the level conformation (refer to section 4.2).

2.2.2.3. Multi level traffic models ($\mathcal{P}_2, \mathcal{P}_3, \mathcal{P}_4$ models)

The models for $\mathcal{P}_2$ are firstly similar to $\mathcal{P}_1$ models in the sense that a description of the packet size and time of occurrence is required, so that a description of the abstract unit streams is necessary. What makes the models different is the dependency of the considered stream of the streams present at the application level.

For these reasons, let us introduce the following definitions.

Definition 2.2.22 (Dependent generator)**.** *This generator is dependent of other generators located at the lower level. Let n_g be the number of lower level generators. Each of these generators have an abstract unit stream Φ_k. A dependent generator is then a function*

$$G((p_1, \ldots, p_k, \Phi_1, ..., \Phi_{n_g}) = \Phi \tag{2.7}$$

with Φ is the output abstracts streams and $(p_1, \ldots, p_k)$ a list of parameters.

In the n_g lower level generators can be either dependent or independent generators. Figure 2.11 shows a dependent generator that is depending of three independent generators at one level lower.

Definition 2.2.23 (Equivalent model)**.** *An equivalent model is a independent generator having the same properties as a (multi-stage) dependent model*

The unanswered question is what are the ”same” properties of the abstract unit streams. Heuristically, the properties of the abstract streams are the same if the profile of packets transmission are kept when an equivalent is obtained. We will define this more precisely later by defining performance measures in order to have metrics to compare two abstract unit streams.

Definition 2.2.24 (Aggregation level)**.** *Considering one level of the pyramid (application, subscriber, node or transport level), the traffic aggregated up to this level represents all the traffic generated by the levels below this one.*

2.2.3. Mapping of protocols within the pyramid model

The OSI (Open Systems Interconnection) model is a reference model for network architecture developed by ISO (described in [Day95],[DZ83]) to describe the connection of open systems. It is divided in seven layers which are:

- the physical layer: concerned with the real transmission on the physical medium,
- the data link layer: it is concerned with the correct transmission of frame over the physical medium,
- the network layer: it is concerned with the routing of the packets from one side to the other side of the network,
- the transport layer: it is concerned with the provision of packet to the network layer and the determination of type of service to the sessions
- the session layer: it allows the establishment of session between users
- the presentation layer: it enables with a check of received data and of the management of appropriate data structure for an exchange with the application
- the application layer: it contains protocols that are needed by users.

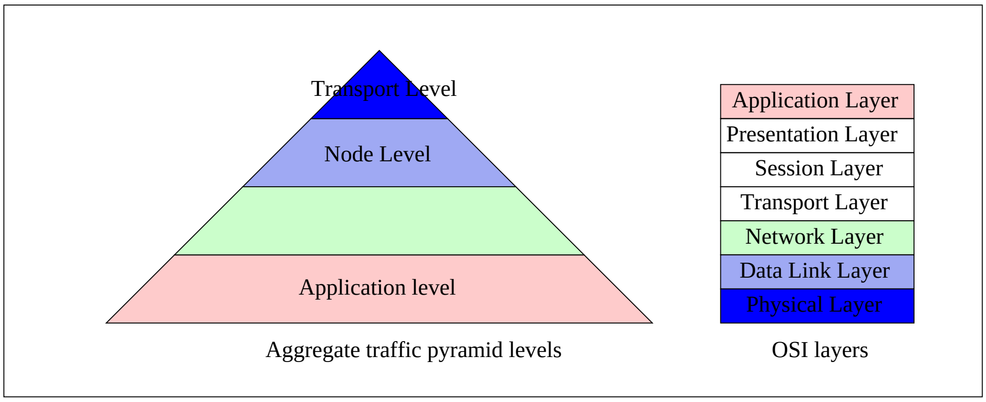


Figure 2.12.: Comparison between the pyramid levels and the OSI reference layers

Finally, the level 5 and 6 (session and presentation layers) have not such a major role and sometimes they are neglected as in [Tan96]. As we can see, the pyramid we have presented, has some analogies with the OSI reference model. For example, the terms ”transport” and ”application” are present in both models. But, in order to avoid confusion, the pyramid uses level instead on layer for the OSI reference model. Figure 2.12 presents the relationship between both models.

2.3. Performance evaluation

The communication system under consideration is supposed to be in an operating state so all performance concepts related to system recovery, faulty behavior, maintenance issues are not considered here.

2.3.1. Principle for stream comparison and metrics evaluation

2.3.1.1. Comparison of abstract streams units

In this section, the comparison of two abstract streams is investigated. It is assumed that two traffic streams have been measured at two reference points. The similitude between the two finite streams $\Phi_1 = \{\mathfrak{A}_1\}_{1,..,n_1}$ and $\Phi_2 = \{\mathfrak{A}_2\}_{1,..,n_2}$ need to be evaluated.

Definition 2.3.1 (Traffic comparator)**.** *A traffic comparator is providing a metric measuring the similitude between two traffic streams. A traffic comparator is then a function from Φ_1 and Φ_2*

$$TC(\Phi_1, \Phi_2) \geq 0 \tag{2.8}$$

The following properties can be observed:

$$TC(\Phi_1, \Phi_2) = TC(\Phi_2, \Phi_1) \tag{2.9}$$

and

$$TC(\Phi_1, \Phi_1) = 0 \tag{2.10}$$

So, if Φ_1 is similar to Φ_2, we should get $TC(\Phi_1, \Phi_2) \approx 0$ and otherwise in absence of similitude $TC(\Phi_1, \Phi_2) \gg 0$

2.3.1.2. Metrics: definition and properties

Definition 2.3.2 (Metric)**.** *A metric $\mathcal{M}$ is derived from an observable value that is evaluated from one or several entities in the network. An observable value is a time variable real value that is obtained from the abstract stream unit.*

This metric can be evaluated at:

- any time instant, it is an instantaneous metric $\mathcal{M}(t)$
- at sequential points of time, it is a sequential metric $\mathcal{M}_n$
- as a (temporal) average, it is an average metric $\overline{\mathcal{M}}$
- by its statistical distribution, it is a density metric $\mathcal{M}_{[m]}$ where m is a metric value.
- as its extremum value, it is a maximum or a minimum metric $\mathcal{M}_{\max}$ or $\mathcal{M}_{\min}$

The density metric $\mathcal{M}_{[m]}$ is the probability density of the values of m. If a sequence $\{m_1, .., m_k\}$ of the metric $\mathcal{M}$ have been collected, this density can be evaluated by an histogram on N_b interval bins mapping at least the interval $[\min_i(m_i), \max_i(m_i)]$. For every bin $[b_b(k), b_e(k)]$, the histogram takes the values $h(k) = \frac{\#\{i, m_i \in [b_b(k), b_e(k)], 1 \leq i \leq k\}}{k}$.

Of course, there exist some relationships between the different metrics that can be calculated. Figure 2.13 shows these relationships. For example with an instantaneous metric, it is possible to get the extremums or a sequential. Similarly from the density, it is possible to obtain the mean. The instantaneous

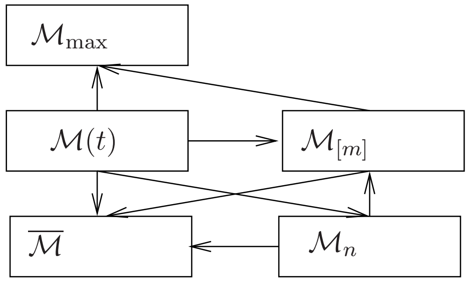


Figure 2.13.: Relationship between the different metrics

metric enables the calculation of every other metrics, but the evaluation of this metric requires a lot of storage capacity.

The physical point at which the metric can be evaluated is dependent of the location in the network where is was observed (or recorded). The following points of measurement are possible:

- on a link
  Metrics evaluated on a link are abstract stream metrics. They are providing information on the internal structure of the data packets carried by the link. Since the output of a generator is a link, these metrics can be used also for generators
- at two generator outputs (for model comparison)
  Metrics evaluated between two generators can be used for model comparison. They are monitoring ”how different” the outputs are.
- at two measurement points, located at the same level
  At these points, the metrics are related to performance evaluation because they access the difference that occurred during transmission.

The general metric $\mathcal{M}$ needs to be stated more precisely. Here are some possible observable values that can be evaluated and typical metric evaluation.

- Data Volume (instantaneous, average)
- Packets inter arrival time (distribution)
- Arrival Rate (average)
- Size-weighted arrival rate (average)
- Queueing statistics at double rate (statistical distribution)
- Delay, Delay jitter (maximum, average)
- Session length at receive side (maximal, average)

Metrics will be used to compare streams. If the models are equivalent, the metrics have similar values. Different evaluation of the difference can be conducted.

2.3.1.3. Metrics for stream comparison

The following comparators can be distinguished.

- integral comparator,
- difference comparator,
- statistical comparator

Integral comparator

$$\int (f(\Phi_1, t) - f(\Phi_2, t))^2 dt$$

The integral comparator can be used to compare function-valued metrics (for example the inter-arrival time distribution of two streams). It is inspired by the norm derived by the scalar product on the Hilbert space of real-valued functions of real variable that are squared-integrable. In practice, if two functions

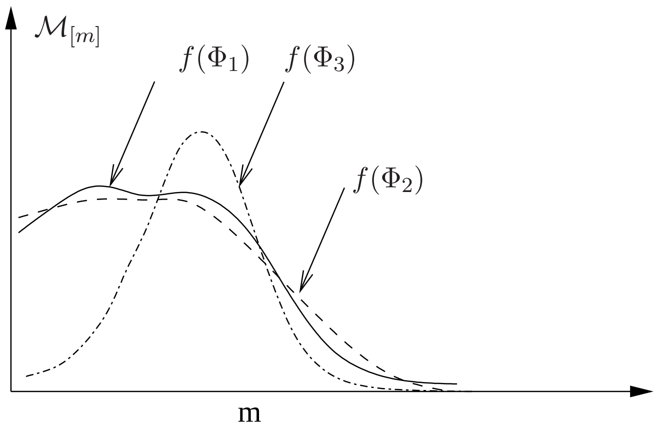


Figure 2.14.: Example of three metrics which are compared with the integral comparator

have a similar shape, the area of their squared-difference shall be small and if they mismatch, this area should be significative.

On figure 2.14 a distribution metric derived from three different abstract unit streams are compared. The integral comparator can be used to say that $f(\Phi_1) \approx f(\Phi_2)$ and that $f(\Phi_1) \neq f(\Phi_3)$

Difference comparator

$$f(\Phi_1, t) - f(\Phi_2, t))$$

The difference comparator can be used to compare scalar-valued metrics. For example, two average metrics can be compared by the difference comparator.

Statistical comparator From the series of values of one metric, a 95% or a 99% confidence interval can be derived. If another metric has a range within this interval, the values can be assumed to be equal. It is assumed that two traffic streams have been measured at two reference points. The similitude between the two finite streams $\Phi_1 = \{\mathfrak{A}_1\}_{1,..,n_1}$ and $\Phi_2 = \{\mathfrak{A}_2\}_{1,..,n_2}$ need to be evaluated.

2.3.2. Metrics available in the pyramid model

2.3.2.1. Metrics derived from a single abstract unit stream

Given an abstract unit stream Φ, metrics from the abstract unit stream can be derived, for example from a finite sequence of abstract unit collected during a test period T.

Data volume evaluation The observable value considered here is the volume of data is $V(t) = \sum_{i=1}^{N(t)} \mathcal{S}_i$. It is an increasing function defined an a continuous time base. In the slotted approach, the sequence V_n can be obtained similarly.

Definition 2.3.3 (Metric data volume $\mathcal{M}_V(t)$)**.** *The data volume is a instantaneous metric obtained by summing the size of all the packets seen in the interval* $[0, t]$. *From an abstract stream unit* $\Phi = \{\mathfrak{A}_i\} = \{(\mathcal{S}, \mathcal{I})\}$ *the volume is the partial sum of the sequence* $\{\mathcal{S}\}$

Figure 2.15 represents $\mathcal{M}_V(t)$ for a given stream unit. The volume is located on the lower graph and is the sum of the sizes of the packets represented on the upper graph.

Data rate evaluation The ratio over time of the data volume is the data rate.

Definition 2.3.4 (Metric data rate $\mathcal{M}_{Rn}$)**.** *The data rate is evaluated with a resolution* T_R *on a corresponding time scale and is the ratio of the packets size over the slot duration.*

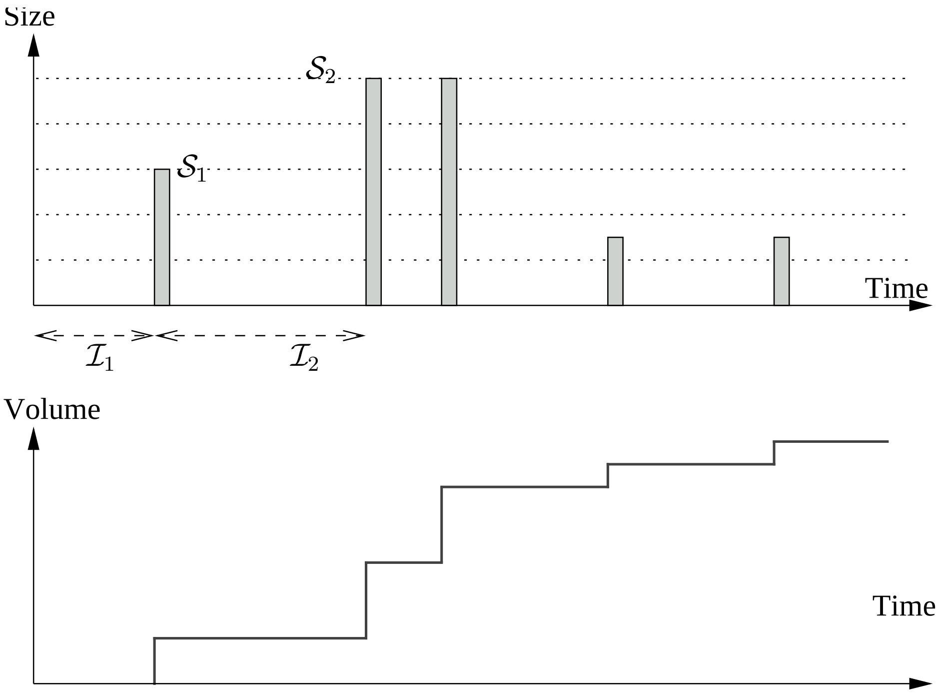


Figure 2.15.: Data Volume metric

Definition 2.3.5 (Metric Cumulated rate $\mathcal{M}_{R_{\textbf{cum}}}$)**.** *The cumulated data rate is derived from the data volume by the following formula*

$$\mathcal{M}_{R\textbf{\textit{cum}}}(t) = \frac{\int_0^t V(t)dt}{t} \tag{2.11}$$

Definition 2.3.6 (Metric Inter-arrival time statistic $\mathcal{M}_{IAT[m]}$)**.** *The Iat is a density metric obtained by the collection of the inter arrivals sequence* $\{\mathcal{I}\}$

Arrival rate and similar metrics The observable value is the rate of arrival of the packets, which in queueing theory is called λ, it is measured from the inverse of inter-arrival time.

Definition 2.3.7. *Arrival Rate metric* $\overline{\mathcal{M}_{AR}}$ *is an average metric obtained from the evaluation of the new packets arrival during a period of time.*

If packets of multiples sizes or time synchronous packet arrivals are considered, the definition of the rate needs to be precisely defined. Inter-arrival of duration 0 have to be avoided. A first approach is to consider all arrivals as a single arrival of a batch of packets. In this case, the arrival rate of batches is considered and a probability of batch sizes is defined $P(z) = \sum_{b=1}^{N} \Pr(\mathcal{S} = b) z^b$ where b is the batch size varying from 1 to n. This point will be tackled again when batch traffic models will be introduced. Queueing properties of M/M/1 queues with batch arrivals can be easily computed.

Combined metrics If the information present in both series $\{\mathcal{S}\}$ and $\{\mathcal{I}\}$ present in $\Psi = \{(\mathcal{S}, \mathcal{I})\}$ is used, the metric is combining the information. Hence, following metric can be defined.

Definition 2.3.8 (Metric Throughput $\mathcal{M}_{T_n}$)**.** *The throughput is a metric evaluating the amount of bits that have been transported over a stream. It is evaluated at regular basis.*

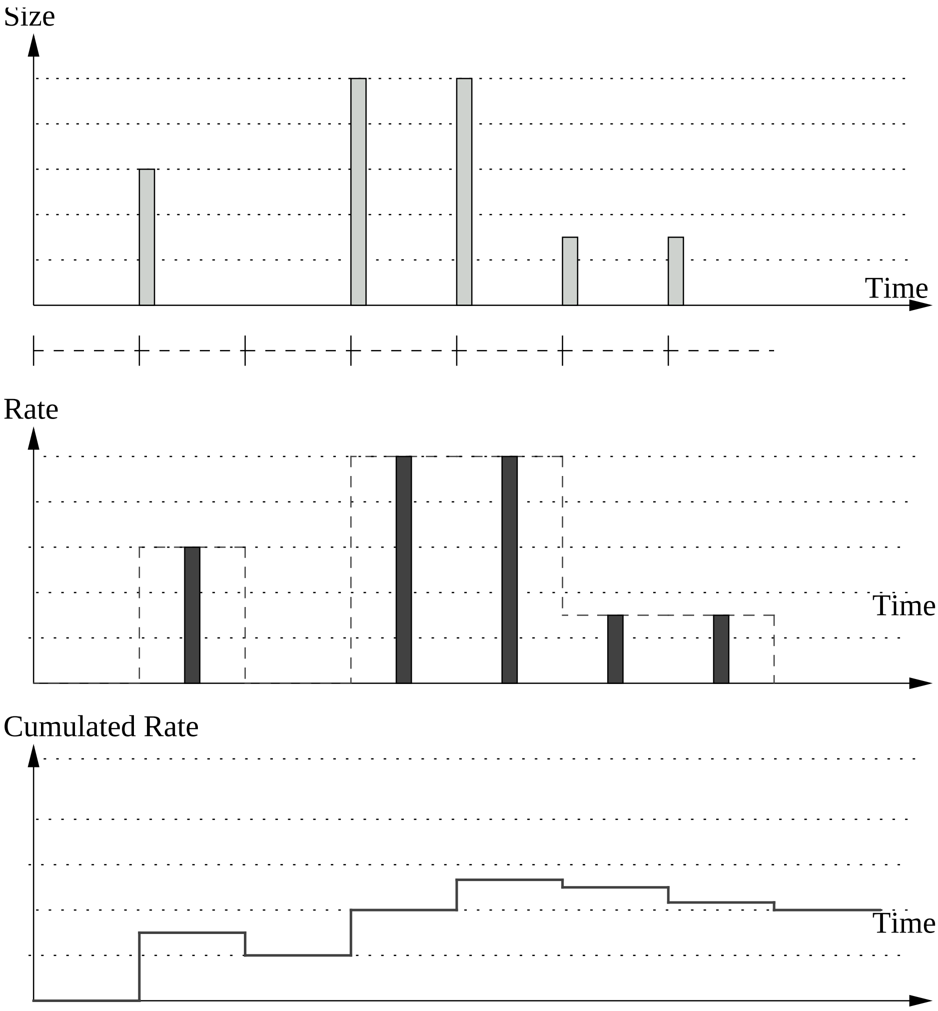


Figure 2.16.: Data Rate metric

Queuing Metrics Queueing metrics furnish interesting metrics for a traffic flow. They are obtained by assuming that incoming packets are stored in a common waiting room and are served with a particular policy. If the queues do not correspond to a real physical queue, it can be assumed that the queue is served with a rate $\mu^{-1} = 2.\lambda$ (where μ is the inverse of the mean service time and λ is the mean arrival rate) constituting a double rate queue. This value is interesting for obtaining queue size statistics because the queue size is kept under control (when $\rho = \frac{\lambda}{\mu}$ tends to 1, the queue size explodes) and the queue is not constantly empty that occurs when the load is too low.

Queuing metrics related to the queue size can be evaluated at different instant of time : i) at the arrival of a new packet in the waiting room (at arrivals) ii) when a packets departs from the queue in order to be serviced (at departure) iii) at randomly sampled instants of time (arbitrary instant).

Figure 2.17 represents the principles of a metric related to queueing properties. In the depicted period, five packets arrivals have occurred with respective size 3, 5, 5, 1.5 and 1.5. The queue size at packet departure is monitoring the number of packets in the queue at each packet departure from the queue to enter into service. If the queue is empty at one packet arrival (as at the first and second arrival), a first work unit can be serviced immediately. The queue size at departure is then the number of arrived packets minus the one served immediately. When no arrival occurs, the queue size is decremented after each packet service. The service queue of one packet is straightforwardly indicated on the picture.

Definition 2.3.9 (Metric Queue size distribution at packet departure $\mathcal{M}_{Q_d[m]}$)**.** *Given a queue with load ρ, the queue size distribution is the number of packet remaining in the queue*

Definition 2.3.10 (Metric stationary Queueing time at rate ρ $\mathcal{M}_{W(t)]}$)**.** *The queueing time is the time that*

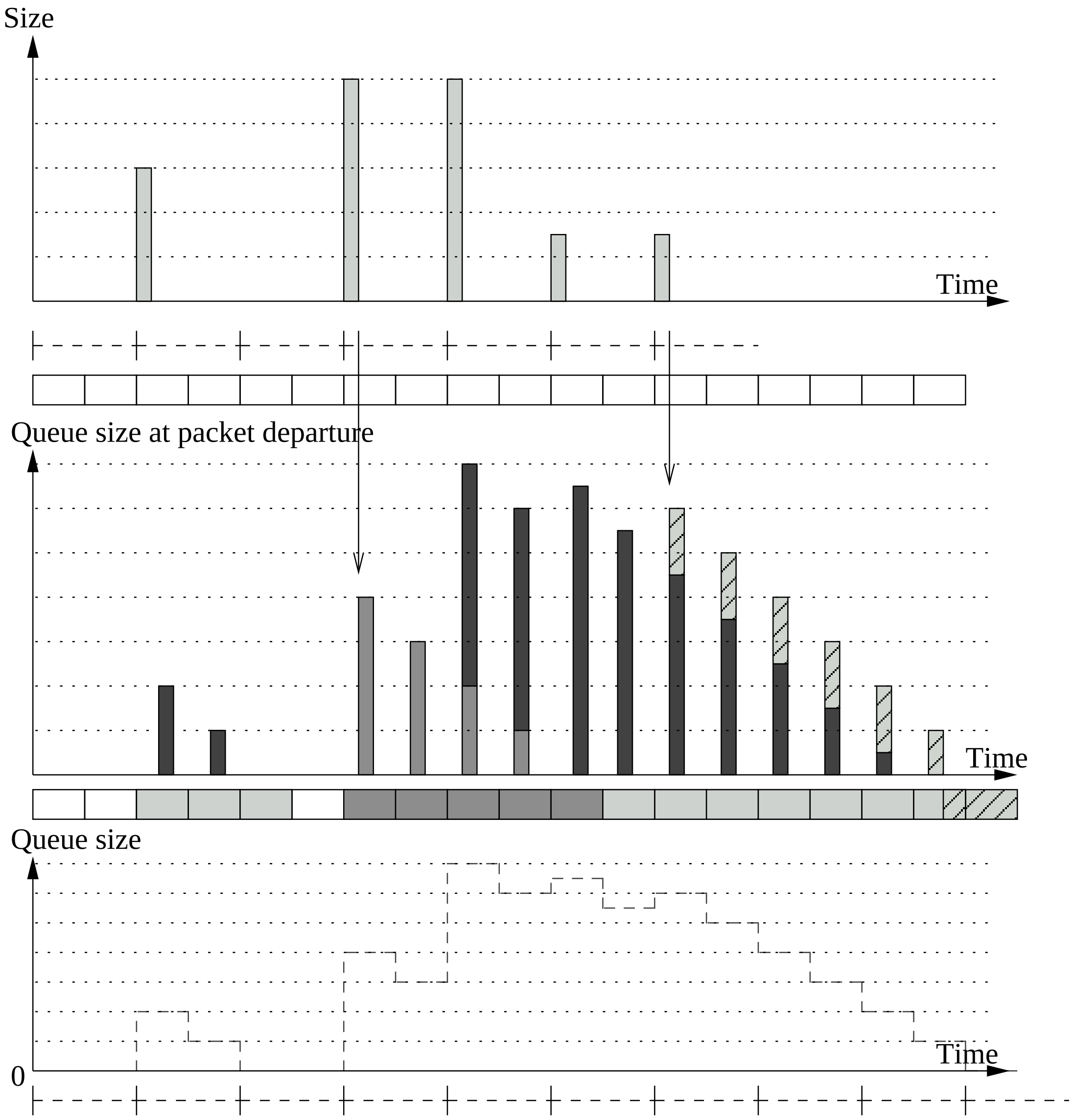


Figure 2.17.: Queueing metrics: packet arrivals and size, queue occupancy at packet departures, stationary queue length

a packet arriving at t will have to wait before to be serviced.

2.3.2.2. Packet oriented metrics

Relationship between packets

Definition 2.3.11 (Relationship between packets)**.** *Given two abstracts units $\mathfrak{A}^{(1)}$ and $\mathfrak{A}^{(2)}$ measured at two reference points r_1 and r_2 with r_1 is before r_2 in the transmission path,*
*$\mathfrak{A}^{(2)}$ is a **fragment** of $\mathfrak{A}^{(1)}$ if the part of the information conveyed in $\mathfrak{A}^{(1)}$ is contained in $\mathfrak{A}^{(2)}$ and it is noted $\mathfrak{A}^{(1)} \triangleleft \mathfrak{A}^{(2)} = 1$ otherwise $\mathfrak{A}^{(1)} \triangleleft \mathfrak{A}^{(2)} = 0$ if the information in both abstact units is not related*
*$\mathfrak{A}^{(1)}$ is an **origin atom** of $\mathfrak{A}^{(1)}$ if part of the information conveyed in $\mathfrak{A}^{(2)}$ was originally contained in $\mathfrak{A}^{(1)}$ and it is noted $\mathfrak{A}^{(1)} \triangleright \mathfrak{A}^{(2)} = 1$ otherwise if the information in both abstact units is not related $\mathfrak{A}^{(1)} \triangleright \mathfrak{A}^{(2)} = 0$*

Definition 2.3.12 (Originating and destination maps)**.** *Given two abstracts flows streams Φ_1 and Φ_2 of size (n_1 and n_2), the originating map is a function that associates to an abstract unit of Φ_2 the set of abstract unit(s) of Φ_1 which are the origin of the abstract unit (or the set of all the origin atoms of the*

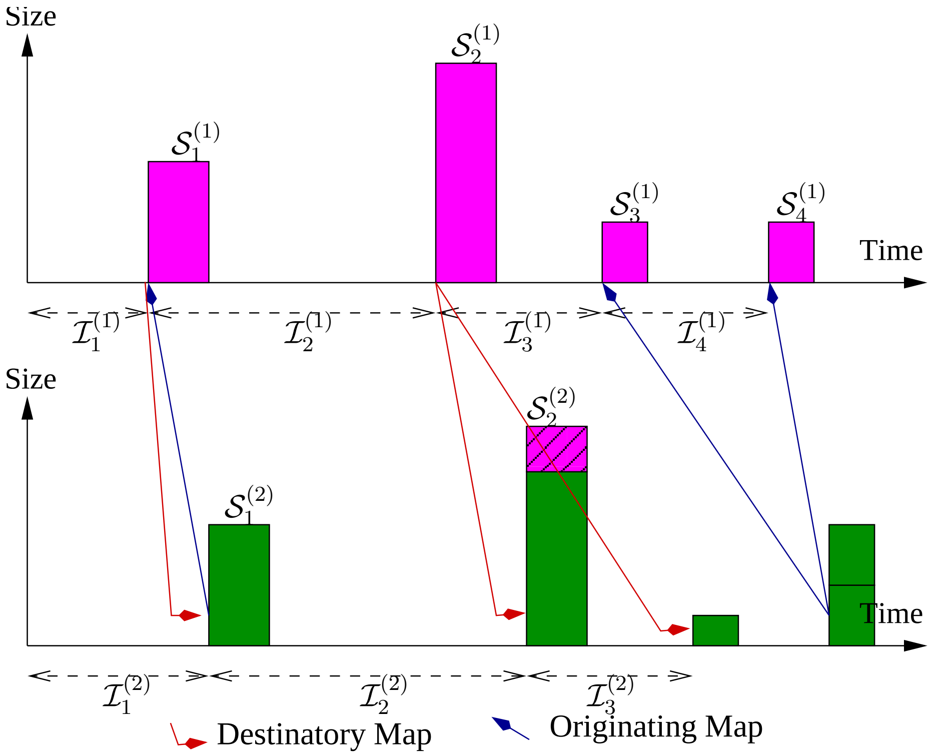


Figure 2.18.: Exemplary originating and destination maps

abstract unit). The originating map is noted

$$\mathcal{O}(\mathfrak{A}_j^{(2)}) = \{\mathfrak{A}_i^{(1)}, \mathfrak{A}_i^{(1)} \in \Phi_1, \mathfrak{A}^{(1)} \lhd \mathfrak{A}^{(2)} = 1\} \textit{ for } \mathfrak{A}_j^{(2)} \in \Phi_2 \tag{2.12}$$

the destination map is a function that associates to an abstract unit of Φ_1 the set of abstract unit(s) of Φ_2 which are correspond to the transmission of the abstract unit (or the set of all the fragment of the abstract unit). The destination map is noted

$$\mathcal{D}(\mathfrak{A}_i^{(1)}) = \{\mathfrak{A}_j^{(2)}, \mathfrak{A}_j \in \Phi_2, \mathfrak{A}_i^{(1)} \rhd \mathfrak{A}_j^{(2)} = 1\} \textit{for} \mathfrak{A}_i^{(1)} \in \Phi_1 \tag{2.13}$$

The destination map could be empty ($\emptyset$), on the contrary the originating map should not.

On figure 2.18, the originating and destination maps are depicted. In this case, the maps for the first four packets have the following values:

$$\begin{array}{llll}
\mathcal{O}(\mathfrak{A}_1^{(2)}) = \{\mathfrak{A}_1^{(1)}\} & \mathcal{O}(\mathfrak{A}_2^{(2)}) = \{\mathfrak{A}_2^{(1)}\} & \mathcal{O}(\mathfrak{A}_3^{(2)}) = \{\mathfrak{A}_2^{(1)}\} & \mathcal{O}(\mathfrak{A}_4^{(2)}) = \{\mathfrak{A}_3^{(1)}, \mathfrak{A}_4^{(1)}\} \\
\mathcal{D}(\mathfrak{A}_1^{(1)}) = \{\mathfrak{A}_1^{(2)}\} & \mathcal{D}(\mathfrak{A}_2^{(1)}) = \{\mathfrak{A}_2^{(2)}, \mathfrak{A}_3^{(2)}\} & \mathcal{D}(\mathfrak{A}_3^{(1)}) = \{\mathfrak{A}_4^{(2)}\} & \mathcal{D}(\mathfrak{A}_4^{(1)}) = \{\mathfrak{A}_4^{(2)}\}
\end{array}$$

The size of the abstract units could have been modified, because of overhead that is added during transmission. This will happen in particular if the traffic streams are considered at different levels.

Losses

Definition 2.3.13 (Flow Packet Losses)**.** *Given two abstracts flows streams Φ_1 and Φ_2, measured at the SAME level ℓ. Every abstract unit of Φ_2, $\mathfrak{A}_j = (\mathcal{S}_j, \mathcal{I}_j)$ shall have a corresponding originating abstract unit $\mathcal{O}(j)$. The missing abstract units ($\mathcal{C}_{\mathbb{N}}\mathcal{O}(j), j = 1..n_2$) have been lost during transmission.*

Delay Delay can be easily defined when the transmission is observed at the same levels for the emission and the reception. For each correctly transmitted packet it is possible to calculate the experimented delay by this packet. The delay is related to the evaluation of the transmission time in the system.

Definition 2.3.14 (Packet Delay)**.** *the delay is the measure of the time difference between the packet seen in the first reference point and in the second reference point.*

$$\delta_j = \sum_{k=1}^{\tau(j)} \mathcal{I}_k^{(2)} - \sum_{k=1}^{\tau(j)} \mathcal{I}_k^{(1)} \quad j = 1..n_2 \tag{2.14}$$

If the packet is not received, the delay can not be evaluated. Such packets shall be accounted in the lost packets, as explained earlier.

2.3.3. Performance evaluation oriented metrics

2.3.3.1. ITU recommendations

In the document [IT95b] some guidelines are given for evaluating the performance of an ATM link supporting B-ISDN services. These recommendations could be easily ”translated” for every type of transport level.

Classification of transferred cells At two reference points r_1 and r_2 with r_1 is before r_2 in the transmission path and located at the **same level** ($\ell = 4$), two abstracts units $\mathfrak{A}^{(1)}$ and $\mathfrak{A}^{(2)}$ are considered.

Successful cell transfer outcome A successful cell transfer outcome occurs when at the reference point r_2 a cell $\mathfrak{A}_j^{(2)} = (\mathcal{S}_j^{(2)}, \mathcal{I}_j^{(2)})$ with a corresponding cell $\mathfrak{A}_i^{(1)} = (\mathcal{S}_i^{(1)}, \mathcal{I}_i^{(1)})$ at the reference point r_1 (such that $\mathfrak{A}_i^{(1)} \triangleright \mathfrak{A}_j^{(2)} = \mathfrak{A}_i^{(1)} \triangleleft \mathfrak{A}_j^{(2)} = 1$) is received with a delay smaller that T_{max} and the content of the received cell is identical to the transmitted cell and the header of the received cell is valid.

Errored cell outcome An errored cell outcome occurs when at the reference point r_2 a cell $\mathfrak{A}_j^{(2)} = (\mathcal{S}_j^{(2)}, \mathcal{I}_j^{(2)})$ with a corresponding cell $\mathfrak{A}_i^{(1)} = (\mathcal{S}_i^{(1)}, \mathcal{I}_i^{(1)})$ at the reference point r_1 (such that $\mathfrak{A}_i^{(1)} \triangleright \mathfrak{A}_j^{(2)} = \mathfrak{A}_i^{(1)} \triangleleft \mathfrak{A}_j^{(2)} = 1$) is received with a delay smaller that T_{max} but the content of the received cell is different from the transmitted cell content and the header of the received cell is indicating the presence of errors.

Lost cell outcome A lost cell outcome occurs when a cell $\mathfrak{A}_i^{(1)} = (\mathcal{S}_i^{(1)}, \mathcal{I}_i^{(1)})$ emitted at the reference point r_1 is not received at the reference point r_2 within T_{max} ($\forall j \in \Phi^{(2)} \mathfrak{A}_i^{(1)} \triangleright \mathfrak{A}_j^{(2)} = 0$ where $\Phi^{(2)}$ is the abstract stream unit measured at r_2 in the interval $[0, \tau_i + T_{max}]$ and τ_i the time of emission of $\mathfrak{A}_i^{(1)}$)

Misinserted cell outcome A misinserted cell outcome occurs when a cell $\mathfrak{A}_j^{(2)} = (\mathcal{S}_j^{(2)}, \mathcal{I}_j^{(2)})$ received at the reference point r_2 had no corresponding emitted cells at the reference point r_1 ($\forall i \in \Phi^{(1)} \mathfrak{A}_i^{(1)} \triangleleft \mathfrak{A}_j^{(2)} = 0$ where $\Phi^{(1)}$ is the abstract stream unit measured at r_1 in the interval $[0, \tau_j]$ and τ_j the time of emission of $\mathfrak{A}_j^{(2)}$)

Performance parameters

Cell error ratio The cell error ratio is the ratio of total errored cells to the total of successfully transferred and errored cells.

Cell loss ratio (CLR) The cell loss ratio is the ratio of total lost cells to the total of successfully transferred cells.

Cell transfer delay CTD (average cell transfer delay)

$$CTD = t_r - t_s$$

where t_r: receive time relative to the synchronized time reference when the cell reached the Test Cell Output side. t_s: transmit time relative to the synchronized time reference when the cell left the Test Cell Input side.

Note: CTD is one of the cell transfer performance parameter. It very much depends on the cell traffic in the test traffic, the controlled traffic and the real traffic as such, as well as on the background traffic.

Mean cell transfer delay (MCTD) The Mean CTD is the arithmetic average (mean) of the CTD measured over the time period T_1. $MCTD(T_1) = \frac{\sum cdt}{a}$ a where a: number of received and correct $A'_c ells$ (no payload CRC fault: correct test cells) cdt: Summation of the CDT (tr - ts) of all correct A'_{cells}.

Maximum (resp. minimum) cell transfer delay (CTDmax, CTDmin) The maximum CTDmax(resp. minimum CTDmin) is the maximum (resp. minimum) of the CTD measured over the time period T_1.

Cell delay variation (CDV) Two types of CDV can be defined (1-point and 2-points CDV) depending wether it is measured at one or two reference points.

- *1-point CDV:* The 1 point CDV is measured at a singe reference point r. The negotiated peak cell rate is required $\frac{1}{T_P}$. An expected time of arrival is defined for each cell by $c_k = k \times T_P$.
- *2-points CDV:* The 2 point CDV is measured as the difference in the sequence of the cell delays.

The cell delay variation is the maximum cell transfer delay minus the minimum cell transfer delay measured over the time period t1. CDV=CTDmax-CTDmin CDTmax= maximum cell transfer delay CDTmin= minimum cell transfer delay

Transfer delay jitter The transfer delay jitter can be seen as the average variation of the CTD (Cell Transfer Delay) around the mean cell transfer delay(MCTD). Theoretically it is represented by the variance of the random variable CTD. Using the mathematical formula:

$$J = \mathbb{E}[(CTD - \mathbb{E}[CTD])^2] = \mathbb{E}[CTD^2] - \mathbb{E}[CTD]^2$$

where $\mathbb{E}[]$: is the expectation of a random variable. The transfer delay jitter can be computed by the difference between the mean square cell delay transfer and the square cell delay transfer.

Evaluation method

- **Cell Error Ratio (CER)** A continuous test cell stream containing N_1 is sent at the first reference point. At the second first reference point the received cells are analyzed during a time interval of T. CER is calculated as follows: CER $=\frac{N_e}{N_2}$ where N_2 is the number of received test cells and N_e is the number of errored cells in these N_2 received cells. This method is only valid in the absence of misinserted cells.
- **Cell Loss Ratio (CLR)** A continuous test cell stream containing N_1 cells is sent at the first reference point. At the second reference point N_2 cells are analyzed during a time interval of duration T. CLR is calculated as follows: $CLR(T) = \frac{(N_1 - N_2)}{N_1}$ where N_1 is the number of sent cells and N_2 the number of received cells. This method is only valid in the absence of misinserted cells.

Connection Level Performance metrics

End-to-End Call Blocking Probability Number of end-to-end call blocked/Number total of end-to-end call

Throughput Number of bits that can be transferred per unit of time over the network

Connection Set-Up Delay Time elapsed between the connection demand and the connection acknowledgement

Connection Release Delay Time elapsed between the connection release demand and the connection release acknowledgement

2.3.3.2. Link with Quality of service

QoS is defined in the ITU Recommendation E.800 ([IT93]) as follows: Collective effect of service performances which determine the degree of satisfaction of a user of the service. The note of Recommendation E.800 underlines that the QOS is characterized by to the combined aspects of: service support and service operability performance; and servability and service integrity performance. The definition of Quality of Service in Recommendation E.800 is a wide one encompassing many areas of work, including subjective customer satisfaction. However, within this Recommendation the aspects of Quality of Service that are covered are restricted to the identification of parameters that can be directly observed and measured at the point at which the service is accessed by the user. Other types of QoS parameters which are subjective in nature, i.e. depend upon user actions or subjective opinions, will not be specified in the I-Series Recommendations on QoS (like [IT95a]).

2.3.4. Performance evaluation on the different levels of the pyramid

The evaluation of performance metrics can be required at different levels. The first case is related to the transmission and the reception of packets with the origin and destination located at the same level. The second case corresponds to an evaluation considering different levels. Most problems occur in the second case, because different packets entities are considered.

2.3.4.1. Intra-level performance

The problem investigated here is concerned with the derivation of performance metrics from traffic stream recorded at the same level, then the identification of particular stream inside a particular stream.

Stream performance measures When two traffic streams are considered, $\{\mathcal{I}^1, \mathcal{S}^1\}$ and $\{\mathcal{I}^2, \mathcal{S}^2\}$, for every received packet in the second stream, the delay can be calculated as

$$\delta_i = \sum_{k=1}^{i} \mathcal{I}_k^2 - \sum_{k=1}^{i'} \mathcal{I}_k^1$$

where i' is the index of the emitted cell in the origin stream. In the absolute time reference, the delay is the difference between the transmitted and received absolute time.

Sub-stream in a stream The identification of a particular set of cells inside an abstract unit stream can be performed using the information in the header. The different type of streams that are distinguished can be:

-) a single application (for example to identify a particular application),
-) a kind of application (to identify a particular type of traffic)
-) a kind of cell type (to identify a particular type of cells (for example a specific QoS type)).

2.3.4.2. Cross level performance

Typical questions that need to be answered in cross level approaches are for example: Given a initial stream $\Phi^{(1)}$ and a measured AUS Φ^4 what are the performance of application ?

It is not possible to compare two streams at different levels, because the cells format is different. For example if 10 packets $\mathcal{P}_1$ are packed in one $\mathcal{P}_2$ packet, it is not possible to measure the delay between each of the single $\mathcal{P}_1$ and the $\mathcal{P}_2$ packets.

In order to be able to perform comparison of two cells collected at level ℓ_1 and ℓ_2, two procedure are possible: 1) from the cells located at level ℓ_1, an equivalent generator of cells at level ℓ_2. This is possible if $\ell_1 > \ell_2$. 2) from the cells at level ℓ_2 the further (virtual) reception is build till to level ℓ_1 with the minimal assumptions. For this it is necessary that $\ell_2 > \ell_1$. Each procedure is then bringing the problem to the comparison of two streams at the same level.

The measured metric will then be metrics valid only if the level conversion has not modified the structure of the structure of the streams.

In the simulator, an inverted pyramid is present in order to get an output stream until level $\ell = 1$. Moreover, most traffic generator will generate a traffic stream located at level $\ell = 1$, even if they were constructed from measurements collected at another level.

2.4. Summary

In this chapter, the aggregate traffic pyramid was introduced. The four levels (application, subscriber, node and transport level) enable a hierarchical description of the traffic generated in a telecommunication system that follows the topological organization of the networks. In the following, the application to an aeronautical satellite communication system will get a particular focus: the relevance of the hierarchical model will be confirmed and some numerical results of simulations for this case will be used to justify equivalencies of traffic models. With the help of the pyramid approach, traffic streams $\Phi = (\mathcal{S}, \mathcal{I})$ can be described at all levels. In the next chapter, the traffic models are introduced, that reproduce the streams at the lowest level (sources models) and that can be used at another level to reproduce the overall level traffic (equivalent models).

Thou, nature, art my goddess; to thy ***law***
My ***services*** *are bound. Wherefore should I*
Stand in the plague of custom, and permit
The curiosity of nations to deprive me,

Shakespaere, King Lear I 2

3

Traffic models in the pyramid model

3.1. Single stream traffic models

3.1.1. Mathematical models for independent generators

In the previous chapter, the independent generators were introduced. At this point, the following definition of a traffic model is given.

Definition 3.1.1 (Traffic model)**.** *A traffic model is a description of the data packets (abstract units) conveyed across a reference point.*

This definition requires to describe packet size, time of transmission and type of content. In general, the content of the packet is not relevant for traffic models. It is then necessary to model the size of the packets $\mathcal{S}$ and the time between packets (inter-arrival time) $\mathcal{I}$. The traffic models can be classified as follow:

- Deterministic / Stochastic: if the size and the inter-arrival time follow a deterministic pattern or if some randomness is introduced for example by the use of statistical distribution
- Slotted / Absolutely timed: if the time reference is based on slots, the traffic model reduce to a description of the content of each slot (occupancy of slots, number of packets in the slots) or if the time is referenced absolutely, each packet conveyed over the reference point can be dated.
- Fixed size / Variable size: if the size of the packets is assumed constant, the abstract unit stream is described by the inter-arrival time $\mathcal{I}$, otherwise it is a variable size model that need to be modelled by a random variable $\mathcal{S}$.

All traffic models are either deterministic or stochastic and slotted or absolutely timed. Some mathematical objects useful to represent traffic model will now be highlighted. The point process is a theoretical process that can be used for stochastic, absolutely timed and fixed-size traffic models. Traffic models are not concerned with the content of the packets. The content would only be necessary to model errors occurring during transmission

3.1.1.1. Point process and renewal process

The theory of point processes is exposed in [Cox67]. Renewal process are useful to model the distribution between events called arrivals (at the origin, they were used to model life duration of components in a system where immediate replacement occurs). If arrivals corresponds to packet arrivals, then a first traffic model can be derived.

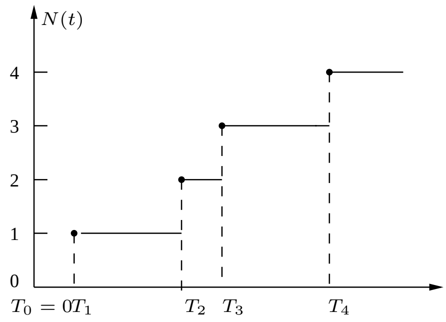


Figure 3.1.: Example of point process

Definitions The theoretical process able to describe the generation process is called a point process:

Definition 3.1.2 (Point process)**.** *A simple point process $\psi = \{T_n : n \geq 0\}$ is a collection of arrival instants $0 = T_0 < T_1 < T_2 <$ ... with $T_n \to \infty$ as $n \to \infty$. Defining $N(0) \triangleq 0$, $N(t)$ is the number of points in the interval $(0, t]$. $\{N(t)\ : t \geq 0\}$ is called the counting process for ψ. If T_n are random variable then ψ is called a random point process. $\mathcal{I}_n = T_n - T_{n-1}, n \geq 1$ is called the nth inter-arrival time.*

The point process was called simple because only one arrival is allowed at the same time.

More information on renewal process is presented in Appendix A.1.1. Point process and renewal process (and their derivatives) are good tools to model packet of constant size in an absolute timing reference. If the size of the packets need to be modelled more accurately, other models will be needed.

3.1.1.2. The PH-renewal process

The following distribution is useful because it fits many distributions and eases many calculations. It was introduced by [Neu75] and [Neu78]. The PH-renewal process is based on PH-distributions which are constructed using the following Markov process on the states $\{1, \ldots, m+1\}$ with an infinitesimal generator

$$Q_{\text{PH}} = \begin{vmatrix} T & \boldsymbol{T}^0 \\ \mathbf{0} & 0 \end{vmatrix} \tag{3.1}$$

where the $m \times m$ matrix T satisfies $T_{ii} < 0$ for $1 \leq i \leq m$ and $T_{ij} > 0$ for $i \neq j$ and $T\mathbf{e}+\boldsymbol{T}^0 = \mathbf{0}$. The initial probability vector of Q is given by $(\boldsymbol{\alpha}, \alpha_{m+1})$ with $\boldsymbol{\alpha}\mathrm{e} + \alpha_{m+1} = 1$. States $1, \ldots, m$ are transients and the state $m+1$ is absorbing. The probability distribution $F(.)$ of the time until absorption in the state $m+1$, corresponding to the initial probability vector $(\boldsymbol{\alpha}, \alpha_{m+1})$ is given by

$$F(x) = 1 - \boldsymbol{\alpha} \exp{(Tx)}\mathbf{e} \qquad \text{for} x \geq 0. \tag{3.2}$$

Definition 3.1.3 (PH-distribution)**.** *A probability distribution $F(.)$ on $[0, \infty]$ is a distribution of phase type (PH-distribution) if it is the distribution of the time until absorption in a finite Markov process of the type defined in 3.1. The pair (α, T) is called a representation of $F()$.*

Some properties of this distribution are presented in Appendix A.1.2.

3.1.1.3. Markov renewal processes

Markov renewal process were introduced by [Cin69]. Some results are recalled in Appendix A.1.3.

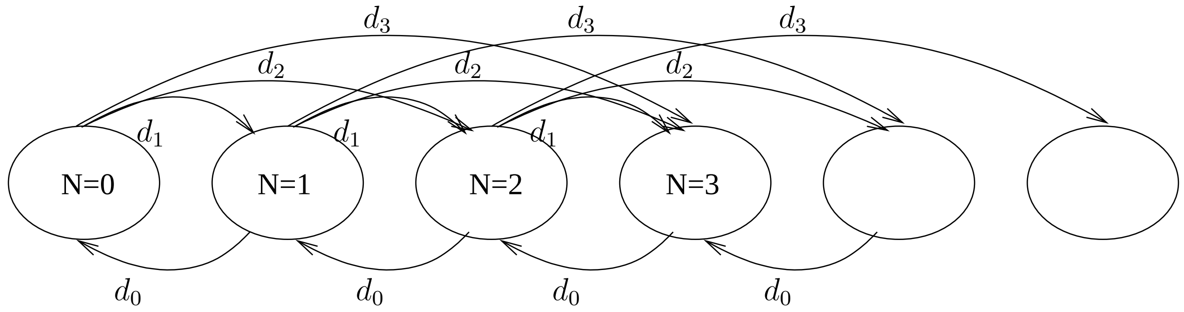


Figure 3.2.: Example of a poisson process with batch arrivals

3.1.2. State based models

3.1.2.1. The batch Markovian arrival process

The versatile Markovian point process was introduced in [Neu76]. The *Batch Markovian Arrival Process* was shown to be equivalent to Neuts' point process, with a more transparent notation.

Definition Consider a Poisson process with batch arrivals. The rate of the Poisson process is λ and batch size probability is p_j for $j \geq 1$. Let $N(t)$ be the counting process. Then the process $\{N(t)\}$ is a Markov process on the state space $\{i; i \geq 0\}$ with the infinitesimal generator Q

$$Q = \begin{bmatrix} d_0 & d_1 & d_2 & d_3 & \cdots \\ & d_0 & d_1 & d_2 & \cdots \\ & & d_0 & d_1 & \cdots \\ & & & . & \cdots \\ & & & & \cdots \end{bmatrix} \tag{3.3}$$

where $d_0 = -\lambda$ (probability of non arrival) and $d_i = \lambda p_j$ for $j \geq 1$ (probability of an arrival of a batch of size j). Figure 3.2 is representing an example of a poisson process with arrivals at rate λ.

The BMAP is a generalization of the previous batch Poisson process to allow non exponential times between the arrivals of the batches.

Definition 3.1.4 (Batch Markovian Arrival Process)**.** *A 2-dimensional Markov process (Continuous Time Markov Process) $\{N(t), J(t)\}$ on the state space $\{(i,j) \ : i \geq 0, 1 \leq j \leq m\}$ with the infinitesimal generator Q having the following structure*

$$Q = \begin{bmatrix} D_0 & D_1 & D_2 & D_3 & \cdots \\ & D_0 & D_1 & D_2 & \cdots \\ & & D_0 & D_1 & \cdots \\ & & & . & \cdots \\ & & & & \cdots \end{bmatrix} \tag{3.4}$$

$N(t)$ is the counting process and $J(t)$ is the phase of the process. It is required that D_0 has negative diagonal elements and non-negative off-diagonal elements, D_k are non-negative, and that $D = \sum_{k=0}^{\infty} D_k$ is an irreducible infinitesimal generator.

The BMAP/G/1 queue The BMAP/G/1 queue can be solved using matrix geometric methods. In [Luc91] the results are presented, some of them are recalled in annex C. The key is the computation of the matrix G (related to arrivals in busy period) which enables the estimation of the moments of the queue size (at departure, at arbitrary size) and of the virtual waiting time can be easily obtained. For this reason, the queueing properties of BMAP generators will be checked,

If the batch arrivals size is widely spread, very large storage space will be needed for the complete solution (storage of the A_n matrices), which can limit the precision of obtained distribution.

As a conclusion, the BMAP traffic is an important class with powerful modelling capacity. The drawback of this class is the number of parameters required, that are more reduced for the next models.

3.1.2.2. The Markov Modulated Poisson Process

The MMPP is a particular case of BMAP with an infinitesimal generator Q and arrival rate matrix $\Lambda = diag(\lambda_1, ..., \lambda_m)$. The parameters characterizing an MMPP are a m-state continuous Markov chain with infinitesimal generator Q and the m-Poisson arrival rates $\lambda_1, ..., \lambda_m$. All arrivals have the same size (no batch arrivals). In the notations from the BMAP we then have :

$$D_0 = Q - \Lambda \qquad D_1 = \Lambda \qquad D_k = 0,\ k \geq 2 \tag{3.5}$$

The stationary probability vector $\boldsymbol{\pi}$ of the Markov process with generator Q verifies

$$\boldsymbol{\pi} Q = \mathbf{0} \qquad \boldsymbol{\pi}\mathbf{e} = 1 \tag{3.6}$$

The state of the underlying chain is $J(t)$ at t. If the MMPP is not a renewal process, the sequence $\{(J_n, X_n), n \geq 0\}$ where J_0 is the state of $J(t)$ at $t = 0$, $X_0 = 0$ and for the kth arrivals J_k is the state of the underlying process and X_k is the time between the $k-1$th and kth arrival is Markov renewal sequence with the following transition probability:

$$\begin{aligned} F(x) &= \int_0^x \exp\left[(Q-\Lambda)u\right] du\, \Lambda \\ &= \{I - e^{(Q-\Lambda)x}\}(Q-\Lambda)^{-1}\Lambda \end{aligned}$$

The element of the matrix F_{ij} are the conditional probabilities $\Pr\{J_k = j, X_k \leq x \,|\, J_{k-1} = i\}$.

The matrix $F(\infty) = (\Lambda - Q)^{-1}\Lambda$ is stochastic and is the transition probability of the Markov chain embedded at arrivals epochs.

Even if MMPP models are just modelling one size of packets, this model forms a compact class that don't have too many parameters and made them suitable for different method for model parameter derivation.

3.1.2.3. Circulant MMPP

Definition A sub-category of MMPP traffic models can be introduced by considering the family of models where the $N \times N$ matrix Q has the first raw of the form

$$\mathbf{a} = [a_o\, a_1\, \ldots\, a_{N-1}]$$

with $a_0 = -\sum_{k=1}^{N-1} a_k$ and the other rows are obtained by circulating the raw of one element to the right. The matrix Q is can be written with the $N \times N$ permutation matrix P:

$$P = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & \\ \vdots & & & \ddots & \\ 0 & \cdots & & 0 & 1 \\ 1 & 0 & \ldots & 0 & 0 \end{pmatrix} \quad Q = \begin{pmatrix} a_0 & a_1 & a_2 & \cdots & a_{N-1} \\ a_{N-1} & a_0 & a_1 & \cdots & a_{N-2} \\ \vdots & a_{N-1} & \ddots & \ddots & \\ a_2 & \cdots & & a_0 & a_1 \\ a_1 & & \cdots & a_{N-1} & a_0 \end{pmatrix}$$

The circulant MMPP have tractable properties for the calculation the power spectrum. The properties of additivity of the power spectrum for aggregated model is an appealing property. Practical problems were encountered for the implementation of parameters derivation procedure (the constraint minimization procedure, to be used was giving incoherent results), so that these models were not used in the numerical investigation. The order of the model to have circulant form is in the range of 50, 100 (with many nul parameters), so that the handling of the parameters is a bit more difficult than for the MMPP(2).

3.2. Considered generators in the pyramid

The traffic streams generators are the elements generating traffic in the pyramid model. The models to be used are particularized for some models that fit well the original sources.

3.2.1. Application level

A generator at the application level will need to consider the type of service that is considered.

3.2.1.1. Voice model

The model considers an alternating process of talk and silence and was exposed in [HL86]. The burstiness of the traffic has its origin in these alternating periods. A mean talking duration α^{-1} of 351 ms and a mean silence duration of β^{-1}= 649 ms have been considered. During the talk period, the source emits packets at fixed intervals of length T. In the case of ISDN telephony at a data rate of 64 kbit/s, we have $T = 6,625$ ms. The mean number of packets sent during the active taking phase is $\overline{N} = \alpha^{-1}/T = 53$. This model uses the following law for the arrival between packet :

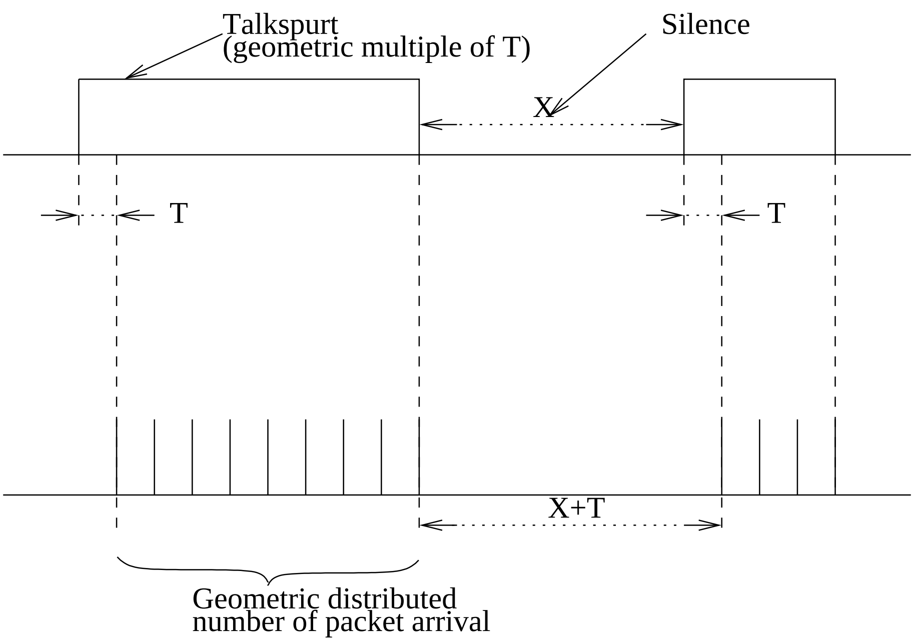


Figure 3.3.: Model for a voice active source

$$f(\lambda_t = t) = p\delta_T(t) + (1-p)\beta e^{-\beta(t-T)} 1_{\mathbb{R}+}(t-T)$$

In this work, the following parameters have been used
- mean ON duration $\lambda_{\text{ON}}^{-1} = 0.350s$ (as in [HL86])
- mean OFF duration $\lambda_{\text{OFF}}^{-1} = 0.630s$ (as in [HL86])
- T=0.01s (corresponding to the aggregation of 5 GSM packets from a simular codec working at 12k)
- the packet size $\mathcal{S}_1$ is 800 bits .

The aggregation of a number N_{vo} of active voice sources can be modelled by an unique global model. In [HL86] a 2 states MMPP (Markov Modulated Poisson Process) is used to capture the behavior of the global traffic. In [KOS97] the model was used for single voice model and an MMPP(2) equivalent model was derived. Measurement from [Bra68] have confirmed the model.

Table 3.1 shows the parameters of some voice codecs and the size of the generated packets. There is the possibility to aggregate a certain amount of packets together to reduce header importance by putting

Codec Name	Size of Data packets	Slot duration	Aggregation level
GSM	256 bits (13.2 kbps)	20 ms	1-5
G.711	20 bytes (64 kbps)	5ms	1-10

Table 3.1.: Voice-codecs

together the data of many time slots into a single data packet. The selected size for $\mathcal{S}_1$ corresponds to the size of the G.711 voice codec, with an aggregation factor of 5.

3.2.1.2. Video model

Models with constant bit rate (CBR) have been used with 500 kbps, in addition with ON/OFF process for audio with a CBR of 64 kbps. In [GW94] a model for VBR video has been developed. The number of bytes sent per frame ($T = 1/24$ s) is modelled by a fractional ARIMA model corrected by a Gamma/Pareto distribution. The study was based on a 2 hours Star Wars movie trace. The estimation of the autocorrelation function (ACF) was an evidence of long range dependence. A stationary process X_t is called process with long range dependence (LRD) if the limit of the correlations can be written $\lim_{x\to\infty}\rho(k) = c_\rho k^{-\alpha}$ with $\alpha \in [0,1]$ [Ber94]. A (0,d,0) fractional ARIMA process was used to model such a process. The f-ARIMA has the following correlation:

$$\rho(k) = \frac{\Gamma(1-d)\Gamma(k+d)}{\Gamma(d)\Gamma(k+1-d)} \sim \frac{\Gamma(1-d)}{\Gamma(d)}|k|^{2*d-1}\ .$$

The process can be generated using Hosking's algorithm ([Hos84]). The generation involves the generation of a sample sequence $x_0, x_1, \ldots, x_{n-1}$ of size n from a stationary process with a normal marginal distribution and correlation function ρ_k Step 1: a starting value x_0 from the stationary distribution of the process $\mathcal{N}(0,\nu_0)$ is generated where ν_0 is the required variance of the x_t. Set $N_0 = 0, D_0 = 0$.
Step 2: For $t = 1, \ldots, n-1$, calculate ϕ_{tj}, $j = 1, \ldots, t$ recursively via the equations :

$$\begin{aligned} N_t &= \rho_t - \sum_{j=1}^{t-1}\phi_{t-1,j}\rho_{t-j} \\ D_t &= D_{t-1} - {N_{t-1}}^2/D_{t-1} \\ \phi_{tt} &= N_t/D_t \end{aligned}$$

$$\phi_{tj} = \phi_{(t-1)j} - \phi_{tt}\phi_{t-1,t-j} j = 1, \ldots, t-1$$

Calculate $m_t = \sum_{j=1}^{t}\phi_{tj}x_{t-j}$ and $\nu_t = (1-{\phi_{tt}}^2).\nu_{t-1}$. The sample x_t is generated from the distribution $\mathcal{N}(m_t,\nu_t)$. For ARIMA(0,d,0) process the algorithm may be simplified by the fact that $\phi_{tt} = d/(t-d)$.

Others traces (for example the one studied in [FM98]) have also long range dependency but are using a different probability distribution. Moreover, the distribution of the required bandwidth per frame exhibits an important probability for the tail of the distribution. Heavy tailed distribution have to be used as the suggested Gamma/Pareto distributions. It is build with the following densities:

$$F_{\Gamma/P} = \begin{cases} F_\Gamma(x) & \text{if } x \le x^* \\ F_P(x) & \text{if } x > x^* \end{cases}$$

where F_Γ is the cumulative distribution of the Gamma probability f_Γ

$$f_\Gamma(x) = \frac{w^s}{\Gamma(s)}x^{s-1}e^{-wx}, \quad x \ge 0$$

with s and w being the shape and scale parameters and $\Gamma(s)$ is the Gamma function and F_P is the cdf of the Pareto distribution :

$$F_P = \begin{cases} 1-(\frac{a}{x})^\alpha & \text{if } x \ge a \\ 0 & \text{if } x < a \end{cases}$$

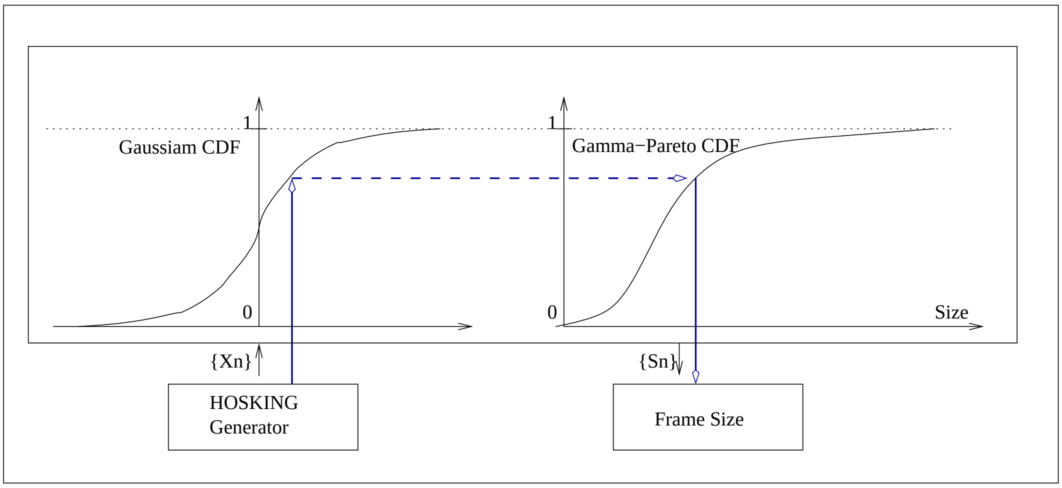


Figure 3.4.: The generation of video sizes samples

with a starting point of the Pareto distribution and α characterizing the slope. The figure 3.8 shows how the two sections of the distribution fits together. The fractional ARIMA is used to model the variability of the traffic. The self similarity parameter (Hurst parameters) H as estimated to be around 0.8 ± 0.088 for this trace. The Pareto/Gamma is used to shape the distribution with the required heavy tailed behavior.

The two important characteristics of video traffic can be combined using the method highlighted in figure 3.4. The samples of the Hosking sequence are mapped into the gaussian distribution, and then using the inverse of the Gamma-Pareto distribution the samples for the video size are obtained. The obtained sequence has of course the Gamma-Pareto distribution and has the long range dependency property too.

3.2.1.3. Web Model

Model Parameters The ETSI model [ETS98] considers a WWW browsing session, that consists of a sequence of packet calls. During a packet call, several packets can be generated. So that a bursty sequence of packets is generated.

In a www session, a packet call corresponds to the download of a www document. When the document is received, the user content of the page is analyzed. The www model has then the following parameters.

- Session arrival process
- Number of packet calls per session, N_{pc}^{sess}
- Reading time between packet calls, R_{pc}. The reading time starts when when the last datagram of the packet call is completely received by the user. The reading time ends when the user makes a request for the next packet call.
- Number of datagrams within a packet call, N_{dtg}^{pc}
- Inter arrival time between datagrams $\mathcal{I}_d$
- Size of a datagram, S_d.

These parameters are defined at different level as shown in figure 3.5. The model suggests the use of the following statistical distributions for modelling www browsing:

- for the session arrival rate, Poisson arrivals for the begin of session,
- for the number of packet call per session N_{pc}^{sess} a geometrical distribution with mean $\mu_{N_{pc}^{sess}}$,
- for the reading time (time between last page packet reception and next packet emission), an exponential distribution that can be approximated by a geometrical distribution with mean μ_{pc}^{R}.
- for the number of datagrams in a packet call (N_{dtg}^{pc}), a geometrical distribution with mean $\mu_{N_{dtg}^{pc}}$ can be used if it fits well to case under study. Alternatively the transmission of a single ”big” packet can be considered, in this case $N_d = 1$
- for the inter-arrival time between datagrams $\mathcal{I}_d$, an exponential distribution which is approached

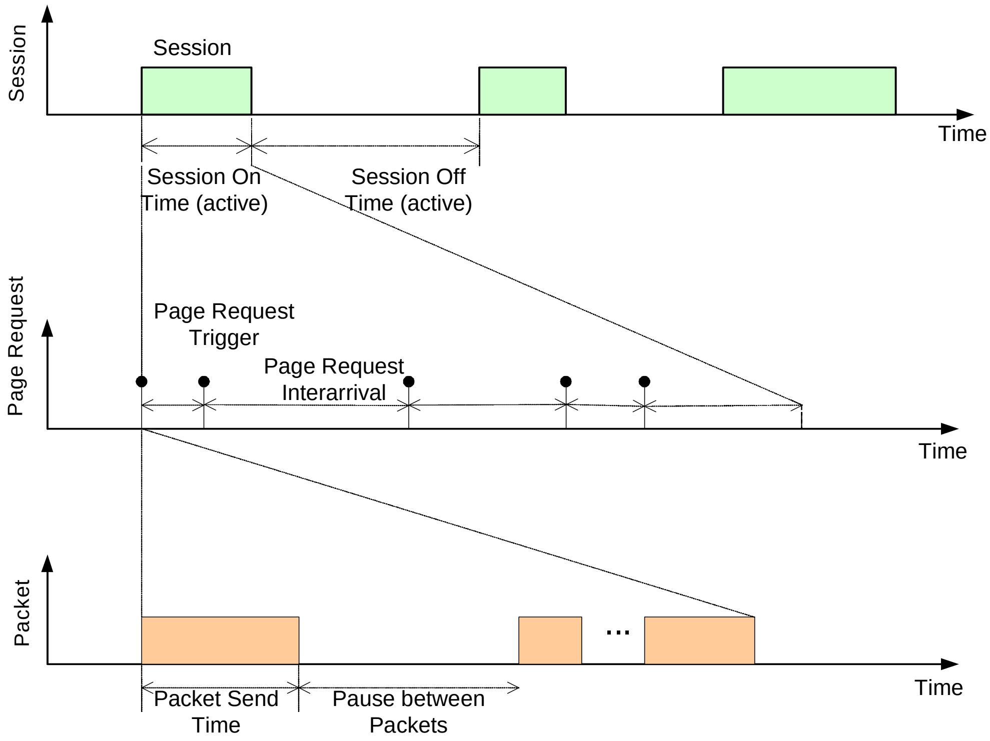


Figure 3.5.: Definition of the ETSI model

by a geometrical distribution.

- for the datagram size $\mathcal{S}_d$, a Pareto distribution with cut-off.

The document gives also some default numerical values for the parameters of the distribution. For the number of packet calls in a session a geometric distribution with mean $\mu_{N_{pc}^{sess}} = 5$; for the reading time between packet calls a geometric distribution with mean $\mu_{pc}^{R} = 412s$; for the number of datagrams in a packet call a geometric distribution with mean $\mu_{N_{dtg}^{pc}} = 25$; for the packet size a Pareto distribution with $\alpha = 1.1$ $k = 81.5$ $bytes$ with a cut off at 66666 $bytes$ (the mean size is then 480 $bytes$) are respectively suggested.

The literature dealing with IP traffic models, and web traffic in particular, can not be exhaustively cited. Every research team has developed its own model. In [SLTG00], the parameters from different web models have been compared. In particular the models from [KL97] and [CL99b] are compared because of the Pareto distribution (that is ”heavy-tailed”).

DLR also developed a web model based on recording from single browser activity. A set of distribution was chosen and fitted with the recording of logs from different users collected during 10 days. An complete list of the used distributions and their parameters can be found in [BG05b]. When multiple instances of the model are multiplexed together, the resulting trace exhibits self similarity as it is expected from a realistic web traffic.

BMAP model In addition to the previous modelling, BMAP models for web were investigated like in [KLL03] who have proved that they provide superior results as Poisson and MMPP models. Moreover, these models are compatible with the other selected models.

The web model that will be used in the following will be a BMAP model. First the values proposed by [KLL03] will be used. As a comparison, the parameters for a web model from the DLR measurements will be presented and the one from the WirelessCabin test fly recording.

It has to be noted that the BMAP is (even if it has distinct states) a ”continuous” model without

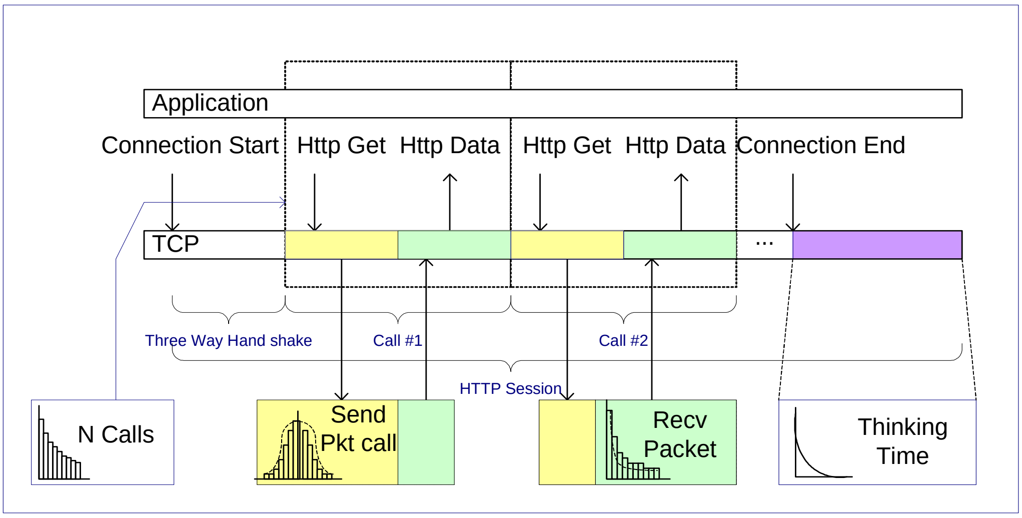


Figure 3.6.: The web model developed by DLR

consideration of the period of inactivity. For this reason, the author of the previous study have used measurement on high loaded trunk, in order to prevent too big inter arrival times in the recording. On the contrary, during the capture of web sessions on a few individual browsers investigated by DLR, silent phases were measured between the pages download and between the session. If these measurements are used to derive BMAP parameters, the inter-arrivals time could be considered only inside each web browsing session.

Session duration estimation One of the issue that occurs, when a BMAP model is used, is that the information on the session duration is lost because only the packets are modelled without protocol information. Nevertheless, it could be of interest to get information on session duration

DLR organized in the framework of the WirelessCabin project a test flight where provision of GSM services and other web based services were demonstrated. This flight was performed from Blagnac Airport near Toulouse, in September 13th, 2004 with an Airbus A340-600 with few passengers with maximum 10 users enjoying the new services ([JMBH$^+$05]). The recordings of a capture performed with `ethereal` during the whole flight duration were analyzed (of a duration of 11756 s - 3h15m56s). 706848 packets were collected in total. These packets were recorded with an average transmission rate of 60,124 packets per second. From them, 94941 were filtered as HTTP packets (13%). The transmission rate is then of 8 packets per s. For the HTTP traffic, the average packet size is of 990.73 bytes.

Afterward, the trace was further investigated with the `tcptrace` tool that is able to process each HTTP connections. The number of connections can get very high because every data transfer with a server is accounted as a connection. For these connections, an estimation of the data throuput can be obtained and the variation of the rtt (round trip time) can be observed.

Figure 3.7 shows the data throughput collected for one of the connection during the test flight.

3.2.2. Subscriber level and node level

There are no models particular for a subscriber since the traffic generated by the user is determined by the applications in use at the application level. In the case of single application subscribers, subscriber models can be straightforwardly derived, but do not correspond to any simplifications.

In this study, the number of applications at subscriber level will be kept small. If it is really needed to have an high number of applications for a subscriber, the subscriber can be decomposed in many different virtual subscribers with an reduced number of applications, then it could be of interest to have aggregate equivalent. In the numerical study only up to two applications per subscribers were considered.

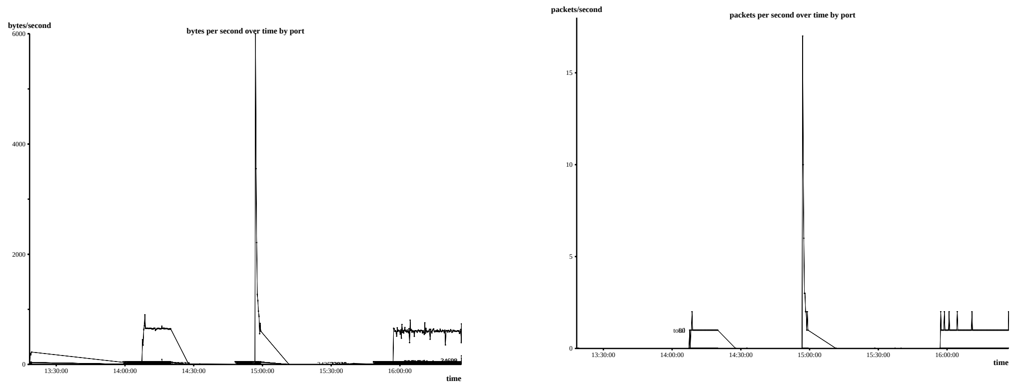


Figure 3.7.: Recording of the HTPP from WirelessCabin test flight (in size and volume)

More gain can be obtained at the next level i.e. the advantages will be evidenced clearer at the node level that at the subscriber level. At this node level, the advantage of aggregate models is more stringent because the number of models the aggregate model is replacing is higher. The number of applications and subscriber replaced leaves some margin for more complex models.

The candidate models for aggregate models are the MMPP and the BMAP. The MMPP and BMAP models revealed as being easy to implement in a simulator. The most computational effort is required in the initial phase for the computation of the state transition probabilities and for the stationary state probabilities. After these calculations have been performed, at each state transition, the next state remains to be calculated. If it correspond to a packet emission, a packet (of the corresponding size if applicable) is sent.

Moreover, dedicated routines have been written to derive BMAP and MMPP parameters from the traffic traces recorded. In particular, a C program has been developed to derive model parameters using the EM algorithm. Having it in C makes the execution fast, even if an increase of the number of records or of the required precision for convergence will reduce the speed.

3.2.3. Transport level

Since the transport level is the upmost level, some general models could be used to describe the level of occupancy. The level of abstraction is maximum and the number of details if minimum, a synthetic and well known distribution has been used.

For example the number of occupied traffic slots in the frame can be modelled by a gaussian law (described in appendix D). It is straightforward to determine the parameter from the mean and variance of the recorded traffic. To take into account the different slots category, a multi dimensional model is required. When two classes of QoS are considered, a two-dimensional model can be used. If (X,Y) are gaussian random values, the density is then:

$$f_{(X,Y)}(x,y) = \frac{1}{2\pi\sqrt{\det\Sigma}} \exp[-\frac{1}{2}((x,y)-\mu)\Sigma^{-1}((x,y)-\mu)^t]$$

where μ is the vector of the mean and Σ the covariance matrix of X, Y

In the case under consideration, the load of the satellite transport level is built on a regular frame basis. Each frame is accepting packets with different type of QoS. When two classes are considered, the number of occupied cell in each category for every frame is the series $\{N_1, N_2\}$ that is modelled with a two dimensional gaussian law.

Simulation of gaussian random values In order to simulate one or two normal law, the method from Box-Muller can be used to obtain them from uniform random values.

If U_1 et U_2 are independent uniform random values on $[0, 1]$, then

$$X_1 = \sqrt{(-2\log(U_1))} \times \cos(2\pi U_2) \tag{3.7}$$
$$X_2 = \sqrt{(-2\log(U_1))} \times \sin(2\pi U_2) \tag{3.8}$$

Then X_1 and X_2 are two normal centered standard distributions.

Then using the Cholesky factorization of the covariance matrix $\Sigma = LL^t$, them $\mu + LX$ is following the normal distribution of $\mathbf{N}(\mu, \Sigma)$.

The gaussian models are then furnishing a basis for light-weight models, that can be used at high level. This is a rich family of models, that represents synthetically the measured traffic.

3.3. Determination of parameters for aggregated models

In this chapter, the methods to obtain the aggregated models from the models considered at the lower levels are considered. The first case considers derivation of the parameter of the MMPP(2) model using statistics from the aggregation of the source models, the second case considers a second method that furnish more accurate queueing results. The derivation of parameters for video models is also explained and, finally, the EM method is explained.

3.3.1. Determination of MMPP(2) parameters with counts statistics

This method is exposed in [HL86]. A MMPP with arrival matrix Λ and Q is considered.

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \qquad Q = \begin{pmatrix} -r_1 & r_1 \\ r_2 & -r_2 \end{pmatrix} \tag{3.9}$$

The mean arrival rate in an interval can be evaluated from an arrival trace. The theoretical value is

$$\mathbb{E}(N_t) = \frac{\lambda_1 r_2 + \lambda_2 r_1}{r_1 + r_2} t \tag{3.10}$$

Similarly for the variance, we have

$$\mathrm{var}(N_t) = \left(1 + \frac{2(\lambda_1 - \lambda_2)^2 r_1 r_2}{(r_1 + r_2)^2(\lambda_1 r_2 + \lambda_2 r_1)}\right) \times \frac{\lambda_1 r_2 + \lambda_2 r_1}{r_1 + r_2} t - \frac{2(\lambda_1 - \lambda_2)^2 r_1 r_2}{(r_1 + r_2)^4}(1 - e^{-(r_1+r_2)t}) \tag{3.11}$$

The index of dispersion of counts from the simulated data was calculated (IDC $= \frac{\mathrm{var}(N_t)}{\mathbb{E}(N_t)}$, defined in 3.1.1.1)

For the fitting proposed by Heffes, the evaluation of the third moment is also required:

$$\mathbb{E}[(N_t - \overline{N_t})^3] = \frac{\partial^3 g(z,t)}{\partial z^3}\Big|_{z=1} - 3\overline{N_t}(\overline{N_t} - 1)\frac{\mathrm{var}(N_t)}{\overline{N_t}} - \overline{N_t}(\overline{N_t} - 1)(\overline{N_t} - 2) \tag{3.12}$$

with

$$\frac{\partial^3 g(z,t)}{\partial z^3}\Big|_{z=1} = \frac{6}{r_1 + r_2}[\frac{A_{11}}{6} t^3 + \frac{A_{21}}{2} t^2 + A_{31} t + A_{12} t e^{-(r_1+r_2)t} + A_{41}(1 - e^{-(r_1+r_2)t})] \tag{3.13}$$

where

$$A_{11} = \frac{(\lambda_1 r_2 + \lambda_2 r_1)^3}{(r_1+r_2)^2} \tag{3.14}$$

$$A_{21} = \frac{2r_1 r_2(\lambda_1-\lambda_2)^2(\lambda_1 r_2 + \lambda_2 r_1)}{(r_1+r_2)^3} \tag{3.15}$$

$$A_{31} = \frac{r_1 r_2(\lambda_1-\lambda_2)^2[\lambda_1 r_1 + \lambda_2 r_2 - 2(\lambda_1 r_2 + \lambda_2 r_1)]}{(r_1+r_2)^4} \tag{3.16}$$

$$A_{41} = \frac{-2r_1 r_2(\lambda_1-\lambda_2)^3(r_1 - r_2)}{(r_1+r_2)^5} \tag{3.17}$$

$$A_{12} = \frac{r_1 r_2(\lambda_1-\lambda_2)^2(\lambda_1 r_2 + \lambda_2 r_1)}{(r_1+r_2)^4} \tag{3.18}$$

The parameters are derived using the method described by Heffes. To determine the parameters, the mean arrival rate is evaluated, the IDC is evaluated for t small. The method hence requires to evaluate:

1) the mean arrival rate
2) the variance-to-mean ratio of the number of arrivals in $(0, t_1)$
3) the long term variance-to-mean ratio of the number of arrivals (or the limit of the k-squared coefficient of variation of the intervals)
4) the third moment of the number of arrivals in $(0, t_2)$.

All these parameters can be estimated from a trace with the exception of the long-term of $I(t)$. (An estimation close to the origin can be provided, but the greater the time, the more unprecise it becomes). [HL86] derives it from the single model in an aggregated cases. Otherwise, the following formula can be used:

$$\lim_{k\to\infty} c_k^2 = \frac{\mathrm{var}(\mathcal{I}) + 2\frac{\mathrm{cov}(\mathcal{I}_1,\mathcal{I}_2)}{1-\rho}}{\mathbb{E}(\mathcal{I})} \tag{3.19}$$

where it is assumed that $\mathrm{cov}(\mathcal{I}_1, \mathcal{I}_{1+j}) = \mathrm{cov}(\mathcal{I}_1, \mathcal{I}_2)\rho^{j-1}$. The derivation of 3.19 is detailed in Annex C.12 for an MMPP(2) traffic model. Using the following notations,

$$a = \frac{\mathbb{E}(N_t)}{t} \tag{3.20}$$

$$b_t = \frac{\mathrm{var}(N_t)}{\mathbb{E}(N_t)} \tag{3.21}$$

$$b_\infty = \lim_{t\to\infty} b_t \tag{3.22}$$

$$\mu^3(t) = \mathbb{E}[(N_t - \overline{N_t})^3] \tag{3.23}$$

The solution is obtained by the following equations:

$$\frac{\lambda_1 r_2 + \lambda_2 r_1}{r_1 + r_2} = a \tag{3.24}$$

$$\frac{2(\lambda_1-\lambda_2)^2 r_1 r_2}{(r_1+r_2)^2(\lambda_1 r_2 + \lambda_2 r_1)} = b_\infty - 1 \tag{3.25}$$

$$\frac{1-e^{-(r_1+r_2)t_1}}{(r_1+r_2)t_1} = \frac{b_\infty - b_{t_1}}{b_\infty - 1} \tag{3.26}$$

$$\left.\frac{\partial^3 g(z,t_2)}{\partial z^3}\right|_{z=1} = \mu^3(t_2) + 3at_2(at_2-1)b_{t_2} + at_2(at_2-1)(at_2-2) \tag{3.27}$$

by posing $d = r_1 + r_2$, 3.26 can be solved iteratively by

$$d = \frac{1}{t_1}\left(\frac{b_\infty - 1}{b_\infty - b_{t_1}}\right)(1 - e^{-dt_1})$$

assuming that b_{t_1} is greater that 1´. By posing in the fully developed equation 3.27,

$$K = (\lambda_1 - \lambda_2)(r_1 - r_2)$$

K can be determined from equation, obtained by the combination of 3.27 and 3.13

$$\frac{\partial^3 g(z,t_2)}{\partial z^3}\Big|_{z=1} = a^3{t_2}^3 + 3a^2(b_\infty - 1){t_2}^2 + \frac{3a(b_\infty - 1)}{d}[\frac{K}{d} - a]t_2 + \frac{3a}{d^2}(b_\infty - 1)[K + ad]t_2 e^{-dt_2} - \frac{6a}{d^3}(b_\infty - 1)K(1 - e^{-dt_2}) \tag{3.28}$$

If $K \neq 0$,

$$e = \frac{(b_\infty - 1)ad^3}{2K^2}$$

gives

$$r_1 = \frac{d}{2}(1 + \frac{1}{\sqrt{4e+1}}) \tag{3.29}$$

$$r_2 = d - r_1 \tag{3.30}$$

$$\lambda_2 = (\frac{ad}{r_2} - \frac{K}{r_2 - r_1})(\frac{r_2}{r_1 + r_2}) \tag{3.31}$$

$$\lambda_1 = \frac{K}{r_1 - r_2} + \lambda_2 \tag{3.32}$$

This enables to determine the parameters of the MMPP(2) model.

3.3.2. Determination of MMPP(2) parameters based on queueing statistics

The method has been proposed by [KOS97]. Four input parameters are considered: σ the geometric factor of the autocorrelation function, a_1 and s_1 of the asymptotic of the steady state waiting distribution, and ρ the load of the system. The method has 4 input parameters : σ the geometric factor of the autocorrelation function, a_1 and s_1 of the asymptotic of the steady state waiting distribution, and ρ the load of the system. These four parameters can be determined

i) ρ depends of the model parameters $\rho = \frac{h}{m}$ where m is the inter-arrival time between two cells in the superposed process and h the cell service time.

ii) σ is obtained from the autocorrelation assuming it can be written $c[k] = c_0\sigma^k$ (it is the case for MMPP(2) target model)

iii) a_1 and s_1 are the linear parameters asymptotic of the waiting time distribution in the infinite queuing monitor. They are such that

$$W(s) = \frac{a_1 s_1}{s - s_1} + \frac{a_2 s_2}{s - s_2}$$

so that

$$\lim W(x)_{x\to\infty} = a_1 e^{s_1 x}$$

The fit procedure is performed like the method described by [KOS97] with the following steps :

Step1: Guess an initial vector **g**

Step2: Compute λ_1 from the limit equation for $W(s)$

Step3: Compute r_1, r_2, λ_2 from the others equations (rate, autocorrelation and pole of W(s))

Step4: Compute **g** again

Step5: if **g** is converging stop otherwise go to Step2

To obtain convergence, this method was customized. In particular for Step 4, we calculated **g** by using the following recurrence for the G matrix

$$G_0^{n+1} = 1 - \frac{r_2 G_0^n}{r_1 + (\lambda_1 - \lambda_2)G_0^n} - \exp\Big[-(r_1 + r_2 + \lambda_1 G_0^n + \frac{\lambda_2 r_2 G_0^n}{r_1 + (\lambda_1 - \lambda_1)G_0^n})\Big] \tag{3.33}$$

Moreover, for Step 2, inconsistent solutions had to be avoided. For that, it was proved that $\lambda_1 \in [\lambda_1^{(min)}, \lambda_1^{(max)}]$ where

$$\begin{aligned} \lambda_1^{(min)} &= \frac{s_1 \exp(s_1)}{e^{s_1} - 1} \\ \lambda_1^{(max)} &= \frac{\sigma s_1 \exp(s_1)(s_1 \exp(s_1) - \rho)}{\rho\sigma - e^{s_1}(\rho + \sigma s_1) + e^{2s_1} s_1} \end{aligned}$$

The method corresponds to the simultaneous solution of the determination of the input process and the derivation of its queueing solutions.

This method is working well if the load of the observation is high, otherwise, little statistics on the queueing behavior are gathered.

3.3.3. Derivation for video sources

Frame size distribution In the preceding, a Gamma-Pareto distribution has been introduced to model the frame size distribution. In this section, the derivation of the parameter for the Gamma Pareto are explained with more details.

The parameters for the gamma distribution are chosen to fit to the mean and variance of the measured video size.

$$E[\Gamma] = \frac{s}{\omega} \tag{3.34}$$

$$E[\Gamma^2] = \frac{s}{\omega^2} \tag{3.35}$$

Then the shape of the Pareto is chosen in order in fit to the slope of the tail of complementary cumulative distribution function of measured trace. The second parameter is chosen in order that the two distributions have a common slope at x^*. The distribution is then a Gamma-Pareto distribution

$$f_\Gamma(x) = e^{-\lambda x} \frac{\lambda x (\lambda x)^{s-1}}{\Gamma(s)} \text{ if } x < x^* \tag{3.36}$$

$$f_P(x) = \frac{\alpha k^\alpha}{x^{(\alpha+1)}} \text{ if } x > x^* \tag{3.37}$$

The parameters for the gamma distribution are chosen to fit the mean and variance of the measured video frame size distribution. Since, for the Gamma distribution we have: $E_\Gamma[X] = \frac{s}{\lambda}$ $\text{var}_\Gamma = \frac{s}{\lambda^2}$ the parameters for the Gamma part can be obtained. The shape of the Pareto (k) is chosen in order to fit the slope of the tail of complementary cumulative distribution function of measured trace. The second parameter is chosen in order that the Gamma and the Pareto distributions have a common slope at the junction point x.

Measurements The frame size fitting procedure was applied to different traces collected with the study of [FR01]. Table 3.2 summarizes the results.

The degree of self similarity (not relevant for the rest of the study) is indicated by H the Hurst parameter. The two parameters of the Gamma part of the distribution have been indicated. These are part results of a procedure for the determination.

Figure 3.8 shows the cdf of the video frame size. The first part of the distribution is fitted with a Gamma distribution, obtained to have good agreement in mean and variance. In order to include the heavy tail, a Pareto tail is fitted to the distribution. The slope of the Pareto distribution is fitted to the last part of the distribution. This part is then adapted at the point where the Gamma and the Pareto distribution have the same slopes. All these operations are summarized in the figure.

Trace	H	Gamma.s	Gamma.ω	Fit
StarWars (MPEG2)	0.86	19.58	7.08e-4	Good
StarWars (MPEG4 - high quality)	0.838	3.06	0.00223	Ok
Silence of the Lambs (MPEG4 - high quality)	0.894	2.24	7.77e-4	Average
Office Cam (MPEG4 - medium quality)	0.886	4.36	0.0246	Average/Good

Table 3.2.: Parameters of the gamma distribution for different movies

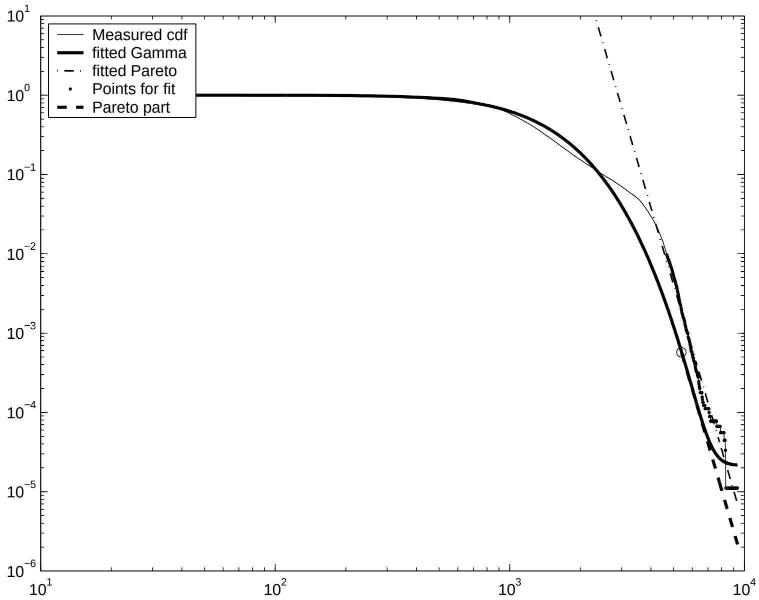


Figure 3.8.: Fit of cdf of video frame size using a Gamma-Pareto distribution

3.3.4. EM methods

These methods are based on the iteration of a two steps procedure. The first step (E) is the expectation calculation and the second step (M) is the maximization of this procedure.

3.3.4.1. Principle

Maximum likelihood The objective of such methods is to obtain the most probable parameters θ of a density function depend of θ ($f(x \,|\, \theta)$). For a data set X of length N, the likelihood is formed (or the log-likelihood) $\mathcal{L}(\theta \,|\, X) = f(X \,|\, \theta) = \prod_{i=1}^{N} f(x_i \,|\, \theta)$. The maximum likelihood estimate $\hat{\theta}$ is the one maximizing $\mathcal{L}(\theta \,|\, X)$.

Example: determination of the parameters of a gaussian distribution
The exponential distribution has the density

$$f(x \,|\, \theta_1, \theta_2) = \frac{1}{\sqrt{2\pi\theta_2}} \exp - \frac{(x-\theta_1)^2}{2\theta_2}$$

with θ_1 mean value, θ_2 variance. in this case $log(\mathcal{L}(\theta_1, \theta_2 \,|\, X)) = -N(\log(\sqrt{2\pi}) + \frac{1}{2}\log(\theta_2) + \sum_{i=1}^{N} - \frac{(x_i-\theta_1)^2}{2\theta_2}$. The maximum likelihood is obtained for $\frac{\delta log(\mathcal{L}(\hat{\theta_1}, \hat{\theta_2} \,|\, X))}{\delta \theta_1} = \frac{\delta log(\mathcal{L}(\hat{\theta_1}, \hat{\theta_2} \,|\, X))}{\delta \theta_2} = 0$, which gives

$$\hat{\theta}_1 = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3.38}$$

$$\hat{\theta}_2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{\theta}_1)^2 \tag{3.39}$$

It corresponds to the standard estimation of mean and variance from a sequence.

EM algorithm The EM procedure is providing a maximum-likelihood estimate when some observations are missing ([Bil97],[DLR77]). If X are the observed data and Y are the missing data, then $Z = \{X, Y\}$ is the complete set of data. The complete data likelihood is then formed $\mathcal{L}^c(\theta \mid z) = \mathcal{L}^c(\theta \mid X, Y) = f(Z \mid \theta) = f(X, Y \mid \theta) = f(Y \mid Y, \theta) f(Y \mid \theta)$. Since only X and θ are known, the complete likelihood can be seen as a random variable of Y, $\mathcal{L}^c(\theta \mid X, Y) = h_{X,\theta}(Y)$. The EM algorithm consist of two steps: the E (expectation) and M (maximization) steps. The E step consists in the calculation of the expected value of the complete-data log-likelihood

$$\mathbb{E}\big[\log f(X, Y \mid \theta) \mid X, \hat{\theta}_{(i-1)}\big] = \int_{y \in Y} \log\big(f(X, y \mid \theta)\big) f(y \mid X, \hat{\theta}_{(i-1)}))dy$$

The M step consists in the maximization of the expectation.

$$\hat{\theta}_{(i)} = \max_{\theta} \mathbb{E}\big[\log f(X, Y \mid \theta) \mid X, \hat{\theta}_{(i-1)}\big]$$

The two steps are repeated as often as necessary. Each iteration is increasing the log-likelihood, so that a (local) maximum is reached.

The determination of $f(Y \mid X, \hat{\theta}_{(i-1)})$ may not always be easy. In particular if the state of unknown values is huge. If not accessible it is replaced by $f(Y, X \mid \hat{\theta}_{(i-1)}) = f(Y \mid X, \hat{\theta}_{(i-1)}) f(X \mid \hat{\theta}_{(i-1)})$

Example:
As an example of the EM, the step ii) of the initialization procedure proposed by [Ryd96] is explained. The hypothesis made is that the sequence of inter-arrival time $\{\mathcal{I}_n\}$ (known parameters) is completed by a state variable s_k. An MMPP(2) is considered, it can be assumed that s_k are i.i.d on 1,2 and $\mathcal{I}_k$ has the density $\lambda_i \exp(-\lambda_i \mathcal{I})$ if $s_k = i$. The model has parameters $\theta = (\alpha_1, \alpha_2, \lambda_1, \lambda_2)$ with $\alpha_1 + \alpha_2 = 1$. The incomplete log likelihood function is

$$\log \mathcal{L}(\alpha_1, \alpha_2, \lambda_1, \lambda_2 \mid \mathcal{I}) = \sum_{k=1}^{n} \log\left(\alpha_1 \lambda_1 \exp(-\lambda_1 \mathcal{I}_k) + \alpha_2 \lambda_2 \exp(-\lambda_2 \mathcal{I}_k)\right)$$

The goal of the EM is to maximize this log likelihood function. The complete likelihood is formed taking advantage of the knowledge of s_k,

$$\log \mathcal{L}^c(\alpha_1, \alpha_2, \lambda_1, \lambda_2 \mid \mathcal{I}, s) = \sum_{k=1}^{n} \log\left(\alpha_{s_k} \lambda_{s_k} \exp\left(-\lambda_{s_k} \mathcal{I}_k\right)\right)$$

At each step of the EM algorithm, the following updates are performed

$$p_1(k) = \frac{\alpha_1^{(i-1)} \lambda_1^{(i-1)} \exp\left(-\lambda_1^{(i-1)} \mathcal{I}_k\right)}{\alpha_1^{(i-1)} \lambda_1^{(i-1)} \exp\left(-\lambda_1^{(i-1)} \mathcal{I}_k\right) + \alpha_2^{(i-1)} \lambda_2^{(i-1)} \exp\left(-\lambda_2^{(i-1)} \mathcal{I}_k\right)} \tag{3.40}$$

$$p_2(k) = \frac{\alpha_2^{(i-1)} \lambda_2^{(i-1} \exp\left(-\lambda_2^{(i-1)} \mathcal{I}_k\right)}{\alpha_1^{(i-1)} \lambda_1^{(i-1} \exp\left(-\lambda_1^{(i-1)} \mathcal{I}_k\right) + \alpha_2^{(i-1)} \lambda_2^{(i-1} \exp\left(-\lambda_2^{(i-1)} \mathcal{I}_k\right)} \tag{3.41}$$

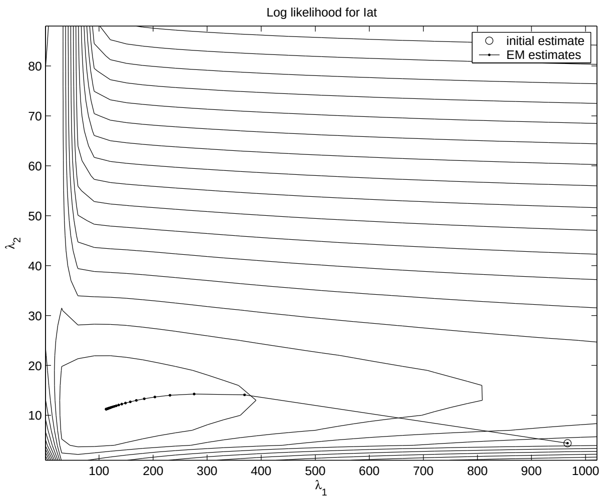


Figure 3.9.: Example EM algorithm (second step of the initialization procedure)

$$\alpha_1^{(i)} = \frac{\sum_{k=1}^{n} p_1(k)}{n} \qquad \alpha_2^{(i)} = \frac{\sum_{k=1}^{n} p_2(k)}{n} \tag{3.42}$$

$$\lambda_1^{(i)} = \frac{\sum_{k=1}^{n} p_1(k)}{\sum_{k=1}^{n} \mathcal{I}_k p_1(k)} \qquad \lambda_2^{(i)} = \frac{\sum_{k=1}^{n} p_2(k)}{\sum_{k=1}^{n} \mathcal{I}_k p_2(k)} \tag{3.43}$$

Figure 3.9 shows the level lines of the incomplete log likelihood ($\log \mathcal{L}(\frac{1}{2}, \frac{1}{2}, \lambda_1, \lambda_2)$ for (λ_1, λ_2). The function has a maximum that can be seen on the graph. The initial value was guessed using the first step of the procedure indicated by Ryden. It can be seen that each steps of the EM algorithm is approaching to this maximum, so that the maximum is reached after some iterations.

3.3.4.2. Principle for MMPP and BMAP

For MMPP or BMAP models, if the parameters of the models have to be estimated, the problem is that at packet arrival no knowledge of the Markov chain state is available and the instants of transition are unknown. These method was used by [Ryd94], [AR99] for MMPPs and can be used also for BMAP after minor modifications. The parameters of the models are $\theta = (\pi_i, Q, \Lambda)$ for the MMPP and $\theta = (\pi_i, D_0, D_1, ..D_R)$ for the BMAP. If a sequence of inter-arrivals time are available $\mathcal{I} = \mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_{n_a}$ The objective is to find θ that maximizes $\mathcal{L}(\theta \,|\, \mathcal{I})$. As suggested in the previous paragraph, some unknowns will be added to the collected data to have a more tractable $\mathcal{L}^c(\theta \,|\, \mathcal{I}, X)$, so will be used to obtain iteratively the requested maximum. The order of the model defining the dimension of the matrices (number of states in the underlying Markov process)is supposed to be known and is noted m.

BMAP packet size mapping The difference between BMAP and MMPP is that BMAP is taking into account the packet size. The BMAP is using a range of packet size from 1 to K to describe packet sizes. Each packet from a measurement has to be classified into one of the K-th category. [KLL02] affirmed that a range up to $K = 3$ shall be sufficient to model IP traffic. The span of the packet size is then splitted in K intervals. The size of each packet is then mapped to the adequate category. The sequence of mapped size for each sample of the inter-arrival time is available

Complete likelihood for MMPP and BMAP Assuming a MMPP with infinitesimal generator Q and arrival rate Λ. Given some observation of n_a arrivals with the inter-arrival time sequence $\{\mathcal{I}_1, \ldots, \mathcal{I}_{n_a}\}$, the unknown information is the instants at which the MMPP has changed its state. Even if not available it is denoted $\{\tau_1, \tau_2, \ldots, \tau_{n_t}\}$ if there were n_t transitions. The state of the process after the state has been changed is $s_k = \mathcal{J}(\tau_k+)$. The time spend in state s_k is noted $\Delta\tau_{k+1} = \tau_{k+1} - \tau_k$. The process $\{\mathcal{J}(t)\}$ is called the underlying continuous time Markov chain. The number of arrivals in the intervals $[\tau_{k-1}, \tau_k]$ is written z_k.

If Y_k is the time between the $(k-1)$st event and kth event $\{\mathcal{J}_{k-1}, Y_k\}$ is a Markov renewal process with transition matrix

$$f(y) = \exp\left\{(Q - \Lambda)y\right\}\Lambda \tag{3.44}$$

and the transition probability matrix of the Markov chain $\{J_k\}$ is

$$P = \int_0^\infty f(y)dy = (\Lambda - Q)^{-1}\Lambda \tag{3.45}$$

The maximum likelihood of the MMPP with absolute knowledge is

$$\mathcal{L}(\Theta \,|\, \mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_{n_a}, \tau_1, \tau_2, \ldots, \tau_{n_t}) = \tag{3.46}$$

$$\pi_{X(T_1)} \quad \times \Big\{ \prod_{k=1}^{n_t} q_{X(Tk)} e^{q_X \Delta T} \times \frac{q_{X,X}}{q_X} \Big\} \times e^{-q_{s_{n_a+1}} \Delta\tau_{n_a+1}} \times \Big\{ \prod_{k=1}^{n_t+1} \frac{(\lambda_{s_k}\Delta\tau_k)^{z_k}}{z_k!} e^{-\lambda_{s_k}\Delta\tau_k} \times \frac{z_k!}{(\Delta\tau_k)^{z_k}} \Big\}$$

The first term is the initial state probability, the second one the conditional density of the realization of X and the third the conditional density of the observed arrivals

The derived-algorithm is them as follows

1. set $L(0) = \pi^0$ and for $k = 1, \ldots, n$ $L(k) = L(k-1) \times f(\Delta T_k)$
2. set $R(n+1) = \mathbf{1}$ and for $k = n, \ldots, 1$ set $R(k) = f(\Delta T_k)R(k+1)$
3. set $A_{ij} = 0$ and $B_i = 0$ for $i, j = 1, \ldots, r$.
4. for $k = 1, \ldots, n$ set:

$$A_{ij} \leftarrow A_{ij} + L(k-1) \int_{t_{k-1}}^{t_k} \overline{F}(t - t_k) \times \mathbf{1}_i \times \mathbf{1}_j^t \times f(t_k - t) dt R(k+1) \tag{3.47}$$

5. for $k = 1, \ldots, n$ set:

$$B_i \leftarrow B_i + L(k-1) \times \mathbf{1}_i \times \mathbf{1}_i^t \times R(k+1) = B_i + L_i(k-1) \times R_i(k+1) \tag{3.48}$$

6. compute likelihood $\mathcal{L} = L(n) \times \mathbf{1}$ and update estimates $\hat{q_{ij}} = \hat{\lambda}_i =$

$$\hat{q_{ij}} = q_{ij}^0 \frac{A_{ij}}{A_{ii}}, \; i, j = 1, \ldots, n \quad \hat{\lambda}_i = \frac{B_i}{A_{ii}}, \; i = 1, \ldots, r \tag{3.49}$$

For the BMAP, the sequence of batch sizes is also observed and is denoted $\{b_1, \ldots, b_{n_a}\}$. If the BMAP is handling M packet size categories, the following matrices are required $D_0, D_1, ..., D_M$. The transition density matrix of each arrival if then $f_k(t) = \exp(D_0 t) \times D_k \; 1 \le k \le M$.

3.3.4.3. Initialization procedures

In order to have EM estimates converging quickly a good initialization procedure is required

Ryden - for MMPP(r) Ryden suggests the following procedure for the initialization

(i) Initial Rate Estimation - Only the central 90% of the sorted samples are considered, form the sequence of inter-arrival times in the ascending order $\{y_{(k)}\}$ and compute $d = (\log y_{([0.95n])} - \log y_{([0.05n])})/(r-1)$ the first estimates are $\lambda_i^{(1)} = 1/\exp\{\log y_{([0.05n])} + (i-1)d\}$ for $i = 1, ..., r$ and $\alpha_i^{(1)} = 1/r$ for $i = 1, ..., r$

(ii) EM exponential

The EM algorithm for independent regime of exponential densities with initial values $\alpha^{(1)}$ and $\lambda^{(1)}$ shall be used to find estimates of the model $\alpha^{(2)}$ and $\lambda^{(2)}$. The obtained results are estimates of π and λ.

(iii) MAP estimation - After the determination of the parameters in step ii) the states are reevaluated using the maximum a posteriori probability. $\hat{x}(t_k) = \arg\max_i \hat{\alpha_i}^{(2)}\hat{\lambda}_i^{(2)} \exp\{-\hat{\lambda}_i^{(2)} y_k\}$. The elements of the matrix $\hat{P}^{(3)}$ are evaluated by $\hat{p}_{ij}^{(3)} = n_{ij}/n_i$ where $\hat{n}_{ij} = \#\{k : 1 \leq k < n, \hat{x}(t_k) = i, \hat{x}(t_{k+1}) = j\}$ and $\hat{n}_i = \#\{k : 1 \leq k < n, \hat{x}(t_k) = i\}$

(iv) EM HMM estimation Initial estimates $\hat{P}^{(3)}$ and $\hat{\lambda}^{(2)}$... $\hat{P}^{(4)}$ and $\hat{\lambda}^{(4)}$ are estimates of P and λ respectively.

(v) Final Forming - Let $\hat{Q}^{(5)} = \text{diag}(\hat{\lambda}^{(4)})[I - (\hat{P}^{(4)})^{-1}]$. $\hat{Q}^{(5)}$ is an estimate of Q.

Alternative Initialization procedure for MMPP(2) Because of the results of the previous method were odd for some cases (the matrix Q had negative non diagonal elements and positive diagonal elements), alternatives method for the initialization procedure were investigated. For example, a dedicated method is presented in [MH87]. Initialization could also be completed by an combination of EM algorithm assuming an exponential independent model, followed by a Hidden Markov Chain estimation to get into further precision, that making initial guesses of the initial model parameters. For MMPP(2) others procedures can be applied using the characteristics of the inputs stream.

As each initialization procedure gives different initial estimates, the EM algorithm is then converging to a different solution, since the MMPP model parameters are not unique. Parameters obtained can then not be compared directly but the mean arrival rate or other quantitative parameters have to be looked at.

3.4. Traffic models for aggregated flows

3.4.1. Theoretical aggregation

We suppose that we have 2 MMPP process with rate and generator matrixes

$$\Lambda_1 = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \quad Q_1 = \begin{pmatrix} -r_1 & r_1 \\ r_2 & -r_2 \end{pmatrix} \tag{3.50}$$

$$\Lambda_2 = \begin{pmatrix} \lambda'_1 & 0 \\ 0 & \lambda'_2 \end{pmatrix} \quad Q_2 = \begin{pmatrix} -r'_1 & r'_1 \\ r'_2 & -r'_2 \end{pmatrix} \tag{3.51}$$

If the first model had the states $\{1, 2\}$ and the second the states $\{1', 2'\}$, then the superposition has the states $\{1 + 1', 1 + 2', 2 + 1', 2 + 2'\}$.

Then the superposition process is described by a MMPP with rate and generator matrices $Q = Q_1 \boxplus Q_2$ and $\Lambda = \Lambda_1 \boxplus \Lambda_2$ where

$$\Lambda = \begin{pmatrix} \lambda_1 + \lambda'_1 & 0 & 0 & 0 \\ 0 & \lambda_1 + \lambda'_2 & 0 & 0 \\ 0 & 0 & \lambda_2 + \lambda'_1 & 0 \\ 0 & 0 & 0 & \lambda_2 + \lambda'_2 \end{pmatrix} \tag{3.52}$$

$$Q = \begin{pmatrix} -(r_1 + r'_1) & r'_1 & r_1 & 0 \\ r'_2 & -(r_1 + r'_2) & 0 & r_1 \\ r_2 & 0 & -(r_2 + r'_1) & r'_1 \\ 0 & r_2 & r'_2 & -(r_2 + r'_2) \end{pmatrix} \tag{3.53}$$

with $A \boxplus B = A \otimes I_B + I_A \otimes B$ and $\otimes$ is the Kronecker product [1]. The Kronecker product of two matrix

[1]The operation of Kronecker sum written with $\boxplus$ is often noted $\oplus$ but was not used here because it is the notation used for the aggregation operator

C ($m \times n$) and D ($r \times s$) is the ($mr \times ns$) matrix

$$C \otimes D = \begin{pmatrix} c_{11}D & \dots & c_{1n}D \\ \vdots & & \vdots \\ c_{n1}D & & c_{nm}D \end{pmatrix} \tag{3.54}$$

For the superposition of D-BMAP, [Spa02] considered the superposition of M D-BMAP [2] with matrix $D_k^{(i)}$ $k \geq 0$, $1 \leq i \leq M$ and showed it is a D-BMAP with matrix $D_k, k \geq 0$

$$\begin{aligned} D &= \bigotimes_{i=1}^{M} D^{(i)} \\ D_0 &= \bigotimes_{i=1}^{M} D_0^{(i)} \\ D_1 &= D_1^{(1)} \otimes (\bigotimes_{i=2}^{M} D_0^{(i)}) + \ldots + (\bigotimes_{i=1}^{M-1} D_0^{(i)}) \otimes D_1^{(M)} \\ &\vdots \end{aligned} \tag{3.55}$$

For the modelling of web traffic, a 3×3 with $M = 3$ BMAP models is used, the size of the matrix D for the superposition is then $3^M \times 3^M$ and the number of matrix D_k is growing up to M^M...This making the superposition even more untractable, without further simplifications.

Superposition of ON/OFF processes Here, the states can be regrouped, so that the superposition of M processes has only $M + 1$ states: in state i, where $0 \leq i \leq M$ there are i sources in the ON state. In this case, the state explosion can be prevented.

If it is assumed that each source has a input rate in the on state γ_{ON} with an ON probability of p_{ON}, for the superposition of M sources, it comes

$$\Pr[i \times \gamma_{ON}] = C_M^i (p_{ON})^i (1 - p_{ON})^{M-i}$$

Hence, the M+1 states correspond to these arrival rates. The transition towards the other states can be derived by considering the probability that one source finishes to send or that an additional sources get active.

Usage report In the numerical study, the aggregation of two MMPP(2) models has been derived. The initial MMPP(2) model is the model that was derived based on the recording from the aggregation of 100 VoIP sources. For two of these models, using the EM method, a MMPP(2) equivalent was derived. But no conclusions were drawn from these results conducting to a method for simplifications.

3.4.2. Methodological directions

When the theoretic aggregation of MMPP models is considered, the number of states explodes quite quickly. For example for the superposition of n MMPP(2) the resulting matrix Q are $2^n \times 2^n$. The storage of the matrix can quickly become a problem.

[2] for the considered result, D-BMAP are equivalent to BMAP because D-BMAP are just discrete time equivalent of BMAP

3.4.2.1. Asymptotics

[Neu81] introduces asymptotics for PH-distribution $(T, \boldsymbol{\alpha})$ in theorem 2.3.1 if the complementary CDF of the renewal distribution has the following tendency:

$$1 - F(t) = Ke^{-\eta t} + o(e^{-\eta t}), \quad t \to \infty \tag{3.56}$$

where $-\eta$ is the eigenvalue of T with largest real part and $K = \alpha\mathbf{v}$. $\mathbf{v}$ is determined from the couple $(\mathbf{u}, \mathbf{v})$ of the left and right eigenvectors of T for the eigenvalue $-\eta$ verifying $\mathbf{uv} = \mathbf{ue} = 1$. In other words, every PH-distribution is asymptotically exponential distributed. For an MMPP with (Q, Λ) since $f(s) = (sI - Q + \Lambda)^{-1}\Lambda$, the pole are the eigenvalues of $Q - \Lambda$, so $-\eta$ is the eigenvalue of $Q - \Lambda$ with the maximum real part.

Also for a MMPP(2) $F(t) = 1 - q(e^{(-u_1 t)}) - (1 - q)(e^{(-u_2 t)})$

3.4.2.2. Matrix simplification

In [Ris02] two aggregation methods are discussed: first the exact decomposition of the matrix Q and secondly the stochastic complementation. The first one consist in not consider "small" terms of the matrix to get a decomposable matrix Q. The second method is based on the combination of the states into larger states: in stead of estimating the probability $q_{i,j}$ to jump from a state to another, the aggregated probability is used $q_i = \sum_j q_{i,j}$. This enables to obtain much compacter results. But it is worth, for models of quite larger states, in our case, the number of states is kept small in order to reduce the state explosion.

Circulant matching procedure The purpose of such method is to find a circulant MMPP which power spectrum $P_c(w)$ and cumulative distribution $F_c(w)$ matching with the spectrum and cumulative distribution of the superposition.

it was used for example [Spa02] for discrete D-BMAP. A continuous version was developed within development of the SMAQ tool (Statistical Match And Queueing tool) [LPA94] using results from [SqL93] for discrete models and [qLH93] for continuous models.

In the case of circulant MMPP, the matrix Q has the first line of the form

$$[-a_o\, a_1\, \ldots\, a_{N-1}]$$

with $a_0 = \sum_{k=1}^{N-1} a_k$

Two steps are necessary for the construction

1. construction of $\mathbf{a}$ in order to have P_c matching P
2. construction of γ in order to have F_c matching F

The described method could not be completely reproduced. A lot of procedure were not reaching convergence with the criterion required, or using the algorithms suggested in the literature. For example, the use of a constraint optimization algorithm was impossible because the constraints were too stringent. This is due to a misinterpretation of the text of the corresponding articles. The corresponding authors were contacted but were not able to furnish more information, because the work have been stopped for many years.

3.5. Summary of models used

The table 3.3 summarizes the different models used in the aggregate traffic pyramid.

At application level, the On/Off model is used for VoIP, a session-based model is used for web and a Gamma-Pareto distribution with a a self similar sequence for the video model.

At subscriber level, the interest of developing dedicated aggregate models was too low that no models were suggested. The node models could be used if necessary.

Level	VoIP	Video	web
Applications	On/Off	FGN + Γ, Pareto	sessioned-BMAP
Subscriber			
Node	MMPP(2)		BMAP
Transport	Gaussian	Gaussian	Gaussian

Table 3.3.: Summary of the traffic models used in the aggregate traffic pyramid

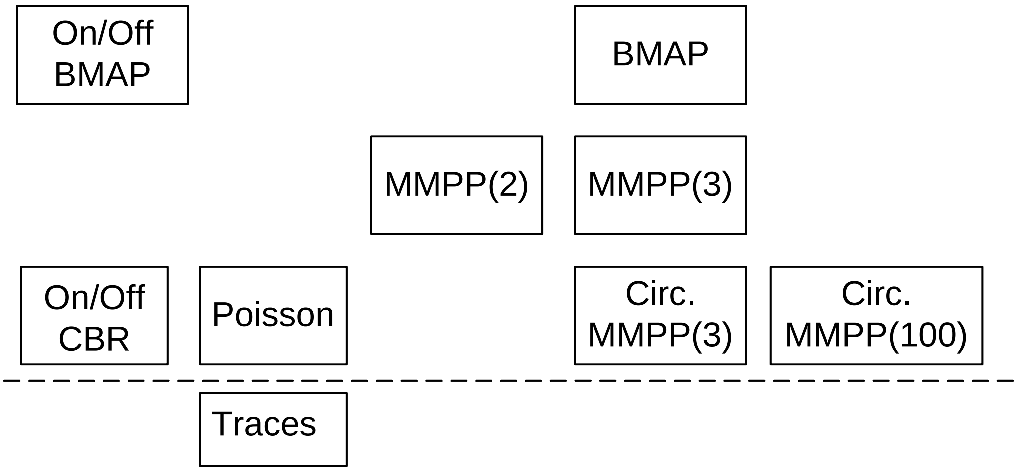


Figure 3.10.: Summary of models used in this work

At node level, the MMPP(2) is used for modelling the voice service and the BMAP is used for web and heterogeneous traffic mixes.

At the transport level, a gaussian model is used for modelling the physical frame occupation.

Figure 3.10 summarizes the models used. The most realistic model is the one down in the picture and consists of the collection of traffic traces.

Figure 3.11 shows the different models that will be investigated, the corresponding models and where equivalency are considered

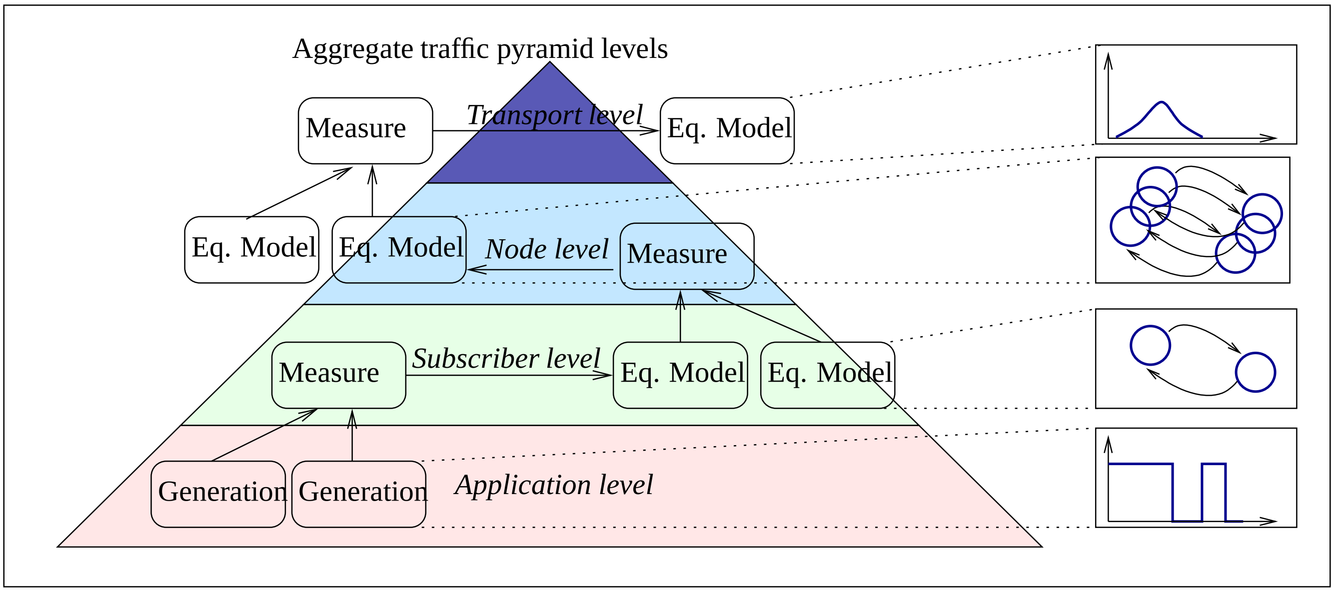


Figure 3.11.: Architecture of models in the aggregate traffic pyramid

Je prendrai mon bâton et sur la grand route
J'irai, et je dirai aux ânes, mes amis :
Je suis Francis Jammes et je vais au Paradis.

Francis Jammes, Le deuil des primevères (1901)

4 Application of the pyramid approach to a practical case

In this chapter, a practical case is discussed to illustrate the applicability of the pyramidal approach. The objectives of this chapter are to show how the system can be modelled with this approach and to present the simulator that will be used in the following chapters to show numerical results demonstrating the interest of the aggregate traffic model pyramid.

4.1. Description of practical scenario

4.1.1. Presentation of case under study

The system under consideration is a satellite system offering to aircraft passengers web access and telephony via dedicated aeronautical terminals. The system under consideration is then composed of the elements described on figure 4.1. Their single characteristics will be discussed in the following. Systems have also been designed for satellite-based provision of Internet access and voice telephony inside airplane for example within the WirelessCabin project [BCC+04b],[BCC+04a].

The core element is a satellite system in order to provide communication services for aircraft. A Geostationary Earth Orbit (GEO) satellite has been chosen because it is providing a wide coverage where a large number of potential aircraft are served.

Figure 4.1 illustrates the considered case using a pyramidal shape. The application level is composed of two applications that are voice and web services (for voice service VoIP will be preferred in regard to GSM for modelling purposes). The subscriber level is composed of all passengers of the aircraft covered by the system. The node level is composed of all these aircraft. The transport level is satellite system providing the connectivity with the ground. For the diversity of flying aircraft (from 4 seat-Cessna to the latest Airbus A380 with up to 800 seats) and the diversity of flight routes (ranging from a local flight to intercontinental flights) no unique system is providing a high data rate connectivity with the ground, even if, because of the cruise flight altitude up to 10 km for commercial jet, the access to the terrestrial infrastructure is difficult (in particular over oceans) and legally restricted (like GSM usage in commercial aircraft). Such satellite systems are still under development or in early deployment. It is then interesting to investigate (or forecast) the future traffic load of such system. To use satellite services, each aircraft will need a dedicated antenna and a satellite terminal on-board. The main part of the traffic will be generated from the passengers instead of the crew (but the crew could require more secured connections), so that the passengers are considered as the subscribers of the pyramid model introduced

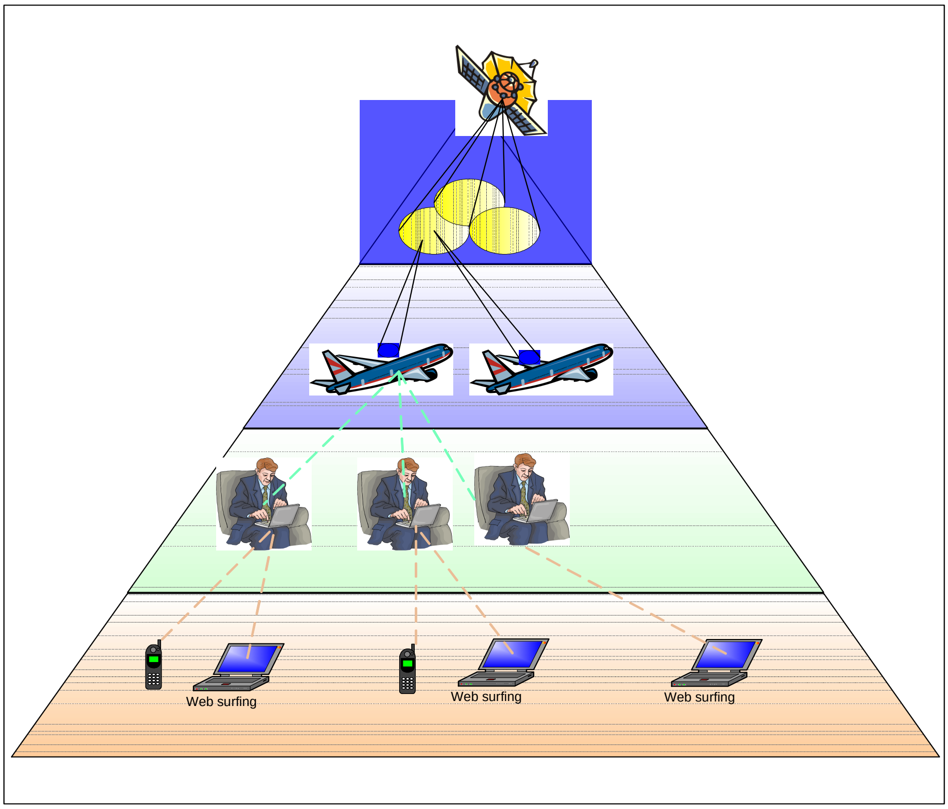


Figure 4.1.: Overview of the considered scenario

Levels	Application level	Subscriber level	Node level	Transport level
Entities	Web surfing	Passengers	Aircraft	Satellite system
	VoIP Client		Satellite Terminal	Up and Down-link

Table 4.1.: Location of entities generating traffic in the aeronautical scenario

earlier. Table 4.1 is showing the mapping of the different entities into the pyramid levels.

4.1.2. Summary of system hypothesis

4.1.2.1. Protocols and packet sizes

In the considered system, the satellite could be kept generic and it could be assumed that protocols like Digital Video Broadcasting - Satellite version 2 (DVB-S2) and Digital Video Broadcasting - Return Channel via Satellite (DVB-RCS) (defined in the following standards [ETS05a],[ETS05b]) are implemented. The capacity of the system is fixed for the aeronautical users considered here.

At subscriber level, it is reasonable to assume that IP packets are manipulated (for voice application using GSM the effective translation is performed within the base station installed inside the aircraft, but as said earlier VoIP is preferred for this study). So the subscriber packets are obtained from the applications data content after addition of an appropriate header. At node level, a scheme implying packet fragmentation was chosen because the satellite access scheme requires the transmission of fixed size packets, so packets will eventually be fragmented to conform to the corresponding size (that will in practice be the size of an ATM or MPEG frame) used in the transport level.

The transport level packets were modelled assuming it is offering N_1 free cells per frame and per node. The frame duration is assumed to be $T = 26.5ms$, ($\mathcal{P}_4$ packets). Figure 4.2 shows the transport level format. The packet has the capacity of $BW \times N_n$ $\mathcal{P}_3$ packets every frame. The $\mathcal{P}_3$ packets are built into the $\mathcal{P}_4$ in two steps. The first step consists in assigning the packets of the first QoS up to the maximum

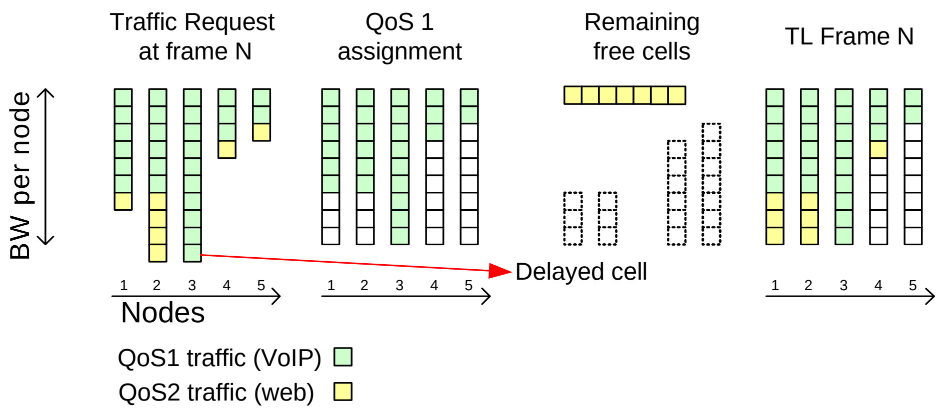


Figure 4.2.: Format of the packet at the transport level

node capacity. The second step consists in assigning capacity to the shared remaining cells of the second QoS type. In case of insufficient resource, the second QoS cells are served according to queue length. In the example if only 3 cells would be available for the second category, the 3 cells from the 2^{nd} queue would be served. The drawback of this policy is also shown on figure 4.2, where a first QoS type is being delayed. It is assumed that these priority scheme are managed transparently because transmission are handled over IP packets (in real system such situations could be avoided by the procedure of Call Admission Control / Connection Admission Control (CAC) that should not accept traffic sources that would generate a data volume above the node capacity).

The node level packets ($\mathcal{P}_3$ packets) are modelled assuming that the cells are of fixed size with a net capacity of 424 bits and an header of 56 bits . This corresponds to the classical ATM cell content of 53 bytes and a satellite-specific header like the one used in the EuroSkyWay system ([CL99a]). It gives a total size $\mathcal{S}_3 = 480$ bits.

The subscriber level packets ($\mathcal{P}_2$ packets) were modelled assuming a header of 48 bits. Indeed as defined in their specifications, TCP and IP have each an header of 20 + 20 bytes [Pos81a], [Pos81b]. Data are included with an header of 48 bytes. A compression scheme of 1:8 was assumed. This can be achieved using mechanisms like the ones described by [Jac90] (with a fixed size of 32 bits). It gives $\mathcal{S}_2 = \mathcal{S}_1 + 48$ bits. Because of the size of the $\mathcal{P}_1$ is below 12000 bits (1500 bytes), only the header addition has to be considered.

The application level packets ($\mathcal{P}_1$ packets) are modelled by their real size in bits. For Voice over IP (VoIP), the value of $\mathcal{S}_1 = 800$ bits was often used. For web traffic variable size were used, for instance the BMAP standard model uses a set of 752, 4600 and 11712 bits packets like in [KLL02].

These packet formats correspond to the original packets size for voice packet and TCP packets for the web model at the application level, at the subscriber level the addition of an header is similar to IP packet header addition. The fixed format of the node level packets corresponds to the fixed size ATM or MPEG frame. With the described format of the transport level packet, a model close to the real behavior with a bounded capacity and a carrier allocation per terminal is obtained. The values chosen for the packets sizes and header and their respective characteristics are in accordance with the values that would occur in a real system.

4.1.3. QoS strategy

The DVB-RCS foresees some schemes for resources assignment in the system

- Continuous Rate Assignment (CRA)
- Rate Based Dynamic Capacity (RBDC)
- Volume Based Dynamic Capacity (VBDC) and Absolute Volume Based Dynamic Capacity (AVBDC)
- Free Capacity Assignment (FCA)

In the simulator, since two types of traffic are considered, two type of QoS are considered (cf. to 4.4 to further details on the simulator): the telephony services were mapped to the first class of Quality of service, web services were mapped to the second class. By doing this, a minimum delay is expected for voice service because these applications are sensitive to delay variations. The web applications shall be able to use the over-assigned bandwidth for the transmission of typical best effort traffic. Each of these categories describes the type of capacity assignment that will be obtained in the system. In the simulator, the first category of QoS is corresponding to the volume based dynamic scheme (VBDC) and the second category of of QoS is corresponding to a free-capacity assignment (FCA) taking into consideration the limited amount of capacity of the satellite (or sub-portion if only a portion of the system is dedicated to aeronautical applications).

At the reception of the packets, QoS parameters for the end to end transmission can be evaluated. For example, it can be checked if the range of the delay variation is acceptable or if the delay jitter corresponds to the QoS requirements that were requested for transmission are meet.

4.1.4. Resource allocation mechanisms

For the allocation of the satellite bandwidth to the different aircraft, it was assumed that the satellite is regenerative i.e. the use of the available resource is managed autonomously by the satellite for the downlink and the uplink links. Hence estimations of resource requirements have to be exchanged by the different actors via signalling. The different entities implied in this process are depicted of figure 4.3. They include a Network Control Center (NCC) in charge of the management of new connections. A Traffic Resource Manager (TRM) located on board of the satellite is responsible of the management resource management (by enforcing the decision of resource assignment on the physical link that were taken either directly on board (short-term resource management) or by the NCC (medium and long-term resource management) and transmitted by signalling). Inside the NCC, the CAC is the entity taking decision on acceptance or refusal of new connections establishment. For IP connections, the connection life time need to be known and corresponds to the session duration that can be keep ”permanent” by the transmission of periodic keep-alive packets. Before to be able to exchange traffic, the node needs to exchange information about the connection parameters with the NCC. Inside the NCC, it is necessary to store the requirements that were advertised by the users. Afterwards, this information is transmitted to the resource handler in charge of resource management. This entity (indicated with RM) will transmit the assignment to the TRM on board of the satellite.

In the simulator, equivalent model will be used, so it is not possible to get estimates of the approximated user. At the best, ”approximated” estimates could be produced but they would not allow precise resource management, where each source has to declare an estimation of its traffic volume. Somehow, the behavior of the resource allocation mechanisms will be modified. Since the work was concerned more with the elaboration of approximate models that with the elaboration of approximate estimate of traffic volume, the inclusion of a policer after each source was not considered.

Moreover, the precise consideration of resource management includes the consideration of delay due to the request / assignment procedure. Since a ”stationary” system was considered (with established connections), no precise resource scheme was considered, with an accurate modelling of establishment time and potential delay occurring when the system is loaded.

Anyway, even with these simplifications on the resource scheme, the following effects can be included.

1. Limited bandwidth of the satellite link;
2. Prioritized resource assignment;

At every node also, a mechanism for the management of the different connections of all passengers though a common terminal as indicated in figure 4.4 is required. For example, inside an aircraft, the number of outgoing calls could be limited by a preliminary CAC, so that new outgoing calls could be prevented and will remain refused temporarily as long as the other calls have not been finished.

In the simulator, no particular focus to this topic is given, since this multiplex of subscribers is not typical of a satellite network. Moreover, if the node traffic is policed by such scheme, it is easy to correct

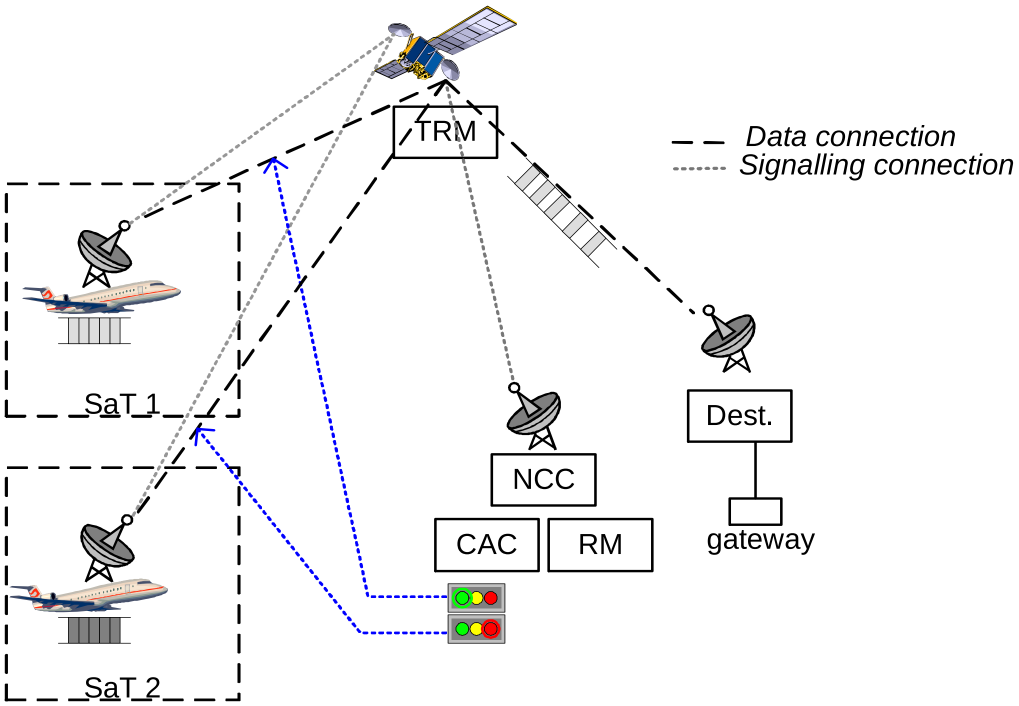


Figure 4.3.: Entities implied by resource management in the satellite system

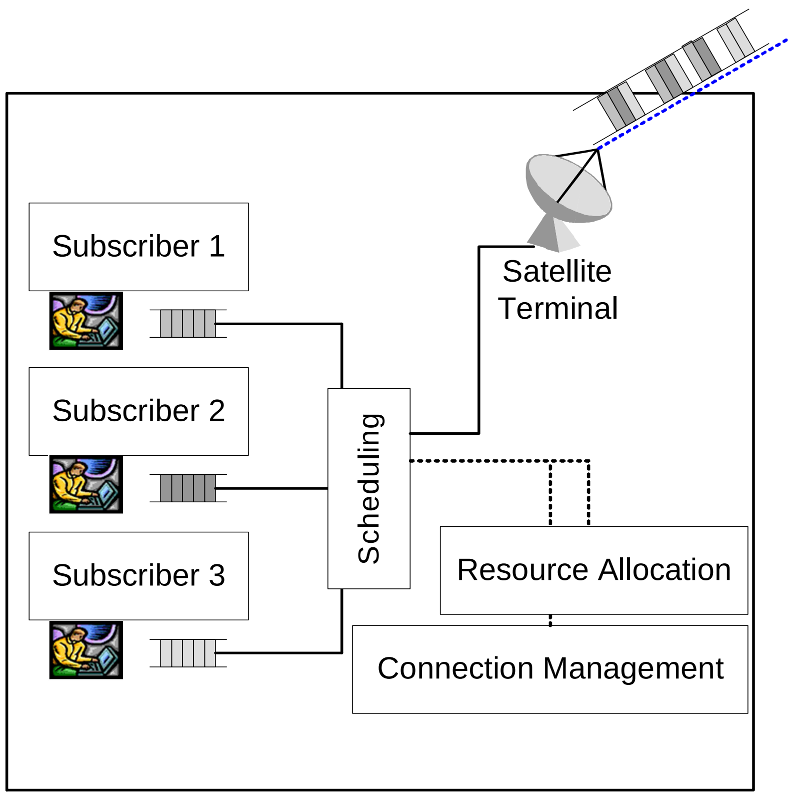


Figure 4.4.: Entities implied in the node output elaboration

the results by introducing a corrective factor between the number of active users and the number of accepted users.

4.1.5. Numerical parameters

In this paragraph, the numerical parameters necessary for the system description are summarized and together with the packet sizes introduced in 4.1.2.1 is describing numerically the case studied.

The operating capacity of the transport level has been fixed to $BW = 250\ \mathcal{P}_4$ cells per frame and per node. This corresponds to a net capacity at the node level of 120000 bits per node and per frame. So a net capacity rate of $4.52\,10^6$ bits per s for a single node. With such values the system can accommodate telephony services and web services for a reasonable number of users (with the standard voice model, up to 130 voice users can be accepted in the system) inside each served aircraft.

The standard system is defined with the following elements. Deviations of these values are possible but will be mentioned appropriately. The number of aircraft accommodated by the system is set to 50, hence corresponding to an overall capacity of 226 Mbits per s. (28.3 Mbytes/s). This corresponds to the full capacity of the system (or the portion of the system allocated to aeronautical services). The number of passengers per aircraft is set to 100. The duration of a frame at transport level is set to $26.5ms$. With all these schemes and numerical settings, a good description of the case under study is available.

4.2. Characterization of the operators

The different operators that were defined in chapter 2 are now described in this application example. Their effect on the flows generated by the traffic models will be investigated in the next chapter.

4.2.1. Subscriber level

4.2.1.1. Operator Addition $\oplus$

At the subscriber level, the addition operator $\oplus$ is responsible of the aggregation of the flows coming from the different applications the subscriber is connected to.

The operator will consist in addition of two streams from the application level. If the packets have different arrivals time, the addition is straightforward: packet output occurs in respect to arrival time without modification of the packet lengths. If the arrival time is the same, the two packets are output at this time with their original size.

For the addition of two streams $\Phi_1^{(1)}$ and $\Phi_2^{(1)}$ from two applications, the resulting stream Φ is the sequence of packet size ordered in respect to the absolute time. The addition can be computed more practically in the absolute time reference. For two sequences $(\mathcal{S}_i^{(1)}, \mathcal{T}_i^{(1)})$ and $(\mathcal{S}_j^{(2)}, \mathcal{T}_j^{(2)})$, a new sequence $(\mathcal{S}_k, \mathcal{T}_k)$ is built with $\mathcal{T}_k \in \{\mathcal{T}_i^{(1)}, i \in N\} \bigcup \{\mathcal{T}_j^{(2)}, j \in N\}$ and $\mathcal{T}_k \leq \mathcal{T}_{k+1}\ \forall k$. $\mathcal{S}_k = \mathcal{S}_i^{(1)}$ if $\mathcal{T}_k \in \{\mathcal{T}_i^{(1)}\}$, otherwise $\mathcal{S}_k = \mathcal{S}_j^{(2)}$.

4.2.1.2. Operator Level conversion Δ

At this level, the operator of level conversion $\Delta_{1\to 2}$ consists only in the addition of a fixed header (of 48 bits) as suggested in 4.1.2.1. Internally, the information contained in the $\mathcal{P}_1$ packet is copied into the $\mathcal{P}_2$ packet.

For a sequence $(\mathcal{S}_i, \mathcal{I}_i)$, the $\Delta_{1\to 2}$ transformed sequence $(\mathcal{S}'_i, \mathcal{I}'_i)$ is built with $\mathcal{S}'_i = \mathcal{S}_i + 48$ and $\mathcal{I}'_i = \mathcal{I}_i$.

4.2.1.3. Operator Level conformation $\sqsupset$

At this level, the operator of level conformation $\sqsupset_2$ consists in a minimal number of actions. Each incoming packet (after the level conversion) has already the format of the next level, so that no fur-

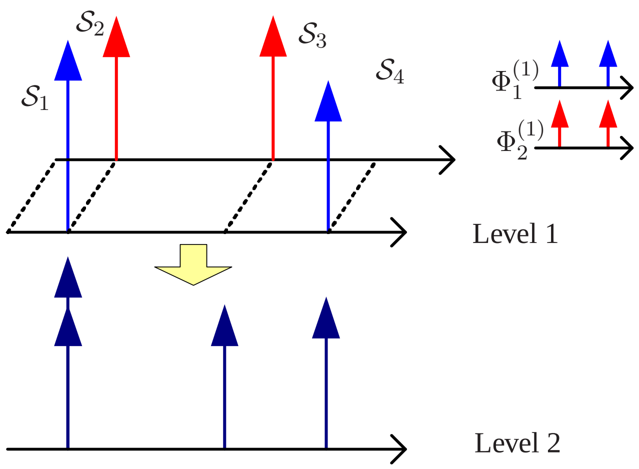


Figure 4.5.: Aggregation of two level 1 traffic streams into a level 2 traffic stream

ther modifications are required. An identifier is introduced in the $\mathcal{P}_2$ header to indicate the originating application.

Finally, it can be written, without using the level conformation operator.

$$\Phi^{(2)} = (\Phi_1^{(1)} \oplus \Phi_2^{(1)})^{\Delta_{1\to 2}}$$

4.2.2. Node level

4.2.2.1. Operator Addition $\oplus$

The operator $\oplus$ is performing the addition of the traffic streams from the subscriber level. The operator is identical to the one described previously.

4.2.2.2. Operator Level conversion Δ

At the node level, fixed size packets are requested. The selected format for $\mathcal{P}_3$ packet is 480 bits including a header of 56 bits. A $\mathcal{P}_2$ packet of size $\mathcal{S}_2$ is then fragmented in $n_P = \frac{\mathcal{S}_2}{424}$ packets. The operator of level conversion $\Delta_{2\to 3}$ has to perform this size conversion.

Some information from the $\mathcal{P}_2$ need to be recorded in the $\mathcal{P}_3$ packet in order to be able to reconstruct them after reception. Six different types of $\mathcal{P}_3$ packets have been defined: SCM and SCMd for single cell message with and without dummys, BOM for the first cell within a sequence, COM for the next one and EOM and EOMd for the final cell. If their is a mismatch between the original information and the net cell capacity, some dummy content is added in the last (or single) packet.

For an abstract unit $(\mathcal{S}, \mathcal{I})$, the $\Delta_{2\to 3}$ transformed abstract units sequence $(\mathcal{S}'_j, \mathcal{I}'_j)$ is built with $\mathcal{S}'_j = 480$ for $j = 1, ..., n_P$ and $\mathcal{I}'_j = \mathcal{I}$ where $n_P = \lfloor \frac{\mathcal{S}_2}{424} \rfloor$. These numerical values and the fragmentation mechanism were chosen because they were in use in the EuroSkyWay system [CL99a].

4.2.2.3. Operator Level conformation $\sqsupset$

The node level is preparing the insertion of the traffic into the satellite transport level. For this reason, the node level contains some buffers that enable to store the traffic during the connection establishment. Since the concern was on established connections, the modelling of these buffers (implying a delay between the first transmitted packet and the initiation of the connection) is not capital: the limiting element is the satellite level treated in the next paragraph, where capacity limits show their influence.

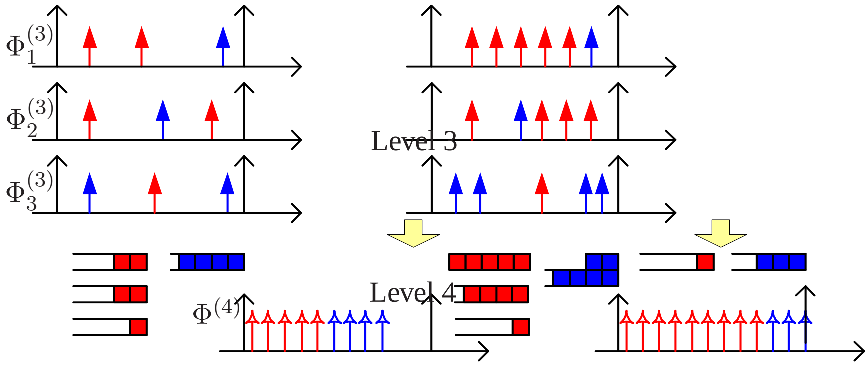


Figure 4.6.: Modification of the level 3 traffic stream before transmission over level 4

One important parameter is to decide the number of outputs of the level conformation operator. As explained in chapter 2, one output is preferable for compatibility with the other operators, but one output per QoS class would facilitate the expression of the next level addition operator.

4.2.3. Transport level

The big difference when moving to the transport level is that the transport level is working with a given clock time ($T_s = 26.5$ms) used to build the frame. This value is the duration of the frame duration in the EuroSkyWay system [CL99a] and in the DVB-RCS specification [ETS05b]. The transport level is the level where the most modifications of the traffic streams take place. These modifications are mainly due to adaption required by the structure of the link (with a bounded capacity) and the consideration of different QoS.

Figure 4.6 shows the modification of the level 3 traffic stream before transmission over level 4. The buffers are considered initially empty. They are 5 first category packet arrivals and 4 second category in the first case depicted. The first category arrivals are all below the capacity allocated per node (4 packets per node in this example), so these packets can all be served. The remaining capacity is used for the transfer of second category packets. On the second case there are 10 arrivals of first category packets and 6 arrivals of second category packets. One of the terminals is exceeding the capacity so some packets are not served, 9 first category packets are served in a first. The remaining 3 slots are assigned to second priority packets.

First, there is a physical bound of the system capacity allocated per node. This bound is noted Bw so that $\sum_{T_s} \mathcal{S} < Bw \times N_{nodes}$, where $\sum_{T_s} \mathcal{S}$ is the sum of the packets size conveyed by the transport level during a slot of length T_s. Secondly, in order to consider some QoS effects, the following bounds have also to be respected i) $\sum_{T_s} \mathcal{S}_{\text{node}=i\,,QoS=1} < Bw$, for $i = 1..N_{nodes}$, that is limiting the output of first category packets transmitted per node ii) $\sum_{T_s} \mathcal{S}_{QoS=2} < Bw \times N_{nodes} - \sum_{i=1}^{N_{nodes}} \sum_{T_s} \mathcal{S}_{\text{node}=i\,,QoS=1}$ that means that packets from the second category are assigned to all the capacity that was not taken by first category packets.

The model of the transport level using the operators is important because they have to describe a major modification of the final stream structure.

The traffic stream $\Phi^{(4)}$ can be equivalently represented by the series $\{N_1\}$ and $\{N_2\}$

4.2.3.1. Operator Addition $\oplus$

The operator $\oplus$ at the transport level has a different implementation than the previously described operators. Indeed the transport level is considering the type of QoS classes before the insertion in the frame and the output of the transport level has a bounded capacity.

First, the addition operator for first QoS type is explained. Each node has a certain amount of cells reserved per frame. As explained earlier the addition is performed in line with the individual constraints (at each node). The sum of each flows in then the addition of the traffic of each node up to the maximal node level, with addition of cells that were previously delayed because of insufficient capacity. Afterwards, the total number of cells transmitted is this first category can be computed and the number of free cells can be evaluated. This number of free cells is then the maximum capacity available during this frame for cells of the second category.

Then, secondly, the action of the addition operator consists of collecting available cells of the second category up to the obtained maximum. If this maximum is reached before all node have been served, a round robin scheme is fairly collecting cells of each of the queue of each node. To increase the fairness, the starting node is drawn randomly at each frame between the non-empty queues.

4.2.3.2. Operator Level conversion $\triangle$ and the operator Level conformation $\sqsupset$

Part of the actions required by these two operators are taken into consideration in the addition scheme that was described previously.

The output of the level conversion operator has to be compliant with the format of the transport level. The constraints have been exposed before and they are more related to a proper insertion of the packets in the frame. No constraints in the format are taken into consideration except the constraints described in the addition operation (individual node constraints for first QoS cells and overall capacity constraint for the second QoS cells). Hence the action of the level conversion will be considered as vooid.

The action of the level conformation operator is modelled by the modification of the single streams composed of $\mathcal{P}_3$ packets into a concatenated form describing the composition of the frame. This operator is considering the limits on the number of packets to be outputted and their corresponding types by the construction of the addition operation. In the simulator, both action are realized in a single operation.

For coherence, the ”transformation” from the node level to the transport level will be written:

$$\Phi^{(4)} = \left(\Phi_1^{(3)} \oplus \Phi_2^{(3)}\right)^{\sqsupset_{3\to 4}}$$

4.2.4. Other operators

In order to measure end to end metrics, some operations are required to retrieve the packets in their original format as they were emitted (usually at the application level). The simulator then also includes modules to perform these operations. These operators are building from a stream from level $\ell + 1$, a stream at level ℓ, reconstructing lower level messages from the header information. This part of the simulator is called the inverse pyramid because it realizes the counterpart of the operations described in the pyramid description.

4.3. Investigation of aggregate models with a simulator

4.3.1. Justification of the simulation approach

The use of a simulator is first justified because a real deployment of the system is not feasible at low cost and rapidly. Also, the choice of a simulation approach permit to focus on the dimensioning problem. Simulation are also easily reproducible, and enable a fine tuning of the system. They can be adapted to a different satellite transport level, permit to isolate different problems. As a conclusion, a traffic simulator reveals to be a handy tool in order to investigate the behavior of a satellite system under different load conditions

4.3.2. Design principles

The implementation of the simulator was performed with a discrete event simulator (DES). There exist different DES simulators for telecommunication networks like Opnet [O205] or NS2 [ns205]. In such simulators, the management of time is performed transparently by proper management of the future events send. All objects are implemented with a state machine, that react to the occurred events. The OMNET++ [Var01] has been retained because of its modularity and its portability and previous study had been performed with this simulator [Bou05]. A set of toolboxes are complementing OMNET for the study of networks (for the IPSuite package with a TCP implementation). In [Wat93], more details on discrete event programming can be found. In the description of the traffic generation modules, the state machine and the event generated will be described.

The simulator is programmed at three planes using different files with their own syntax that have different functions. Presenting some details will help to understand the implementation of the simulator:

- Models implementation (c++ files), there the models are implemented (parsing of parameters, generation of packets...). There all mechanisms for packet processing are also defined, the functionality required for packet level conversion.
- Models architecture (ned files), there the architecture between the models are defined and the connections between the modules are performed.
- Scenario definition (ini file), here the parameters for a particular scenario are defined.

The model implementation (in c++) corresponds to the translation in computer language of the theoretical models presented in chapter 3. If no other models are required, the modules corresponding to each modules don't need modification. If a characteristic of the generated packet needs to be modified, this can be done in the model implementation.

The model architecture is defined in the ned language. It enables to describe the organization of the different modules and submodules. For the simulation of particular scenario, in order to obtained the proper organization of module a ned file is required. For example in this study, four different ned files were produced for the node model, depending of the traffic models to be used inside and the configurability required.

The parameters used for a single simulation are modified for every run via a new ini file. Each modification of the ini file produce a new configuration of the simulator. Scripts have been produced in order to set scenario with variable number of sources, that enable the generation of results with different number of passengers for example.

4.3.3. Expected benefits and limitations of the approach

The goal of the simulations is to demonstrate the use of equivalent traffic models. In order to show that agreements can be obtained between original and the suggested equivalent in term of the generated streams, both cases can be simulated to enable the comparison and the results from a detailed simulation can be used in order to obtain the parameters for the second model. As such a simulation tool seems very useful in order to investigate and optimize traffic model equivalencies.

But of course, the usage of a simulator has also some drawbacks. Firstly, the simulator can have poor performance in the time domain (for example if the models are getting slower that the applications that they model). The level of details must be chosen according to the investigated phenomena. If the level of details is too high, the complexity of the simulator will imply that long simulation time are necessary to collect significant results. Secondly, the use of random generators may imply that the simulator has a deterministic behavior and hence an high number of simulations may be required to get reasonable interval of confidence. Thirdly the approach is well suited for existing applications, but in the future, new services may appear with very different structure with possibly complex peer structure that will make the individual scaling of the application unrealistic.

As expected benefits for the proposed approach, it comes

- Reduction of number of sources: Through the use of aggregate models, it is possible to reduce the number of users that are simulated because one model is synthesizing the overall traffic. For example, if a node model is considered, if the node is composed of 100 passengers with 2 applications each, there is a potential gain of 200. Nevertheless, a gain in events can hardly be realized as will be shown in the analysis of complexity.

- Scalability: Aggregate model brings the advantage that these models can be easily scaled down or up. This property can be helpful for system design. For example after having determined an aggregated model for a node the number of nodes can be determined in order to reach a given constraint of system parameters. With a single node model, it is easy to determine the maximal number of nodes that offers voices services with a minimal delay bound. This gain is obtained because models at each level are self-contained. It then easy to modify the model at a particular level. For example, if the system behavior needs to be investigated with 10% more traffic, it can be choose to increase the number of nodes or subscribers or applications. Increasing the number of nodes will let the node model identical, but the satellite will be stressed differently because of the supplementary nodes. Increasing the number of applications or subscriber will require the determination of a new node model. Once available, it can be utilized as before. On the other hand, the effect of traffic increase could be also seen by reducing the bandwidth allocated per node.

As a conclusion, even of some drawbacks are present, there are overwhelming benefits that can be obtained from an approach based on simulations. For these reasons, a simulator was developped that will now be described.

4.3.4. Simulator architecture

4.3.4.1. Simulator for traffic investigation

Our objective is to build traffic models for each level of the defined pyramid. From the previous approach, a hierarchical traffic generation model can be constructed that will replace progressively all the levels by equivalent models.

The architecture of the simulator is depicted on figure 4.7. There are entities for each of the levels

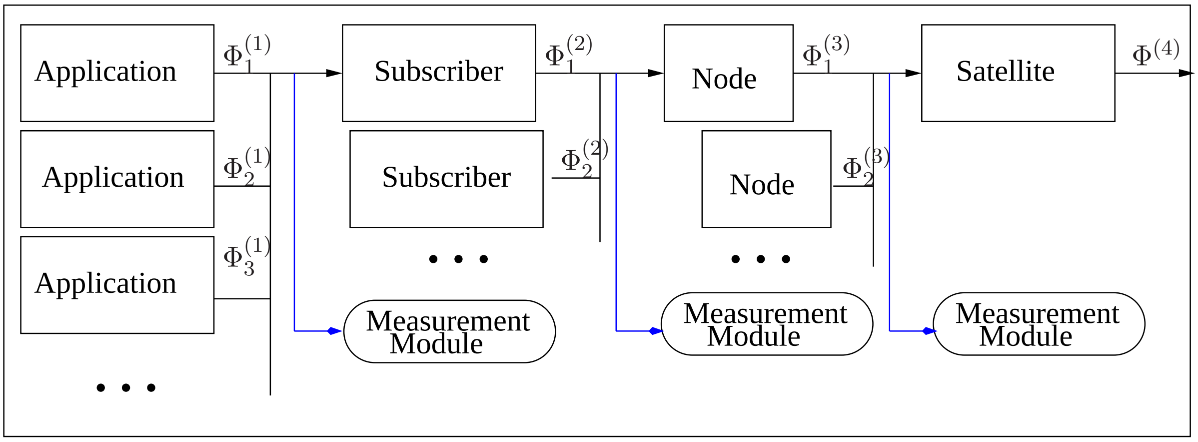


Figure 4.7.: Architecture of the simulator

(application, subscriber, node and transport levels). Measurements modules are included to evaluate some metrics useful for the model parameters estimation. The simulator structure is hierarchical and follows the distinction of the different pyramidal levels introduced in the previous chapters. At the application level, the modules responsible of the traffic generation are the module implementing the single traffic models.

Objectives: The objective is the characterization of the traffic streams at the transport level $\Phi^{(4)}$ and the elaboration of an independent generator capable of generating this traffic stream. This characterization is done level by level.

For this purpose, there is the need to implement traffic generators at the application level and all the mechanisms to process packet up to the transport level. These modules realize then the operations described by the operators that were described in chapter 2, and will be described now.

So the required functions of the simulator are:

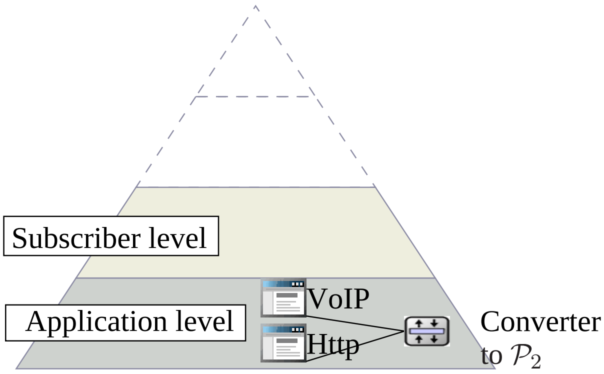


Figure 4.8.: First level of the simulator

- Models for traffic generation: The simulator shall be able to generate traffic according to all models that were discussed in chapter 3
- Operator of the aggregate traffic pyramid: the simulator shall be able to handle the different packet format defined for each level of the pyramid, and to be able to convert them from one format to the other one.
- Evaluation of performance metrics: the simulator shall also include the modules for evaluation of performance criterions as they were defined in chapter 2.

4.3.4.2. Postprocessing and parameter matching routines

When results have been obtained, the obtained results can be further processed to obtain more advanced analysis (derivation of density, summary of different runs). For the analysis of the simulation and the matching of parameters, Matlab was used. On one hand, it is possible to import the results obtained in the simulator in particular when binary files are recorded. On the second hand, some calculations can be conducted. For the implementation of the EM algorithm, better performance could be achieved with an implementation in C, so was first motivated by the use of routines for handling high precision big numbers modules.

4.3.5. Description of the simulator inside the pyramid

The purpose of the simulator is to validate the correct equivalency i.e. to show that a complete pyramidal un-aggregated scenario can be replaced ideally by a unique traffic model or at least a model with less levels. In this view, a traffic scenario with four levels corresponding to the traffic generated inside an aeronautical communication system has been chosen. This scenario will be described with the notations introduced in the previous sections.

4.3.5.1. Application level

At the application level, two applications are considered: the first one is a voice over IP application (telephony) and second one a web surfer (uplink traffic from a web browser). Each of the applications is generating its own $\mathcal{P}_1$ packets.

Figure 4.8 shows the first level of the simulator. The following traffic models have been considered. The voice over IP source is generating fixed size voice packets in accordance with a particular voice codec. (cf. Table 3.1 in 3.2.1.1). An appropriate model was suggested in chapter 3. This is generating a flow $\phi_1^{(1)}$. The web browser is using a BMAP ($d = 3, m = 4$) model that was explained in 3.1.2.1. It is derived from measurement from a classical web session. The parameters are derived from the web packets size and time of arrival (the conversion to IP packets will not require a fragmentation of the

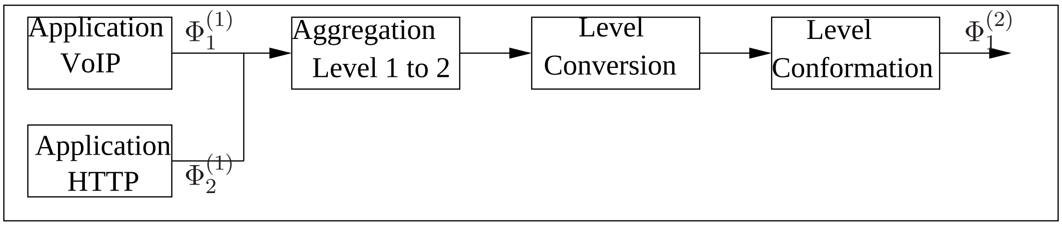


Figure 4.9.: Construction of the subscriber level flow from the application flows

packets). The following flows have to be considered: $\phi_1^{(1)}$ (VoIP) and $\phi_2^{(1)}$ (HTTP). Afterwards, there is the need of the conversion of the application packets into $\mathcal{P}_2$ to provide outputs compatible with this format. The box "converterIP" on figure 4.8 is indicating the conversion to $\mathcal{P}_2$ packets. This box is realizing the operation $\triangle_{1\to2}$ described previously on the flow $\phi_1^{(1)} \oplus \phi_2^{(1)}$. The input towards the next level can then be written

$$\phi_1^{(2)} = \{\phi_1^{(1)} \oplus \phi_2^{(1)}\}^{\triangle_{1\to2}\sqsupset_2} \tag{4.1}$$

The operation is depicted in the synthetic figure 4.9. The aggregation operation is written with the operator $\oplus$, the operator $\triangle_{1\to2}$ represent the action of level conversion operator and the operator $\sqsupset_2$ realizes the level conformation operation.

For the modelling of the operator $\sqsupset_2$ (conformation to the level $\ell = 2$), the following elements have to be taken into consideration:

- VoIP packets must have an higher priority as the HTTP packets.
- Conversion of VoIP in IP packets: addition of an header (this header is making use of compression methods)
- No fragmentation because $\mathcal{S}_1 < MTU(\mathcal{S}_2)$

4.3.5.2. Subscriber level - Passenger (abstract stream $\Phi^{(2)}$)

The subscriber level is representing the traffic generated by a passenger of an aircraft. A traffic model at this level has to model the $\mathcal{P}_2$ packets. There are two possibilities to generate this traffic flow $\Phi^{(2)}$. The first one is an independent generator that is directly generating $\mathcal{P}_2$ packets and the second one is obtained from the aggregation of the output of the traffic models at the application level.

Figure 4.10 shows the second level of the simulator. The flows of both applications are concentrated to the subscriber level.

4.3.5.3. Node level - Aircraft

The node level is representing the traffic generated by a whole aircraft. A traffic model at this level has to model the $\mathcal{P}_3$ packets. It is the flow that was noted $\Phi^{(3)}$. Here similarly the flow is constructed from the inputs $\Phi_i^{(2)}$ where the index i indicates the different passengers inside the considered aircraft (N_{pax}). The resulting flow ($\bigoplus_{i=1} \Phi_i^{(2)}$) is the resulting aggregated traffic in $\mathcal{P}_2$ packets.

The node has also to cope with the mechanisms related to the fact that the bandwidth offered on the satellite system is limited. It has been assumed that the node is providing to a central entity (NOCS or TLMU) estimates of its actual resource requirements.

For the modelling of the operator $\sqsupset_3$ (conformation to the level $\ell = 3$), the following elements have to be taken into consideration:

- Identical priority for every node (homogeneous case).

4.3.5.4. Satellite level - Overall traffic

A traffic model at this level has to model the $\mathcal{P}_4$ packets.

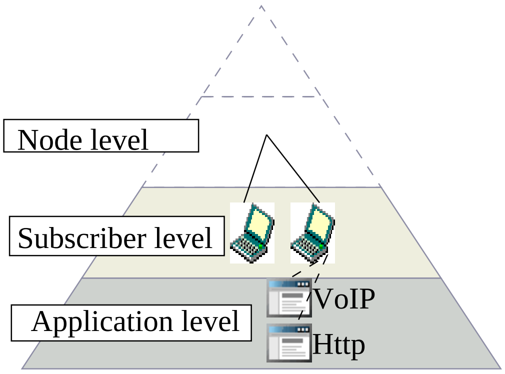


Figure 4.10.: Second level of the simulator

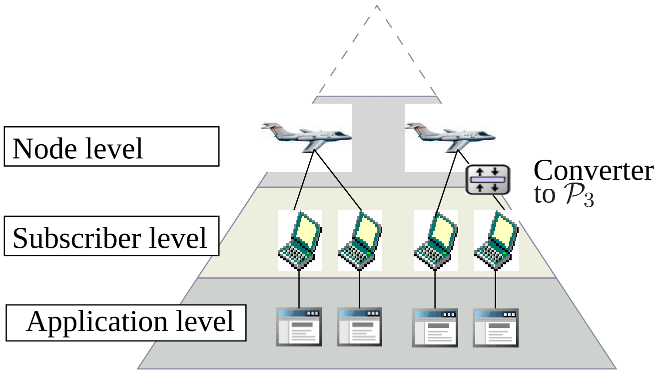


Figure 4.11.: Third level of the simulator

In addition to the bandwidth management performed between the node and the TLMU, the access to the physical resource needs also to be modelled. For this reason, the frame content is described at a whole as the result of the three operators (or two operators since the level conversion operator has no particular action because of this modelling based on the frame structure). Hence the modelling of the operator $\oplus$ (addition) and the operator $\sqsupset_4$ for the level $\ell = 4$ is based on the implementation of an addition operation in line with the earlier description and confirming for individual constraints for the first QoS packets and to an overall (and secondary) constraint for the second QoS packet type.

4.3.5.5. Cost of packet conversions

The conformance operator is modifying the packet size, but the sequence of inter-arrival times are not modified.

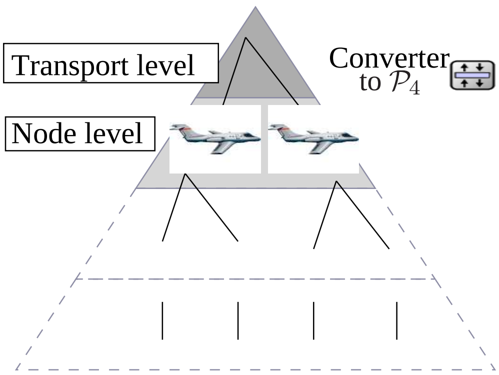


Figure 4.12.: level 4

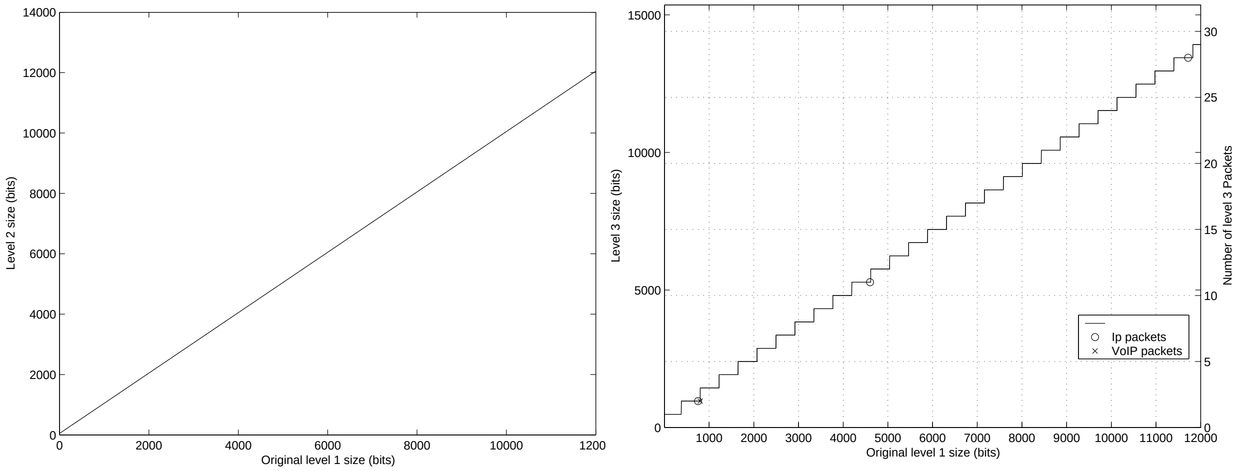


Figure 4.13.: Size of level 2 and level 3 packets vs size of level 1 packet size

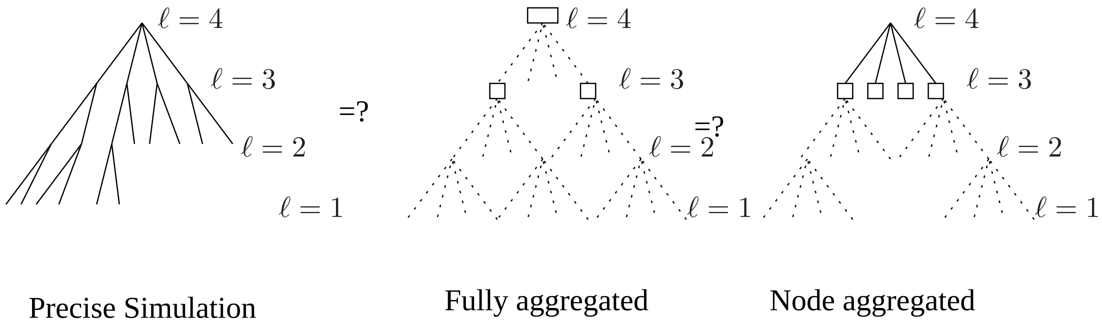


Figure 4.14.: Principles of equivalency of traffic models with different aggregation levels

The only effect of the level conformance operator is then an increase of the overall data volume because the overhead information included in the header has also to be included.

Figure 4.13 is showing the size of the $\mathcal{P}_2$ and $\mathcal{P}_3$ packets depending on their $\mathcal{P}_1$ size. For the size of $\mathcal{P}_2$ packet, a straight line is obtained, the effects of the operators are minimal and implied in the conversion from level 1 to level 2 is quite minimal. For the size of the $\mathcal{P}_3$ packets, because of the fixed size packets, the amount of output is a step function corresponding to packets filled with padding content in order to maintain fixed packet size. For the packet size considered in the different models, the number of $\mathcal{P}_3$ packets can be read. For VoIP packets, 2 $\mathcal{P}_3$ packets are necessary. For medium size web packets (from the web only model), 11 $\mathcal{P}_3$ are generated.

4.3.6. Scenarios for investigation of traffic equivalences

Here, the cases for equivalence of traffic model are defined.

The different way to model the overall traffic flow at the satellite $\Phi^{(4)}$ can be considered

i) as a independent traffic generator

ii) as a dependent traffic flow using independent generator at the lower levels.

The following formula is describing how the transport level traffic flow can be obtained from the lower levels generators.

$$\Phi^{(4)} = \left\langle \oplus_{k=1}^{n_n} \Phi_k^{(3)} \right\rangle_3^{\triangle_{3\to4}\sqsupset_4} \tag{4.2}$$

$$= \left\langle \oplus_{k=1}^{n_n} \left\langle \oplus_{j=1}^{n_s} \Phi_j^{(2)} \right\rangle_2^{\triangle_{2\to3}\sqsupset_3} \right\rangle_3^{\triangle_{3\to4}\sqsupset_4} \tag{4.3}$$

$$= \left\langle \oplus_{k=1}^{n_n} \left\langle \oplus_{j=1}^{n_s} \left\langle \oplus_{i=1}^{n_a} \Phi_i^{(1)} \right\rangle_1^{\triangle_{1\to2}\sqsupset_2} \right\rangle_2^{\triangle_{2\to3}\sqsupset_3} \right\rangle_3^{\triangle_{3\to4}\sqsupset_4} \tag{4.4}$$

where $\langle \ldots \rangle_\ell$ is representing the flow at the level ℓ, n_n the number of nodes, n_s the number of subscriber per node, n_a the number of application per subscriber.

The formula 4.2 is representing the aggregation of the transport level traffic flow from the node level traffic flows, formula 4.3 the aggregation of the transport level traffic flow from the subscriber level traffic flows, formula 4.4 the aggregation of the transport level traffic flow from the application level traffic flows.

Three cases are defined:

1) a multilevel case where independent traffic generators are used at the application level using 4.4
2) a single level case, where a single traffic generator is used to generate the traffic flow $\Phi^{(4)}$.
3) a mixed case, where generator at different levels are combined

In chapter 7, the node models are described with types ranging from I to IV. It is indicated here if the nodes components are using one of this type.

4.3.6.1. Not-aggregated case ($50\times$type I)

This is the case where no simplifications are considered.

The considered parameter are.

- Level 1
  - Application 1: VoIP
  - Application 2: Http
- Level 2
  - Number of subscribers: 100
- Level 3
  - Number of aircraft: 50
  - Resource Scheme: based on traffic estimation
- Level 4 (Generic GEO1)

4.3.6.2. Aggregated case ($50\times$type III)

- Level 3
  - Model: OBM1
  - Number of aircraft: 50
  - Resource Scheme: based on traffic estimation
- Level 4 (Generic GEO1)

4.3.6.3. Mixed case

- Level 1
  - Application 1: Http (application to be evaluated)
- Level 2
  - Number of subscribers: 1
- Level 3 Precise simulation

- Number of aircraft: 1
- Level 3 Aggregate Model from previous case
  - Number of aircraft: 49
  - Resource Scheme: based on traffic estimation
- Level 4 (Generic GEO1)

4.4. Description of the principal modules of the simulator

In this section, the implementation is roughly presented in order to give a background on how the questions dealing with implementation have been solved.

4.4.1. Traffic generation modules

In this paragraph, two examples of traffic models are described in order to show how they are implemented using the discrete event approach. The modules for traffic models are taking advantage of the Discrete Events capabilities of the simulator kernel. Using this approach, it is easy to implement every models. The future events are created in the future event set (FES). The kernel is then responsible of the proper scheduling of these events. It is taking care of reactivating them at the time the future has become the present.

4.4.1.1. Voice Model (on off with exponential duration)

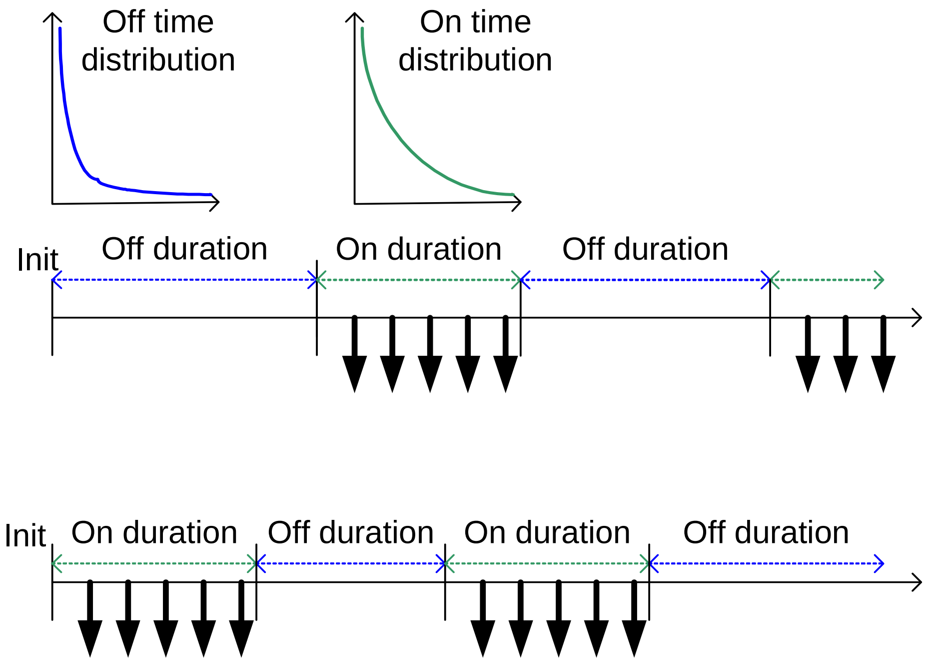


Figure 4.15.: Principle for the implementation of on/off models

For the implementation of the voice model the principles are depicted on figure 4.15. At model initialization, the first state is drawn. The duration of each of the duration is then drawn from the exponential distribution with the corresponding mean duration. If it is an ON period, the number of packets to be send is calculated and the corresponding packets are sent. At the end of each period, the state is switched. In a discrete event approach it is implemented easily with an event indicating the transition between the on and off period. The following short extract of code shows the implementation of such model. In the initialization procedure, a transition event is created and the model parameters and variables are set. In the example, the procedure called at event occurrence is shown.

```
handleMessage(cMessage *msg) {
    ASSERT(msg,"OnOff Change");
    if (state=StateON) {
        Toff=exponential(meanToff);
        state=StateOFF;
        scheduleAt(msg,now()+Toff);
    }
    else {
        Ton=exponential(meanTon);
        Npack=floor(Ton/T);
        for ( unsigned int i=0; i<Npack; i++)
            schedulePacketTx();
        scheduleAt(msg,now()+Ton);
    }
}
```

The only difference in the implementation is that in the first ON state when it is the starting state is ON there are no packet transmission. It is to prevent that all initial transitions occurs at $t = 0$.

4.4.1.2. Web model (BMAP(3,3))

If a BMAP(3,3) with 3 packets sizes m_1, m_2, m_3 and with 3 states called (1), (2) and(3) is considered, the elements of the matrices D_0, D_1, D_2, D_3 have the corresponding interpretation: $D_0(1,1)$ is the mean duration in state (1), $D_0(2,2)$ in state (2), $D_0(3,3)$ in state (3).

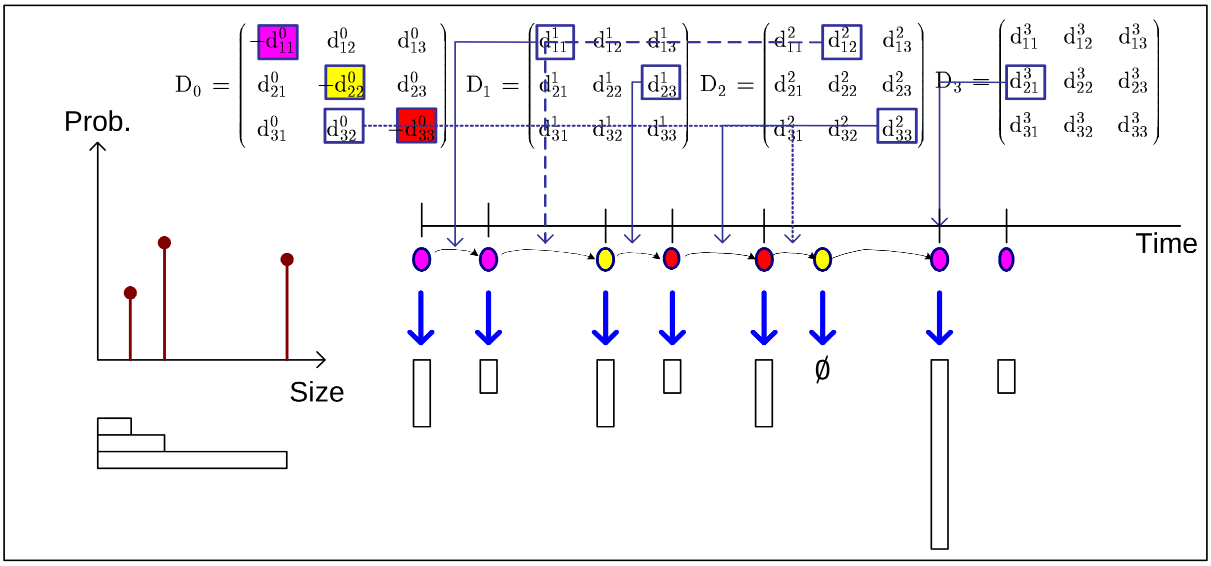


Figure 4.16.: Principle for implementations of the BMAP model

Figure 4.16 shows the principles for the implementation of the BMAP model. The three colored circles represent the different states, the different rectangle the packet transmission with the appropriate size. Event with no packet transmission are indicated with $\emptyset$. The duration between events in each state is dependent of the diagonal elements of matrix D_0. For each event, the type is derived from the parameters of the corresponding ”state” line (without considering the diagonal term of D_0). The possible outcomes are a transition to an another state, a transition with a packet of size 1,2 or 3 (with a possible transition to the same state). In the graph, the transitions to states (1)-(1)-(2)-(3)-(3)-(2)-(1)-(1) have occurred.

4.4.2. Analysis modules

The modules for analysis are implementing:

- the queueing modules
- the end-to-end packet performance analysis module.
- the packet recorder.

The theoretical principles for the queueing module can be derived from the results established by Pollazek and Kinchine. Two different modules were implemented. The first one is the packet based queueing module where a fixed duration for the service time is assumed. The second one has a service time depending of the length of the incoming packet, so this module is characterized by a service time per bits. For the first case, a constant service time is used, so queueing results for the $M/D/1$ queue are of interest. In the second case, the same service distribution is used if the incoming process is considered with variable size packets. Queueing theory results are now exposed.

4.4.2.1. Queuing

Queueing theory consider customers arriving at a service facility, in particular the instants when they arrive, pass through and hopefully leave the system. First, the arrival process needs to be described. It is done with the probability distribution of inter-arrival time of customers $\Pr(\mathcal{I} \leq t) = A(t)$. Then, the amount of work requested by the customer is called the service time. Its probability distribution is noted $B(x)$. A queue has also a waiting room of capacity K where K could be infinite. Finally, a queue has also a discipline, that describes in which order customers are taken into service. Standard disciplines are first-come first-serve (FCFS), last-come first-serve (LCFS). The discipline may also consider some priorities between the customers. Queuing theory will give indications on the waiting time of a customer, the number of customer ion the system, the length of a busy period etc.

An important result when queues are considered are the relations established by Little [Lit61], who has proved that the average number of customers in the queuing system is equal to the averaged time spend in the system T times the average arrival rate of customers λ.

$$N = \lambda T \tag{4.5}$$

When only the waiting room is considered, it comes that

$$N_q = \lambda W$$

where N_q is the mean number of customers in the queue and W is the average waiting time. The waiting time W is the time spent for queueing, it is the time spend in the system (T) minus the time necessary for service ($\bar{x}$), $W + \bar{x} = T$. An important parameter is the parameter ρ called the utilization factor and defined as follows:

$$\rho = \lambda \bar{x} \tag{4.6}$$

In the previous term, $\rho = N - N_q$, and corresponds to the average number of customers being served. Stable queues require that $\rho < 1$. The background for the analysis of queueing systems are presented in [Kle76a] and [Kle76b]. The principal definitions and results are recalled hereafter, at least for the simplest queues.

In figure 4.17, the definitions useful for the analysis for a queueing system are depicted for the case of a first-come first-served server. These definitions are valid for any $A/B/m$ queuing system. C_n represents the nth customer to enter the system. The sequence $\{\tau_n\}$ represents the arrival time of this customer in the system . The sequence $\{t_n\}$ represents the inter-arrivals time between customers C_{n-1} and C_n. The inter-arrivals time are drawn from a distribution $A(t)$ (independent of n) such that $P(t_n \leq t) = A(t)$. At their arrival, the customers are either served if the system is empty (like customer C_{n+2}) or wait in a queueing facility. When a customer has finished its service, a waiting customer can be served. The sequence $\{x_n\}$ represents the service time for C_n and the sequence $\{w_n\}$ represents the waiting time

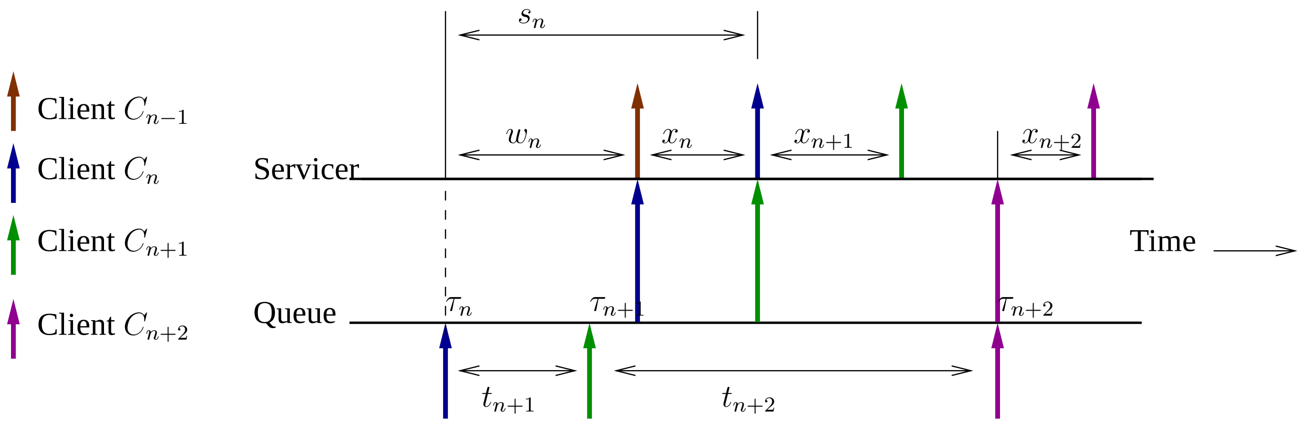


Figure 4.17.: Definition for a queuing system

for C_n. The time spend in the system s_n is then $w_n + s_n$. The service time is assumed drawn from a distribution $B(x)$ (also independent of n)such that $P(x_n \leq x) = B(x)$.

To solve queues of the type M/G/1 (Poisson arrivals, general service law and single server). In this case, the method of the imbedded Markov chain can be used to ”solve” the queue. Two sequences are introduced $\{q_n\}$ and $\{v_n\}$ which represent respectively the number of customers left behind by departure of C_n from service and the number of customers arriving during the service of C_n. The evolution of q_n can be described by the following equations

$$\begin{aligned} q_{n+1} &= q_n - 1 + v_{n+1} \text{ if } q_n > 0 \\ q_{n+1} &= v_{n+1} \text{ if } q_n = 0 \end{aligned}$$

If the time of service is constant ($B = D$), v_{n+1} can be easily obtained by a counter of the received packets in the queue. So the sequence q_n is obtained, via the Pollaczek Khinchin equation.

$$\bar{q} = \rho + \rho^2 \frac{1 + C_b^2}{2(1-\rho)} \tag{4.7}$$

where C_b^2 is the squared coefficient of variation for the service time ($C_b^2 = \frac{\sigma_b^2}{(\bar{x})^2}$)

The main results are:

- Queue size distribution: the queue size distribution can be obtained from the Pollaczek Khinchin transform equation.

$$Q(z) = B^*(\lambda - \lambda z) \frac{(1-\rho)(1-z)}{B^*(\lambda - \lambda z) - z} \tag{4.8}$$

- Waiting time: the Laplace transform of the waiting times can be obtained

$$W^*(s) = \frac{s(1-\rho)}{s - \lambda + \lambda B^*(s)} \tag{4.9}$$

These methods have been extended later to solve the BMAP/G/1 queues and provide useful metrics in the study

4.4.2.2. IAT and evaluation of autocorrelation of IAT

The procedure for the evaluation of Inter-Arrival Time is explained with all details useful for implementation is described in the appendix A(1).

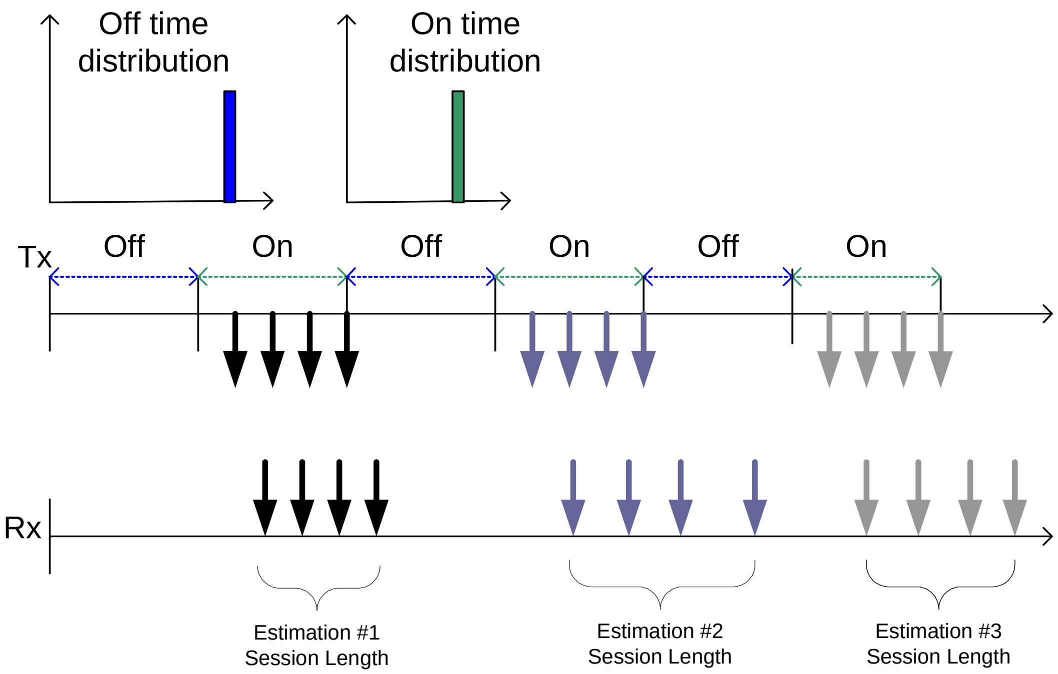


Figure 4.18.: Estimator of session length

4.4.2.3. Estimation of delays

In figure 4.18 the method for estimation of the web session length is explained. It requires two elements: a dedicated transmission model and a module that analyzes the received packets. The transmission model is a modified on-off model (derived for example from the BMAP on-off model), with fixed duration of the on-off period. The sent packets have an identifier counting the on periods. These on periods are assimilated to the active session of a web request. At reception, the information is used to estimate the session duration computed by the difference between the first and last packet of a same on-period.

4.5. Characterization of simulator outputs

4.5.1. Metrics collected by the simulator

In this section the collected parameters from the simulator are described

Traffic parameters

i) at Application level

The generation time of each packet is recorded together with the AppID identifying the application. It enables to get the sequence of the $\{\mathcal{I}\}$ from the traffic stream $\Phi^{(1)}$. If the models is emitting packets of different size, these size are also recorded.

ii) at Subscriber level

The packet arrival time is recorded. The subscriber node queue is receiving a backup of the generated sequence.

iii) at Node level

The packet size, the number of packets, the arrival time of the packet are recorded for the node$\#0$

iv) at Transport level

The thruput is estimated, it is computed on a regular basis from the volume of data transmitted during the period of evaluation. The total number of cells served of QoS class 1 and 2 (3 and 4 are not used in the scenario) are recorded for every frame. The packet size, the number of packets, the arrival time of the packet are recorded. The sequence of the $\{\mathcal{S}, \mathcal{I}\}$ from the traffic stream $\Phi^{(4)}$ is then collected.

Performance parameters

i) inverse pyramid

The inverse pyramid is receiving the packets after their transmission through the transport level. At each level, it can be selected which packet are further converted (a destination node, subscriber destination and applications destinations) (a list can also be recorded).

ii) application sink

The application sink processes i) QoS information if the model is including this kind of information (this a dedicated handling of page number for the computation of page duration and size estimation) ii) all message for the record of packet size and end to end delay.

So, it is possible to estimate the distribution of session duration and of delay. These are performance estimators considered in the numerical study.

4.5.2. Performance indicators and investigations

Performance indicators During a simulation execution (run in cmd mode), the simulator is displaying every 100000 events the following information

- event number,
- Simulation time in the simulated system
- Elapsed time in the computer
- ev/sec current estimate of the number of event per second

So after simulation it is possible to get indications about the performance of the simulations.

Discussion of events: The number of event is also depending of the number of observation modules that are present in the scenario. For example, the queuing modules are generating events for the next inspection (at arbitrary time) and at the end of packet service.

Investigations of performance The simulator is furnishing during execution of a scenario indications of the number of events generated.

The advantage of an aggregate model is that it reduces the number of entities required. In an event driven approach, if every packet transmission is an event, between the original model and its aggregate equivalent, the number of events shall be similar. If the aggregate models (like the MMPP for example) have transitions that are not related to packet arrivals, new events due to a state transition of the associated Markov process are introduced in the simulator which are decorrelated from packet generation events. This tends to increase the number of events in the aggregated model with respect to a classical renewal model.

In general, the simulator shall generate constantly the same number of events (large scale mean, of course local variations will occur). Otherwise if the number of events generated over time is decreasing, it means that the simulator is getting more and more difficulties to generate the same models, it is likely that one or more of the queue inside the simulator is getting saturated (or this is the more probable explanation). If a queue is heavily loaded, it will require to execute the simulation up to a point where the saturation has occurred and then to inspect the objects to detect where the saturation has occurred. Another point able to ”saturate” the system is an overflow of a variable playing a role in the generation of packet, this kind of error has its origin in a single model. For this reason, the implementation of a traffic model needs to be tested individually before to be used in the simulator.

In order to estimate the complexity of the models, it is needed to compute approximatively the number of events generated by the model. For example, for the voice model the number of events generated can be evaluated. It is recalled that this model in an on/off model with $T = 0.01$s, $T_{on} = 0.350$s $T_{off} = 0.630 (AF = 2.8)$. The number of events between two ON phase is then Nevents=35 (packets) + 2 (on off transitions) / (Toff + Ton). This corresponds to 38 ev/s. If 100 sources are needed, $N_{ev} = 3800$ ev/s. In order to simulate 150s, the simulator is using $21\,10^6$ events. The part required for the generation is $5.6\,10^6$ events. Some events are required for the packets conversion. Since the 4th level is based on a regular clock, the number of events can not be reduced below a minimum level.

Another point to consider is the number of events needed for the collection of metrics in the simulator. For example, it was shown that the autocorrelation estimation modules could have an expensive computational cost in particular if the module is feeded with an high number of packets.

Analysis of complexity For example if we have two applications per 100 subscriber at a each node, the precise simulation has a complexity $200 \times c_a$ where c_a is the complexity of the application model. An subscriber model of complexity c_s will have a node complexity of $100 \times n_s$ so it is required that $c_s < 2c_a$ to have a gain. When working at node level, the complexity of the node aggregate model brings a gain if the aggregate node complexity c_n is such has $c_n < c_a 200$.

4.6. Summary

In this chapter, the theoretical definitions from Chapter 2 have been applied for a case that will afterwards be studied numerically. This scenario was described with the parameters used in the remaining of the study. The actions of the operators were described with details of the practical implementation. The translation of this scenario into an appropriate simulator and the realization of this simulator was described. Now the obtained results will be shown and discussed.

5

Pyramidal aggregation of voice models

In this chapter, results concerning the aggregation of voices models are presented. The sources are assumed to be homogeneous and identical. Equivalent models can anyway be derived at the different levels of the pyramid. For the aggregation of voice sources, the use of an MMPP(2) model is suggested at node level. At the transport level, gaussian equivalents will also be investigated.

5.1. Overview of presented results

Figure 5.1 shows the scenario considered for the aggregation of homogeneous subscribers. At first an increasing number of subscribers with only VoIP traffic was considered. In the picture are depicted different organizations with or without the use of aggregate models. Scenario 1.1 is defined by the consideration of 100 voices sources. The input stream is constituted of $\Phi_i^{(1)},\ i = 1, 100$ where $\Phi_i^{(1)}$ is the model for a single voice source. Scenario 1.4 is defined as the aggregated model corresponding to 1.1. Scenario 1.2 is defined by the consideration of 99 voice sources. Scenario 1.3 is defined as composed of a single voice source and an aggregated model equivalent to scenario 1.2. If it is assumed that the

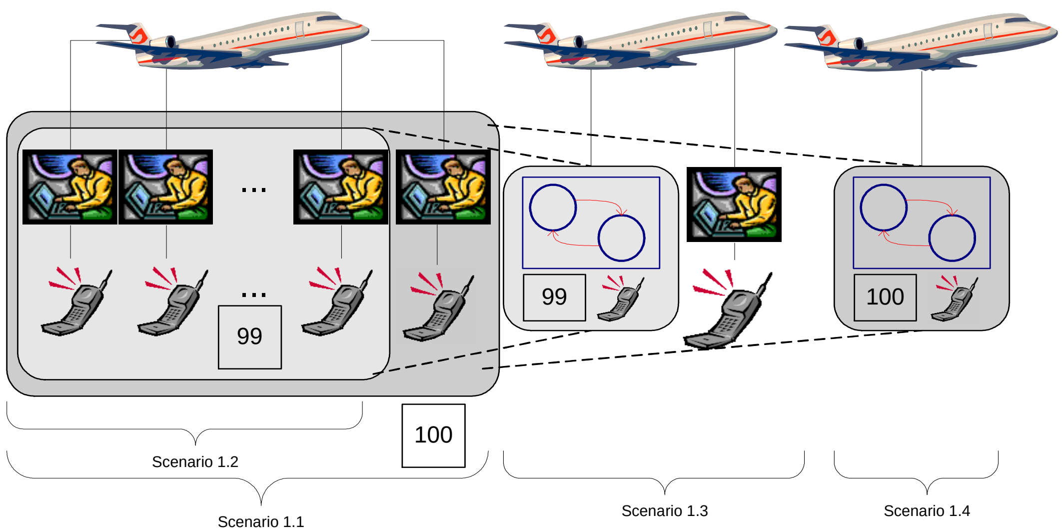


Figure 5.1.: Description of the first set of scenarios considered

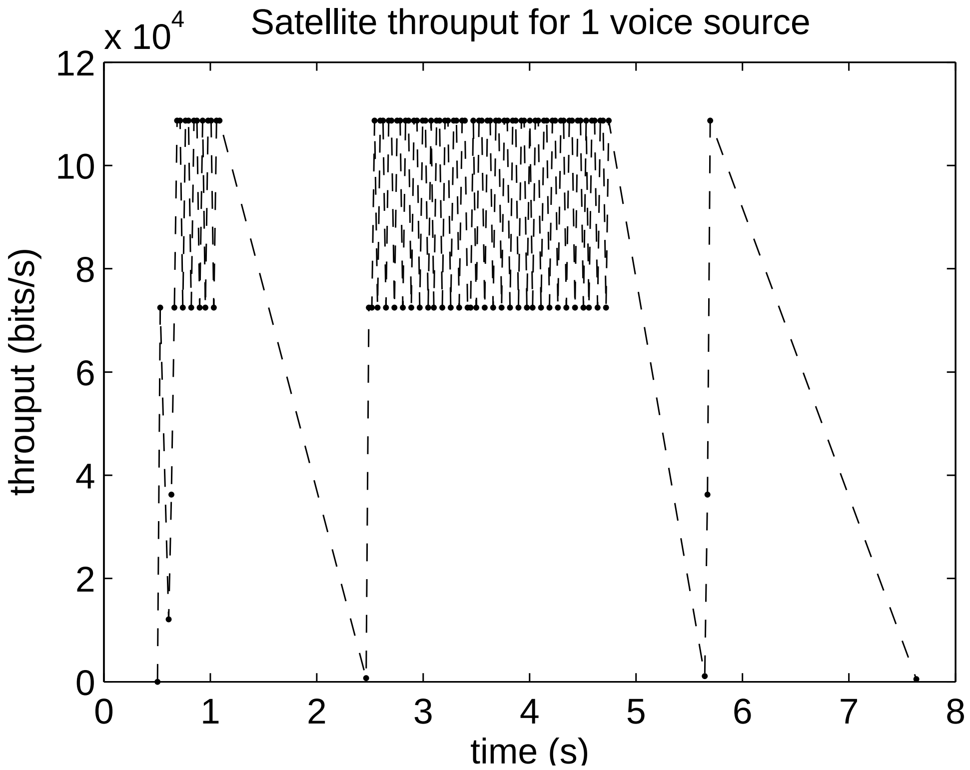


Figure 5.2.: Thruput for 1 voice source

aggregated models are generating a traffic stream at the level 1, following traffic streams can then be compared at the node level:

- $\Phi^{(3)}_{1.1}$ generated from $\Phi^{(1)}_{VoIP}(i)$ $i = 1, ..., 100$.
- $\Phi^{(3)}_{1.4}$ generated from $\Phi^{(1)}_{MMPP}(100)$
- $\Phi^{(3)}_{1.3}$ generated from $\Phi^{(1)}_{VoIP}(1)$ and $\Phi^{(1)}_{MMPP}(99)$

The traffic stream $\Phi^{(3)}_{1.4}$ is the aggregated stream equivalent to the traffic stream $\Phi^{(3)}_{1.1}$. They must have equivalent properties that will be investigated in order to prove the validity of the equivalency. The traffic stream $\Phi^{(3)}_{1.3}$ is also an equivalent model in which one application is kept in order to obtain performance metrics.

5.2. Pyramidal description for this first case

5.2.1. Application level

The voice model was described in 4.4.1.1. Figure 5.2 is showing the activity of voice source measured at the satellite level (throughput of stream $\Phi^{(4)}$). It was measured at the transport level. The maximum throuput is obtained at 10.8610^4 bits/s that corresponds to the transmission of 6 $\mathcal{P}_3$ packets per frame. A lower level of transmission rate can also be seen at 7.2410^4 bits/s that corresponds to the transmission of 4 $\mathcal{P}_3$ packets per frame. The original traffic had a peak rate of 100 $\mathcal{P}_1$ packets per second. The $\mathcal{P}_2$ traffic has similarly a peak rate of 100 $\mathcal{P}_2$ packets per second. This corresponds to 2.65 $\mathcal{P}_2$ packets per frame. Depending from the instant the peak is starting relative to the beginning of the frame 2 to 3 packets can be sent per frame. Since the size of $\mathcal{P}_2$ is 848 bits it fits exactly into 2 $\mathcal{P}_3$ packets. So between 2×2 or 3×2 $\mathcal{P}_3$ packets can be inserted in the frame during the ON period. The behavior of the single voice source reveals the importance of the peak data rate. It seems important to allocate resources up to peak data rate.

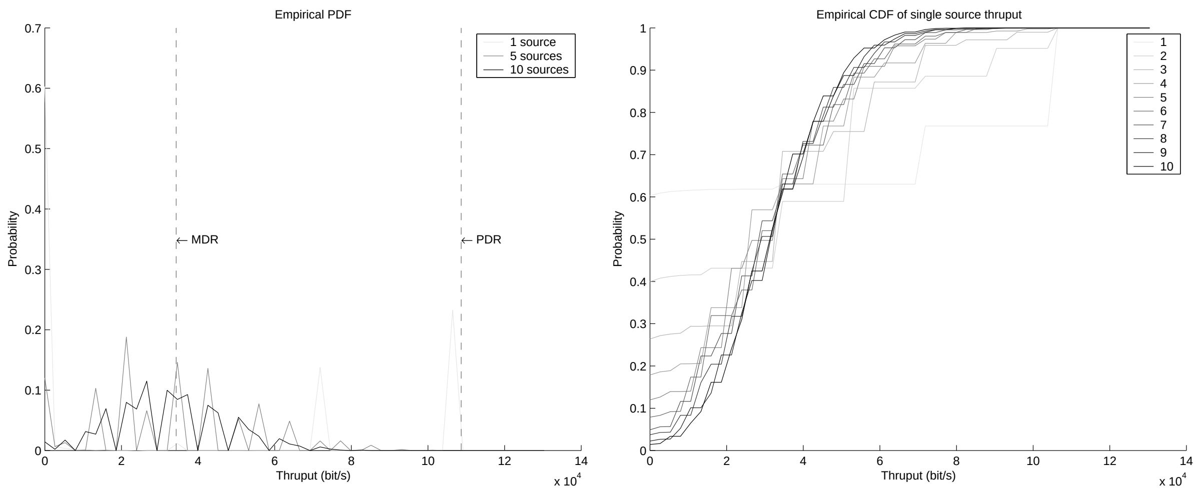


Figure 5.3.: Probability density function of the mean sources throuput

5.2.2. Subscriber/node level

In order to derive a MMPP(2) model to the aggregation of 100 voices sources, any of the method described in chapter 3 can be used. The methods based on the arrival functions, where indicated to have worst fit when used for the derivation. The method of Kang was used when the traffic volume was sufficient to get meaningful queueing statistic (i.e. to have occurrence of ”long” queues length). The EM method was also used, it revealed stable enough to perform iterative evaluations for increasing traffic load.

5.2.2.1. Properties of aggregated streams

Global behavior When the N_v voice sources are aggregated, the effect of ”averaging” should appear because the peak rate is reached at different instants. So a tendency towards the mean rate should appear.

Figure 5.3 is showing the probability density function of the mean throuput per source for the aggregation of 1,5 and 10 sources. The results where obtained at transport level. The pdf tends to a bell-shape when the number of aggregated sources increases, the mean is around the mean data size. For the single source, two peaks are seen for the throuput that corresponds to the 4 and 6 $\mathcal{P}_3$ packets.

The aggregation of homogeneous sources is reducing the effect of the peak value and an averaging effect can be evidenced. To confirm this, figure 5.3 shows also the CDF of the throuput for increasing number of sources. It can be seen that the CDF tends to a limit distribution (that is undoubtedly gaussian) so that the peak behavior does not matter so much.

This can be used to reduce the bandwidth allocation necessary when sources are multiplexed together. Even if, in the case of some models with long-range dependency, this multiplexing gain is not obtained, because big size packets arrivals can occur.

Autocorrelation When the renewal processes were introduced, it was mentioned that the aggregated process was not a renewal process anymore. This can be seen on figure 5.4 that represents the autocorrelation of the $\mathcal{I}$ for the 100 voices sources that are aggregated. There is a peak in this autocorrelation, that indicate a kind of period in the aggregated process. Clearly it should be caused by the next packet transmission of the same user if it remains in the On state. This is reached at $n = \frac{T}{\mathcal{I}} N_v = \frac{0.01}{0.027} 100 = 36$

In figure 5.5, the autocorrelation measured in the simulator for the aggregation of 100,110,120 and 130 is shown. The simulator is giving a sequence of autocorrelations $c(k)$ where k is the number of arrivals. When sequence is normalized by the mean arrival rate, the curves have a good superposition. The Fourier transform of the series has very similar shape with a peak, that corresponds to the damped

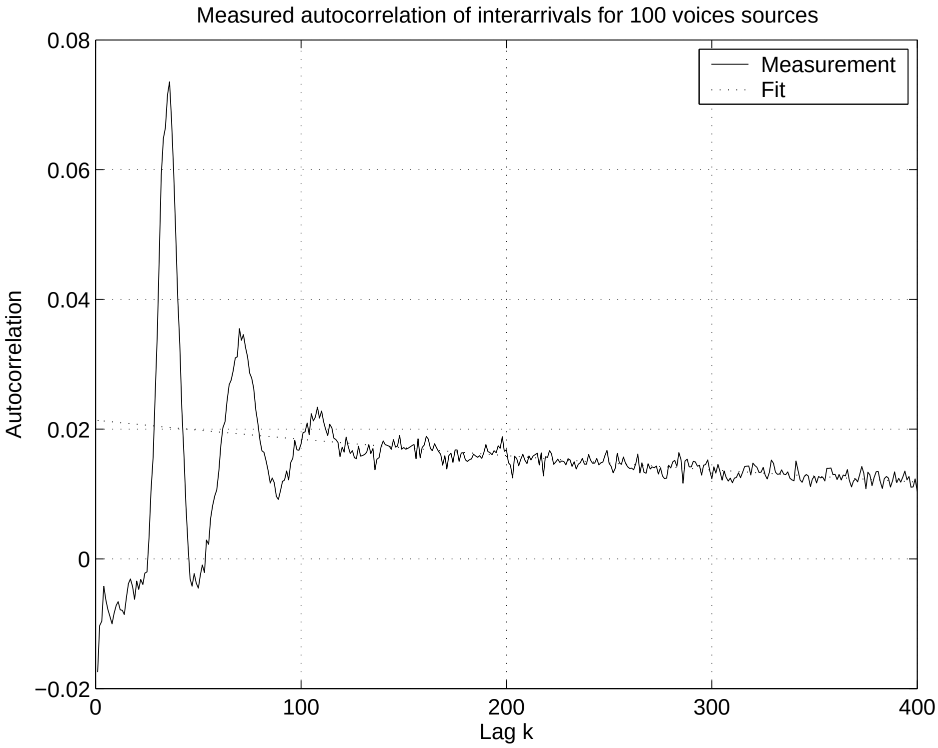


Figure 5.4.: Autocorrelation of the inter-arrival time for 100 voices sources

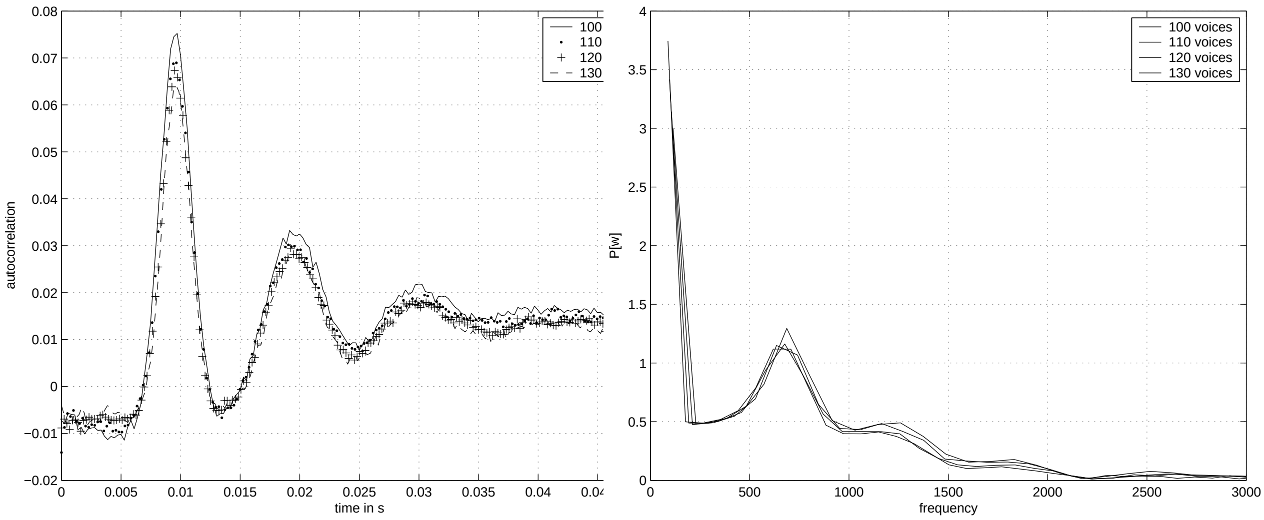


Figure 5.5.: Autocorrelation and power spectrum of the aggregation of 100,110,120 and 130 voices sources

periodic component of the autocorrelation. The peaks of autocorrelation occurs readily at $t = 0.01 = T$ and $t = 0.02 = 2T$ as explained earlier.

5.2.2.2. Parameters derivation

Different methods have been used to derive the parameters of the aggregated models from the recorded stream. From a record of $\Phi_{1.1}^{(3)}$, the parameter of the stream $\Phi_{1.4}^{(3)}$ are obtained. The methods that were used are:

- Based on increasing moments

Voice Sources	λ_1	λ_2	r_1	r_2
100	1.0661010	0.753172	1.59910e-3	3.3754e-6
110	1.0592671	0.816444	1.2421e-3	3.941 e-5
120	1.0372151	0.869265	4.8058e-4	1.17e-4
130	1.060082	0.899490	2.591e-4	2.3420 e-4

Table 5.1.: Normalised parameters of the MMPP(2) for different number of voice sources

- Based on asymptotics
- EM method (initial parameters determined by IPP, HMM (Hidden Markov Model))
- EM method (initial parameters determined by Kathleen MMPP)

In order to determine the aggregate model parameters the following inputs are required:

case 1) Derivation from statistics of counts: it requires the number of packet arrivals as a function of time.

case 2) Derivation using queueing statistics: it requires the estimation of the queue length statistics and an asymptotic of the autocorrelation function. More over the mean inter-arrival time is required.

case 3) Derivation using EM algorithm: it requires the sequence of time arrivals with the record of packet sizes. If the number of packet sizes does not correspond to the number of packets categories that are required, a mapping is necessary and then, an histogram of packet size repartition need to be collected.

Many methods have been used because it was impossible to use an unique method, for each of them some cases occurred where the limits were reached: the first method is using just the arrival process but is generally unprecise, the second method was proposed as outperforming previous method, but in case of low traffic the statistics on the queue size don't allow to obtain sufficient statistics for the asymptotic to make sense. The EM algorithm can be used to obtain good estimates. Nevertheless it requires a high precision and therefore the speed of convergence is low. Convergence speed can be increased by a good initial estimate located not too far from the final solution. A first proposition for this initialization was following a procedure introduced by Ryden but in some of the cases, inconsistent results were obtained. The problem is due to the fact that all arrivals are classified for the same state, that render inconsistent the determination of the MMPP parameters. On the other hand, the hypothesis of single state arrivals (Poisson) was rejected by many statistical tests. To overcome these difficulties, a second initialization procedure was implemented, which is tailored for MMPP(2) models. The initial values obtained here are in general different from previous initialization procedure, but they also lead to convergence at another solution. Indeed the parameters of an MMPP(2) are not unique.

The methods for the determination of the parameters are : 1) for the MMPP, the first method is based on the moments estimates of the arrival process , the second on the asymptotic estimates of the queueing statistics, the third is based on the EM algorithm. 2) for the BMAP first the packet sizes are chosen from the histogram of the packet repartition and then the packet arrival process is chosen taking into account the packet size at arrival using the EM algorithm.

This derivation was performed for different number of voice sources aggregated together. The question is how to retrieve the number of voices sources from an initial model. In a previous study [GRB05], an MMPP(2) (with 4 parameters $\lambda_1, \lambda_2, r_1, r_2$) has been derived.

The parameters in table 5.1 were derived using the method of Kang. This method was working correctly for the load described. When the traffic load was low (like for example for 30 and 15 voices sources), the determination of the model parameters was not possible (with a monitoring queue working at a capacity of about 132 voice sources).

From the simulation of 100 VoIP sources, an MMPP(2) equivalent was searched. Using the EM algorithm, the following equivalent was found.

$$Q = \begin{pmatrix} -5.8e-4 & 5.8e-4 \\ 4.096 & -4.096 \end{pmatrix} \quad L = \begin{pmatrix} 3644.09 & 0 \\ 0 & 1017.62 \end{pmatrix}$$

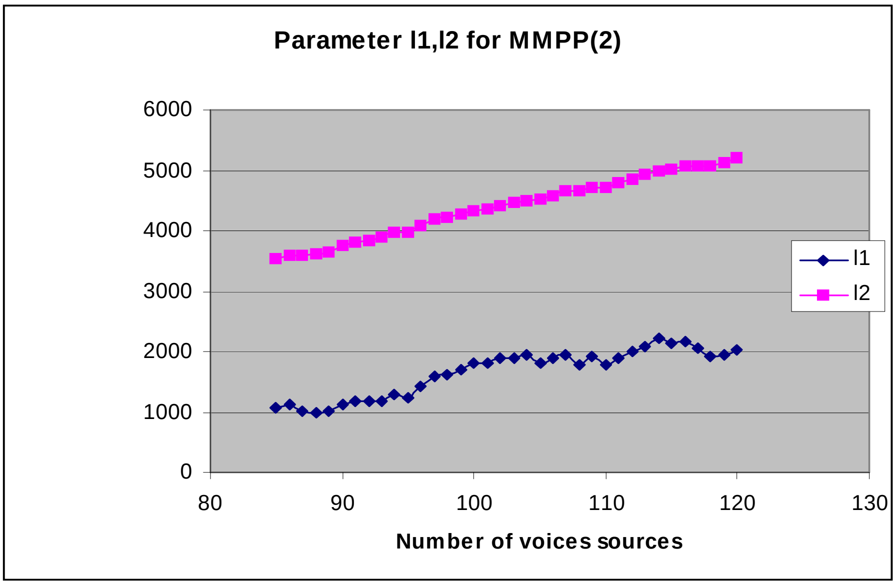


Figure 5.6.: Parameters λ_1, λ_2 for MMPP(2) model for Nvoip

N	$E[N_1]$	$E[N_1^2]$
100 (VoIP)	192.4255	15.2167
100 (MMPP(2))	193.3044	7.5181

Table 5.2.: Parameters of Gaussian models for one node (with bandwidth restriction)

For such a process, the mean arrival rate λ_1 is 3643.7 packets per second. Since the voice model is considering packets of 100 bytes ($\mathcal{P}_1$ packets), and the corresponding $\mathcal{P}_2$ packet has a size of 848 bits, that fits completely in two packets $\mathcal{P}_3$. The packet rate at node level is two times the level 1 rate λ_3 is 7287.4 packets per second when $\mathcal{P}_3$ of size 60 bytes are considered.

The methods for the derivation of equivalent parameters that were investigated have been recalled earlier. Not every method could be used in every case. For the case of voice only, a increasing number of voice sources ranging from 85 to 120 was used to derive equivalent model. Figure 5.6 shows the obtained parameters of a MMPP model based on recordings for an increasing number of users. The parameters were obtained using the EM algorithm. The initialization procedure was the one described in chapter 3. It must be noted that the values obtained during the initialization step were for some recordings wrongs (the sign of the elements of the Q matrix was the contrary as expected), other procedures were examined, but they were giving estimates with different order of magnitude. Moreover the hypothesis of exponential arrivals tested with the different statistical tests was refused. Anyway, the EM algorithm was able to reach meaningful convergence, with coherent estimates in stead of the bad initial guess. The figures show a ”linear” behavior as was obtained using the KANG method for less cases.

5.2.3. Transport level

Since a single node was considered, the results obtained at the transport level are similar to the results obtained at the node level.

First, in table 5.2 the parameters of a gaussian estimates are shown, that was derived as the equivalent of 100 instances of the VoIP model and for one instance of the MMPP(2) (using the parameters from table 5.1). These two gaussian models are transport level equivalents of the original and aggregated equivalent having the maximal simplicity. A slight difference can be seen for between the mean of the first model and the second (that will imply an higher rate for the second model) and less variance for the

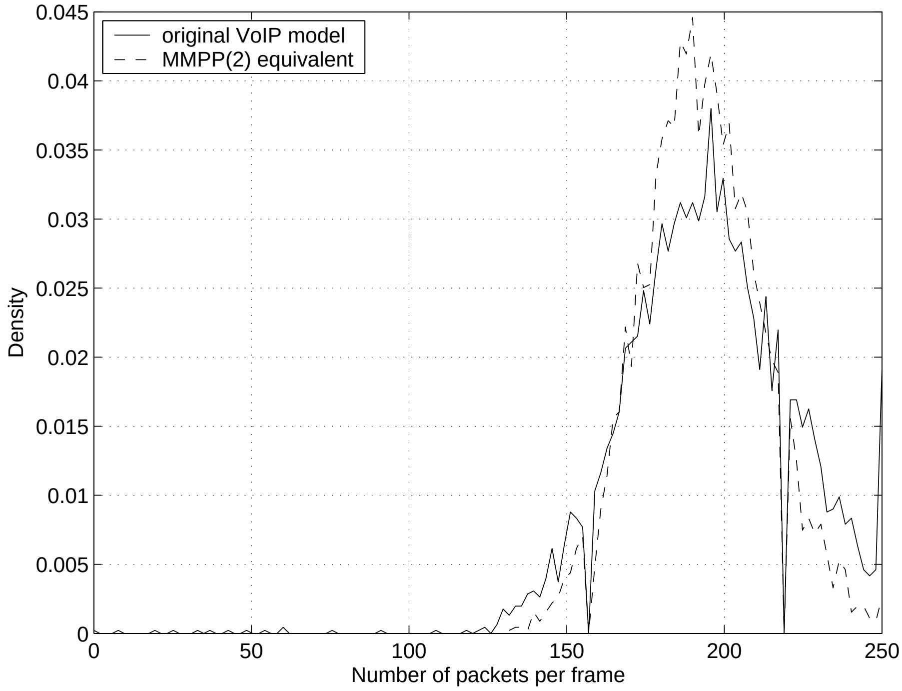


Figure 5.7.: Comparison of the distribution of the number of cells viewed at transport level

MMPP(2) model as in the original case. This difference is due to the characteristic of the MMPP, which have an determined index of dispersion (and is ”contained” in the original traffic recording.)

To complete this analysis, figure 5.7 shows a comparison of the packet composition of the frame (distribution of N_1) for the original and aggregate equivalent.

5.2.4. Summary of the constructed case

To summarize the conducted actions, in this numerical study, the VoIP model has been implemented and different equivalencies were investigated. It can be used to simulate a full size system (for example with 50 nodes and 100 subscribers). Aggregate models have been derived at node level: the MMPP(2) revealed to be able to capture the characteristic of the trace.

As a complement, light-weight models were investigated. As such gaussian equivalents were derived as equivalent for a node. These constructions of full-size analysis will be presented in the next two chapter using also other traffic sources.

5.3. Justification of equivalencies

Since the generated streams uses two distinct generators, it is impossible to have exactly the same arrival time and packet size (if applicable), so the reference model and the aggregate equivalent will always have different realization in $(\mathcal{I}, \mathcal{S})$. Anyway the analogy in the generated sequence can be monitored.

5.3.1. Node level

5.3.1.1. Traffic volume analysis in the pyramid

The first investigated metric is the general shape of the data volume of two recorded traffic stream (for example the amount of bit generated vs time). Since every traffic generator is based on a random generator that have their particular realization, equivalent models can never reach a perfect match.

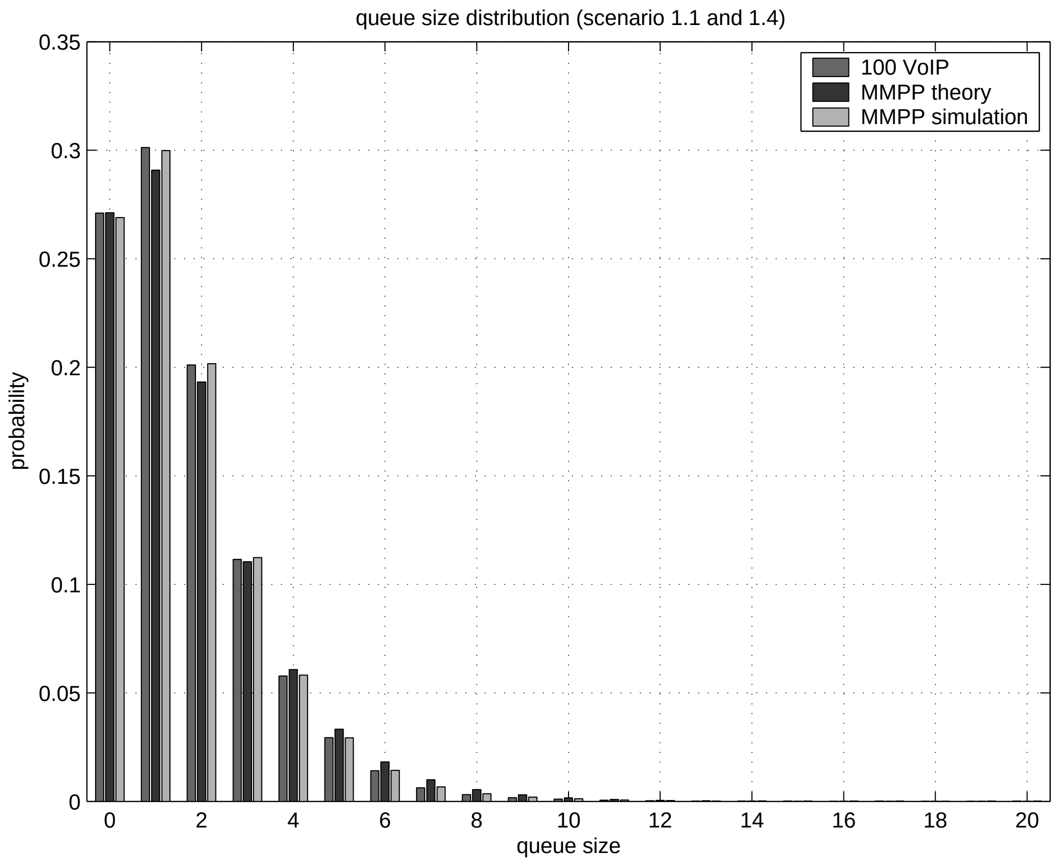


Figure 5.8.: Queue size distribution for 100 voices sources

By comparing the trace of aggregation of on/off traffic and of equivalent MMPP(2) model, the agreement is good, even if the ON/OFF model has a transition period, that is not present in the other model. The transition period is due to the implementation of the model, that initiates the state either in ON or OFF state, but with the duration of a normal ON or OFF period and not of a remaining ON or OFF period time. To prevent this faulty behavior, the model is implemented with no transmission during the first ON period state, given that the first state is an ON state. For this reason, there is a clearly marked transition period at the beginning of the voice model.

5.3.1.2. Queueing characteristics

A second investigated characteristic is the queue length for a queue receiving the traffic stream as input. If the queue is stable (corresponding to an incoming stream volume inferior to the maximum output capacity of the queue), then a stationary length is reached. Theoretical developments determine this queue length distribution for different hypothesis on the arrival process, the service time and the queue parameters. In the simulator, a queue is implemented with either a fixed service time or a service time proportional to the packet length. The service time (per packet or per bit) has to be set according with the input stream arrival rate (per packet or per bit). The stability of the queue has to be checked for example by computation of the load factor ρ.

Figure 5.8 is showing a comparison of the queue length at departure from packet from a monitoring queue located a the node level. Three cases are compared: the first one corresponds to a precise simulation of 100 voice sources (the queue is using the input stream $\Phi_{1.1}^{(3)}$), the second ones corresponds to the theoretical distribution for a MMPP with the parameters derived from the asymptotic parameters of the queuing statistics and the third one from the implementation of the MMPP model (which corresponds to

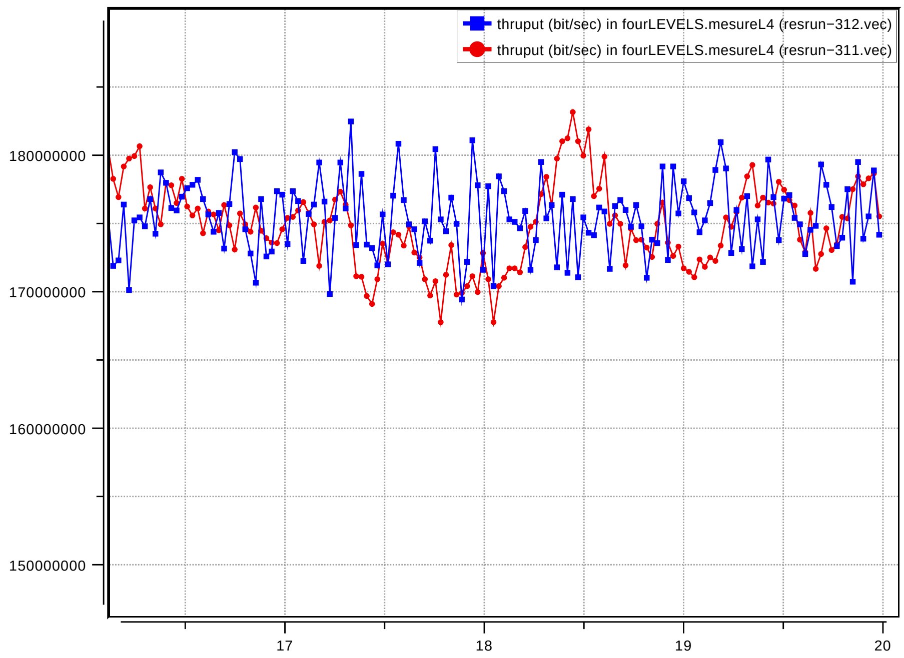


Figure 5.9.: Throughput vs time

the input stream $\Phi_{1.4}^{(3)}$).

The packet service time in the monitoring queue was chosen to $\bar{x} = 2e^{-4}s$, the generation rate is $\lambda = 3643.7 \simeq 100 * \frac{\bar{N}}{T_{on}*AF} = 100 * \frac{35}{0.35*2.8} =$, so $\rho = 0.7287$. The monitoring queue is working in packet mode. This would give a probability of non queueing $\Pr(Q = 0) = 1 - \rho = 0.271 = \mathbf{x}_0$. This value corresponds to the first column in the queue size distribution. The theory of MMPP/Q/1 queue is providing $\mathbf{x}_1$, the probability of having a queue length of 1. Once $\mathbf{x}_0$ and $\mathbf{x}_1$ have been derived, the complete distribution $\mathbf{x}_i$ can be obtained after computation of the matrix G and all derived values. In the case of a constant service time, the iterative solution of $G = \int exp(Gx)dH(x)$ is obtained more easily.

5.3.2. Transport level

5.3.2.1. Analysis of volume

Figure 5.9 is representing the throughput measured at the transport level. The curve in blue corresponds to 50 x MMPP(2) model and the curve in red in 50 x {100 VoIP}. It shows the visual equivalency of both generated sequence. The parameters derived for a gaussian model show that the variance of the MMPP(2) is smaller but that the mean value corresponds.

5.3.2.2. Analysis of delay

Figure 5.10 shows a comparison of the probability density function of the end to end delay for the one voice application in the precise simulation as in scenario 1.1 and with an aggregate model as in scenario 1.3. The delay is uniformly distributed in the frame duration, because the allocated bandwidth is high, so no significant delay is shown. The figure shows for clarity the minimum and the maximum of both density, each density is in the darker area (drawing both histogram results in a picture that is difficult to interpret, the graph does not mean that both. The complementary density function shows that no major difference between both distribution can be detected. With the considered allocated bandwidth (275 cells /frame), the agreement between both model is good.

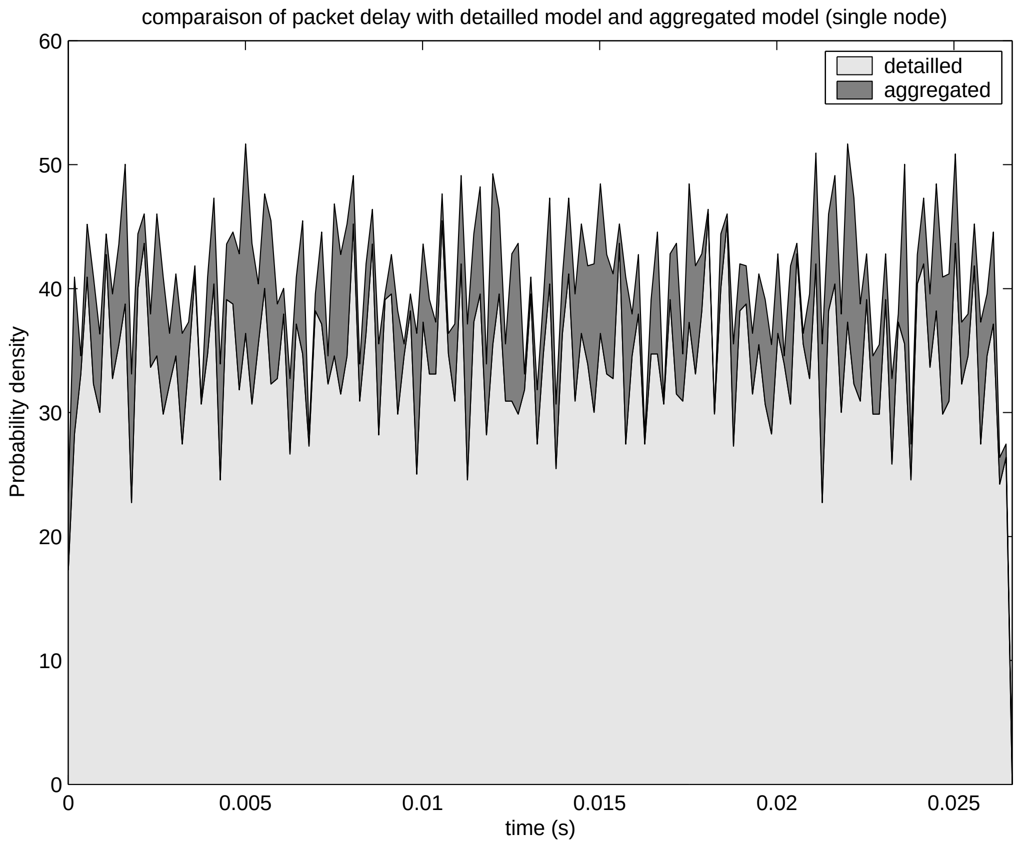


Figure 5.10.: Comparison of the pdf of the delay for the scenarios 1.1 and 1.3

5.4. Summary

In this chapter, the focus was given to a simple case where the different level and the form of equivalencies can be viewed. The focus here was given a case where only VoIP sources were used. The original application model was compared with aggregate (MMPP(2) models and synthetic gaussian models. These generators were compared from the point of view of the volume, but also using supplementary metrics related to performance (delay, queue size distribution). Good equivalencies were obtained.

6 Equivalent nodes for different traffic conditions

With the systems under study, the concept of equivalent traffic model at mode level is of interest as a large number of subscribers per node are considered with a limited number of application per subscriber.

At node level, the influence of the operators modifying the traffic stream prior to the transport level is quite strong. Since the transcription into a format compliant with the satellite transmission occur within this level, most of the modifications of the traffic stream will occur within this level. In addition to the effects of the operators on the traffic stream, the equivalent model are also investigated: for example in the aggregation of web and voices, the use of a BMAP(3) with 3 or 4 packets size is studied.

6.1. Overview of presented results

6.1.1. Description of investigated case

The aggregation of models from different sources are investigated. To determine an equivalent aggregate model for a stream $\Phi = \{\mathcal{S}, \mathcal{I}\}$ a knowledge of the sequence of the aggregated size $\mathcal{S}_i$ and inter-arrival time $\mathcal{I}_i$ is necessary. Because of the heterogeneity, the packet size will need to be modelled. The method is then first to determine the distribution of the packet size and then to derive a model for the packet arrival time taking into account the size of the packet at the instant of arrival.

Figure 6.1 shows the scenario considered for heterogeneous applications. This scenario is consisting of a single node with 100 subscribers using a VoIP application and a BMAP web request model. Using the record of packet arrivals and packet size at node, a BMAP(3) will be derived that has a traffic stream equivalent to a node with this heterogeneous traffic. The scenario indicated 3.1 corresponds to 100 subscribers with the VoIP application and the BMAP on/off for a single user that will be derived later on. The scenario 3.2 corresponds to 100 subscribers with the VoIP application and a BMAP(3) model representing all web browsing application. The scenario 3.3 corresponds to an aggregated model that is equivalent to the previous two. The derivation of this model will be explained hereafter and the comparison with the original will be also investigated to justify the usage of aggregate models.

6.1.2. Data collected at node level

During the simulations, following results were gathered:

- Queueing statistics collected at a queue located at node level.

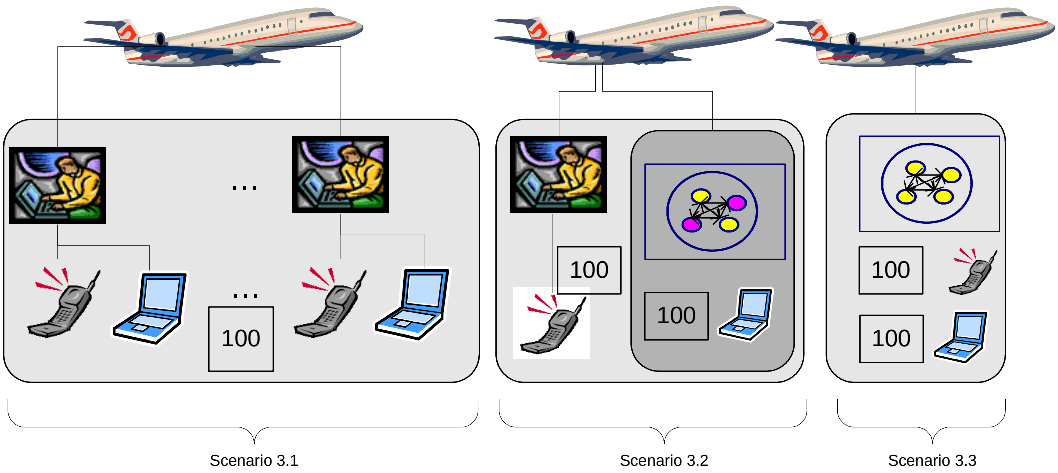


Figure 6.1.: Description of the second set of scenarios considered

- Analysis of volume of data generated vs time. (using records of the instants of packet generation, the packet size).

This data is derived from the stream $\Phi^{(3)}$, measured at the "output" of the node, before its transmission to the last transport level. Some of the results are more meaningful when collected at the transport level, the data are then derived from the stream $\Phi^{(4)}$.

- Measured throughput at transport level

Results at collected at the node level can not be directly compared with the results collected at the transport level.

6.2. Additional traffic sources considered

In order to derive more realistic traffic sources, additional models are required modelling additional applications.

6.2.1. Addition of video streaming traffic model

In a former study, video streaming traffic models were implemented and the equivalent model was derived using again the method from Kang.

Table 6.1 presents the results of a MMPP(2) model that was fitted to a trace composed of 10 and 12 video users or, in an heterogeneous case, with 30 VoIP users and 10 video users or 15 VoIP users and 12 video users.

The model is not taking the packet size into account because both models were generating common size packets. In the video model, the frame size was used to compute the numbers of packets to be send per frame and the duration between packets within the frame was modified accordingly to accommodate this number of packets.

In the following the addition of video traffic will not be considered any more because the effects implied by the video traffic (self similarity etc.) are not investigated. Their addition will not bring too many benefits.

Video Sources				
10	1.074758	0.74758	1.69e-3	7.851e-6
12	1.126232	0.82520	1.9868e-3	1.7872e-4
Voice and video				
30 + 10	1.1836	0.8761	1.4911e-3	3.3174 e-4
15 + 12	1.35821	0.884	1.9161e-3	3.778487e-4

Table 6.1.: Normalised parameters of the MMPP(2) for different mixture of voice and video sources

6.2.2. Addition of a web surfing traffic model

6.2.2.1. Backbone web traffic model model

Model In order to model the backbone web traffic, a BMAP model can be used, here it is necessary to include packets of different size to respect the characteristics of this kind of traffic. These models were presented in 3.1.2.1. Following the approach in [KLL03] where a BMAP 3×3 with three distinct sizes was used. The sizes of the packets are modelled by three categories:

Category	< 500 bytes	$> 500, < 1000$ bytes	> 1000 bytes
Mean size	94 bytes	575 bytes	1469 bytes

.

The generation process depends of the following matrices:

$$D_0 = \begin{pmatrix} -1578.53 & 0 & 289.79 \\ 1168.44 & -5215.23 & 1903.48 \\ 0.81 & 11.52 & -55.61 \end{pmatrix} D_1 = \begin{pmatrix} 41.08 & 0 & 84.04 \\ 0 & 2088.27 & 55.04 \\ 0 & 6.71 & 25.12 \end{pmatrix}$$

$$D_2 = \begin{pmatrix} 664.06 & 261.28 & 198.50 \\ 0 & 0 & 0 \\ 8.80 & 0.98 & 0.70 \end{pmatrix} D_3 = \begin{pmatrix} 13.13 & 16.23 & 10.42 \\ 0 & 0 & 0 \\ 0.48 & 0.41 & 0.08 \end{pmatrix}$$

The corresponding generator was implemented in the simulator. It corresponds to an overall arrival rate of $\lambda = 27746$ (bytes/s) using (A.41).

In order to test the generated stream $\Phi_1^{(1)}$ that consists of the packet generation time and the corresponding packet size, the queue size of a queueing system which input is the generated stream is monitored at periodical instants. The queue serves packets with a service time proportional to their size.

Queueing properties Figure 6.2 shows the arbitrary time queue size distribution for the BMAP/D/1 queue ($\Pr[Q^a = k]$) for a low load $\rho = 0.4$. It is compared the theoretical queue distribution (of the BMAP/G/1 queue) with the queueing statistics collected in the simulator. Moreover, the asymptotics for the queue length have been drawn computed for the BMAP/G/1 queue. The service time is constant for 1 byte and is $\overline{x} = \frac{\rho}{\lambda_{BMAP}} = \frac{0.4}{27746} = 1.4410^{-5}s$. The distribution has a peak for $\Pr[Q^a = 0] = 1 - \rho$ which is not represented, but is present on simulations results as on theoretic distribution. In the represented domain, an excellent match between theoretical distribution and simulated value is obtained. The influence of the different packets size on the queue can be seen by the two peaks at 94 and 575 which are the small and medium packets. The peak at 94 bytes correspond to the arrival in an empty system of a small packet. The decrease to the origin is mainly due to the processing of this small packet.

Packet size Figure 6.3 shows the distribution of the web packets sizes at the output of the generator.

The share of each packets in volume is $\frac{S_i(\pi * D_i * \mathbf{e})}{\lambda_{\text{BMAP}}}$. Using the preceding values we have

$$\pi = \begin{pmatrix} 0.0223 & 0.0081 & 0.9696 \end{pmatrix} \quad D = \begin{pmatrix} -860.26 & 277.51 & 582.75 \\ 1168.4 & -3126.7 & 1958.5 \\ 10.09 & 19.62 & -29.71 \end{pmatrix}$$

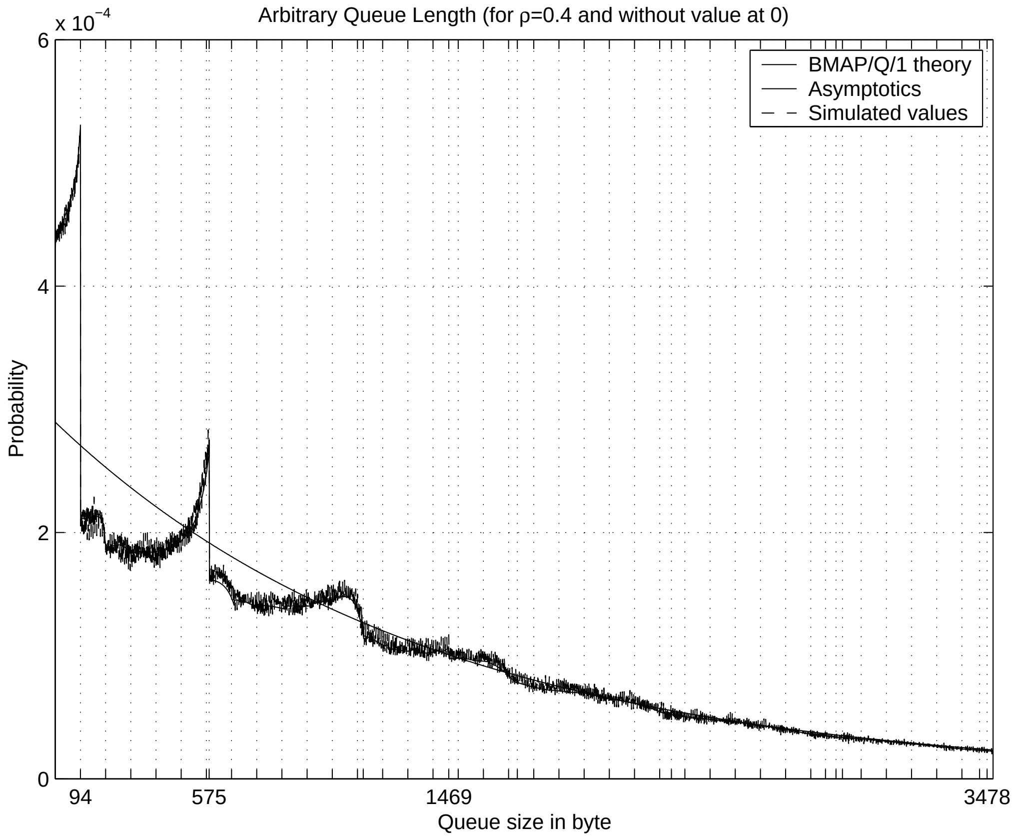


Figure 6.2.: Queuing property of the generated stream for a single web user

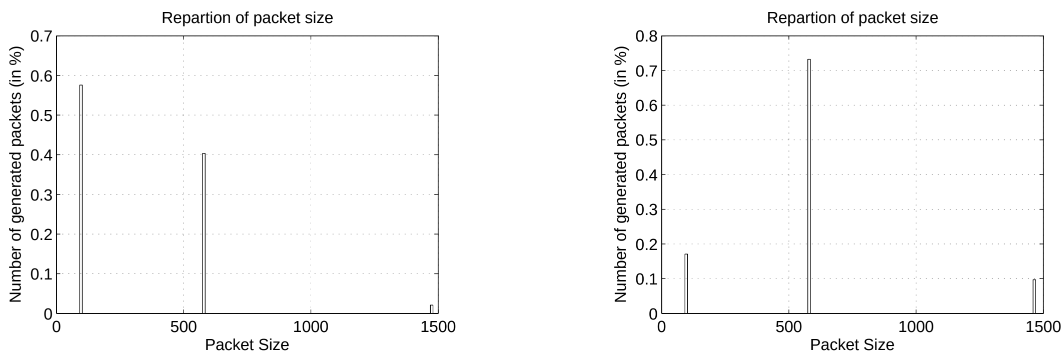


Figure 6.3.: Repartition of generated packets sizes for the web model (in size and volume)

and hence

$$p_V = \begin{pmatrix} 0.1726 & 0.7306 & 0.0968 \end{pmatrix} \;\; p_N = \begin{pmatrix} 0.5787 & 0.4005 & 0.0208 \end{pmatrix} \tag{6.1}$$

This corresponds to the values in the figure.

6.2.2.2. Individual web model

The BMAP model was derived from measurement on a backbone. Obtaining a model for a single terminal requires some more efforts. The BMAP model can be modulated by an ON/OFF process that is reducing the activity of the single source. Like for the voice model, two exponential distributions can be used for the modelling of the on and off duration. For the BMAP model used during the on period, the mean rate of the process has to be adapted to the packet arrival rate expected for

Another reason in favor of the usage of a modulated BMAP model is the capability to get indication

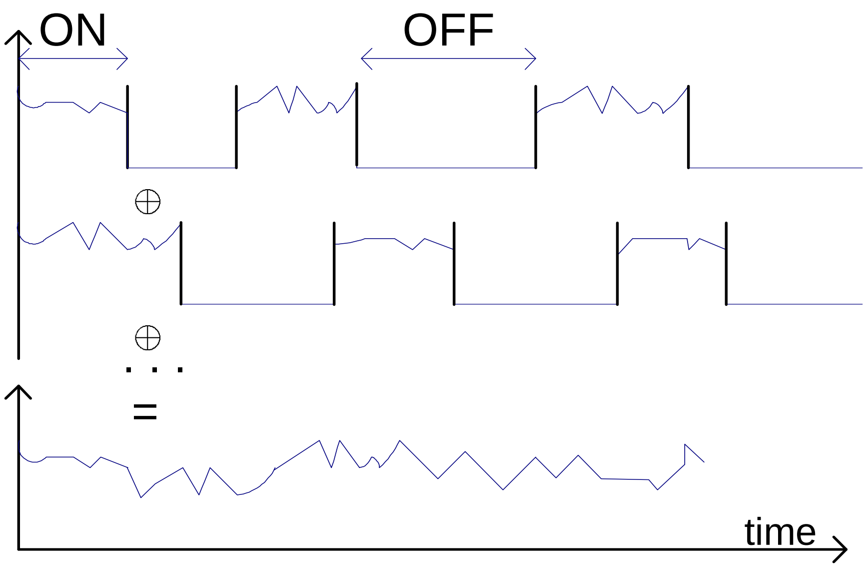


Figure 6.4.: Aggregation of an On/Off model

on the session parameters. Indeed the BMAP model is just describing the packet transmitted over time, but no information about the evolution of the session duration can retrieved. Cells emitted during the different activity period are marked with a session identifier. At cell reception, the cells identifier enable the estimation of the duration of the session (by evaluating the duration between the first packet reception and the last packet reception). Such a metric is from far not a precise estimate of the session duration (because no modelling of the protocol implied in web surfing service is performed), but it is sufficient to give a rough idea of the session duration.

6.2.2.3. Aggregation of single web models

The model for web traffic was derived from measurements on a backbone. For this reason, the model is already an aggregated model, that can generate a traffic stream $\Phi^{(2)}$. It is hence just needed to check if the session-based model agrees with the aggregated model.

Figure 6.4 shows the aggregation of ON/OFF modulated model into a global model. If the single flow has a mean data rate of m in the ON period, then the aggregation of N sources will have a mean data rate $M = Nm\frac{T_{on}}{T_{on}+T_{off}}$. If $T_{off} = 0$ (no pause in the model), it comes $m = M/N$ that shows that just an adaption of the source rate is required. This way if also guarantees that the aggregation has N active sources. If $T_{off} \gg T_{on}$, the ratio is denoted $f = T_{on}/T_{off}$ and is the probability that a modulated single source is active. The overall data rate is then $M = Nmf$. If $T_{on} = 10\text{min}$ and $T_{off} = 50\text{m}$ we have $f = 0.167$. For $N = 100$, it means that $M = 16.67m$, the rate of the single BMAP modulated model must be reduced accordingly. Concerning the number of active sources in this case, it follows a distribution with parameters (N, f) $(\Pr(i) = C_N^i f^{N-i}(1-f)^i)$ which has a maximum for $i_m = \lfloor (N+1)f \rfloor$. It then gives $i_m = 16$. The value (not equal to the mean value) is anyway close to the ratio $\frac{M}{m}$ of 16.67.

For the packet size distribution, the same repartition as for the non-modulated BMAP model is used. As a result in the aggregation no modifications shall appear.

A comparison of the volume and packet repartition for the last two models is presented in section 6.5.1.

6.3. Equivalent model for heterogeneous traffic

The results here were obtained after simulations corresponding to the scenario 3.2. 100 VoIP sources were considered together with the backbone model. From the recording collected, the sequence $(\mathcal{I}, \mathcal{S})$ for the aggregated stream was collected. A BMAP model was searched that would be able to characterize both web and VoIP.

The first step is to classify the packet size in order to classify each packet of size $\mathcal{S}$ to one of the category. The results of the packet repartition in the 3 domains (0-500 bytes / 500-1000 bytes / 1000-1500 bytes) gives the following mapping for the packet size:

Category	< 500 bytes	$> 500, < 1000$ bytes	> 1000 bytes
Mean size	108 bytes	582 bytes	1473 bytes

.

With this information, the classification into the size category (1),(2) or (3) is possible.

Using the EM algorithm, using as input the recording of inter-arrival time and of packet size category, the convergence was obtained after consideration of all samples, based on an initial value obtained after consider of the first 5% of the samples. The obtained parameters of this model for heterogeneous traffic is then

$$D_0' = \begin{pmatrix} -3904.5 & 0.0111 & 0 \\ 2514.25 & -6337.24 & 0 \\ 0 & 7016.5 & -7049.51 \end{pmatrix} D_1' = \begin{pmatrix} 3373.02 & 0.1128 & 529.472 \\ 0.0103 & 2821.84 & 842.896 \\ 0.0029 & 32.4798 & 0.5259 \end{pmatrix}$$

$$D_2' = \begin{pmatrix} 0.0001 & 0.0003 & 0.0001 \\ 0.0001 & 157.714 & 0.5304 \\ 0 & 0 & 0 \end{pmatrix} D_3' = \begin{pmatrix} 1.878 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

This model has an overall arrival rate of $\lambda = 402282$ (bytes/s) using (A.41). (It is a rate measured at level 3).

Using the preceding values we have

$$\pi' = \begin{pmatrix} 0.762802 & 0.160676 & 0.076522 \end{pmatrix} \ D' = \begin{pmatrix} -529.601900 & 0.124200 & 529.472100 \\ 2514.260400 & -3357.686000 & 843.426400 \\ 0.002900 & 7048.979800 & -7048.984100 \end{pmatrix}$$

and hence

$$p_V = \begin{pmatrix} 0.9580 & 0.0368 & 0.0052 \end{pmatrix} \ p_N = \begin{pmatrix} 0.9925 & 0.0071 & 0.0004 \end{pmatrix} \tag{6.2}$$

An on-off version of this model could be derived in order to have an heterogeneous model at the subscriber level. Because of the different activity factor of the web and VoIP application with different ON distribution (and duration), such model would have not much real sense. This model will hence be mainly used as an equivalent at the node level.

So far, a aggregate model for the heterogeneous traffic was derived. This is the model corresponding to the scenario 3.3 in figure 6.1. The properties of the trace generated with this model will be investigated later on (in section 6.5.2)

6.4. Gaussian equivalents for node traffic

In this section, easy to use models are proposed at the level 4.

6.4.1. Derivation for reference case

In table 6.2 the value of the parameters for a Gaussian bi-variate aggregate model for N subscribers are summarized. These values will be used as a generic Gaussian model for aircraft with variable number

N	$E[N_1]$	$E[N_2]$	$E[N_1^2]$	$E[N_1N_2]$	$E[N_2N_1]$	$E[N_2^2]$
100	192.3686	13.9524	677.1368	-36.5608	-36.5608	278.7556
101	193.6049	13.9731	688.0797	-32.467	-32.467	283.8626
102	195.758	14.6172	673.5927	-42.3139	-42.3139	291.2456
103	197.126	14.4281	687.4767	-48.952	-48.952	287.4916
104	199.5021	15.2741	716.9346	-73.1791	-73.1791	298.2165
105	201.2455	14.2794	747.6645	-81.2098	-81.2098	284.6159
106	203.7906	15.2553	709.6874	-81.5548	-81.5548	285.4518
107	204.7979	14.5328	698.8019	-81.8108	-81.8108	278.0913
108	205.3833	14.7514	688.3994	-76.1597	-76.1597	282.6594
109	209.4999	15.7619	754.5811	-139.2813	-139.2813	318.1602
110	209.4085	15.7545	714.3991	-128.8929	-128.8929	298.5579
111	214.0236	15.3711	711.718	-178.9732	-178.9732	316.288
112	215.6324	15.3525	663.2123	-161.6952	-161.6952	305.4141
113	216.7995	15.3654	694.4643	-195.3057	-195.3057	321.5179
114	218.4192	16.5474	655.0058	-231.5391	-231.5391	356.8913
115	218.9856	16.8068	687.7415	-260.5305	-260.5305	374.7777
116	221.3499	16.4693	658.7701	-255.6933	-255.6933	345.4086
117	221.4676	16.9596	674.5477	-299.1251	-299.1251	389.8328
118	225.1854	16.2877	635.9014	-323.9917	-323.9917	388.5999
119	228.4669	16.4404	620.928	-403.2007	-403.2007	439.8298
120	230.0413	16.7187	590.0656	-432.3814	-432.3814	457.1948
121	231.4236	17.3228	571.1954	-492.2139	-492.2139	502.1025
122	233.8018	15.2491	508.0805	-424.7134	-424.7134	428.1476
123	236.0702	13.1045	488.883	-420.2514	-420.2514	420.1586
124	237.0521	12.4173	438.611	-378.4547	-378.4547	378.4163
125	239.8898	9.6096	383.3581	-329.3187	-329.3187	325.7864
126	242.2732	7.2559	310.4893	-251.7943	-251.7943	249.0017
127	243.3529	6.1746	302.4054	-240.8361	-240.8361	233.71
128	242.7198	6.6351	295.272	-233.1219	-233.1219	228.9654
129	246.4786	3.0037	177.728	-119.1915	-119.1915	111.2147
130	248.4163	1.1823	102.4532	-49.3176	-49.3176	43.3222

Table 6.2.: Parameters of Gaussian models for one node (with bandwidth restriction)

of passengers in a study of complete satellite system dimensioning in the next chapter. These values are only valid for a bandwidth/node of 250 cells.

It must be noted, that the mean number of cells of the second type $\hat{N}_2$ experiences a decrease for increasing load. Hence, this synthetic load is not stressing so much the system as a linear model would do. But it is also difficult to study a system close to saturation, because the delay have a dramatic increase when the load is approaching the limit .

These parameters are sufficient to characterize the model to be used at the transport level because the mean and the autocovariance matrix are sufficient to characterize the process.

6.4.2. Influence of Transport level bandwidth

Similarly these values were computed for a reduced bandwidth (with BW=225) and for an increased bandwidth (with BW=275). Interesting results are gathered for the reduced bandwidth. The traffic generated with this model has then this limitation in volume that is occurring earlier. With less subscribers the saturation will be attained, than in the reference case.

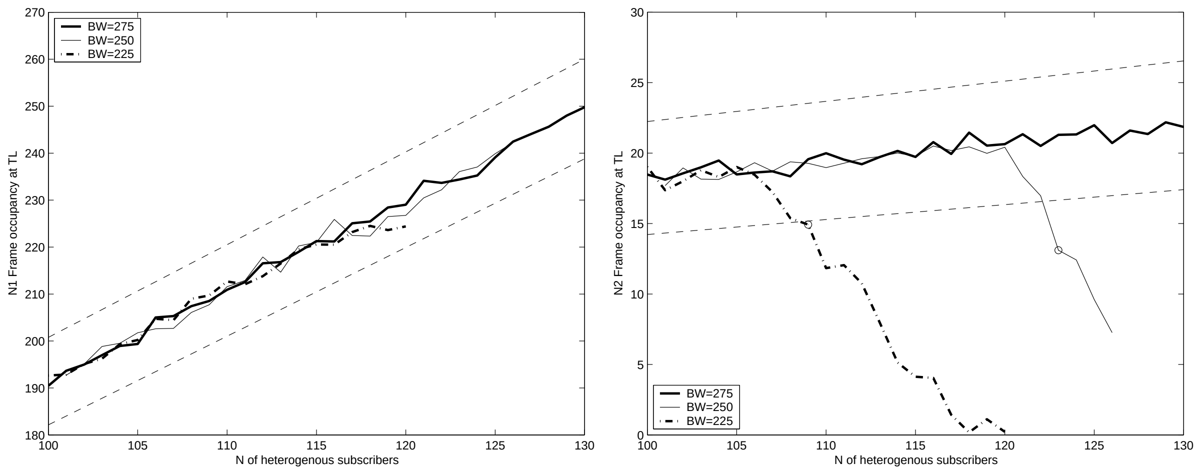


Figure 6.5.: Comparison of the number of cells transmitted (type 1 on the left, type 2 on the right)

In order to measure the differences between each case, a linear model is derived for $\overline{N_1}$ and for $\overline{N_2}$ for varying passengers number. The linear model was derived from the values collected with an increased bandwidth limitation (BW=275), it came $\overline{N_1}(pax) = 1.932 * pax - 1.7755$, with the 95% confidence interval of these two parameters ([1.889 1.975] for the slope, and [-6.745,3.236] for the ordinate), and for $\overline{N_2}(pax) = 0.1247 * pax + 5.756$, with the 95% confidence interval of these two parameters ([0.1061 0.1434] for the slope, and [3.608,7.904] for the ordinate). This range is indicated in figure 6.5 where the measured value are represented. It can be seen that for N_1, the obtained result respect the linear model (except to the reduced bandwidth if the number of customer is further increased). On the contrary, for N_2 the behavior is not linear when the traffic is too important. For a bandwidth limitation of 250 and 225 cells per s, the point outside this level is determinated. It comes, for the first point outside the confidence interval

- $BW = 250$, $N = 123$
- $BW = 225$, $N = 109$

Because of the quality of service scheme, the second category begins first to suffer from the effects of the bandwidth reduction. Concerning N_1, the linear behavior is remarkable and occur up to the maximum bandwidth.

6.5. Traffic volume analysis

6.5.1. Web browsing case

In the simulator both models were implemented. In figure 6.6, a comparison volume of data between the BMAP (continuous model) and the aggregation of On/Off modulated BMAP is depicted. For the ON/OFF model, the modulating processes has following values: $T_{on} = 1s$, $T_{off} = 9s$, $AF = 0.1$. Two sets of value have been used: 1) For 10 users, the same matrices as in the continuous model have been used, 2) for 100 users the matrices ($D_0, D_1, ..., D_3$) have been divided by 10. They are both equivalent to the continuous model. The growth rate of both curve is similar because both curves have a similar slope. Following slopes were measured:

- case i) (web browser) $\hat{V}(t) = 2.7535e4 \times t$ 95% confidence: $[2.7533e4, 2.7537e4]$
- case ii) (10 times BMAP D/10) $\hat{V}(t) = 2.6225e4 \times t$ 95% confidence: $[2.6206e4, 2.6243e4]$
- case iii) (100 times BMAP D/100) $\hat{V}(t) = 2.7589e4 \times t$ 95% confidence: $[2.7582e4, 2.7597e4]$

This shows that the aggregation of the 100 individuals web browser has a very good agreement with the original BMAP model, because the generated traffic volume is equivalent (the difference of traffic volume rate is of less that 0.2%).The theoretical BMAP rate value was of 27746 bytes/s.

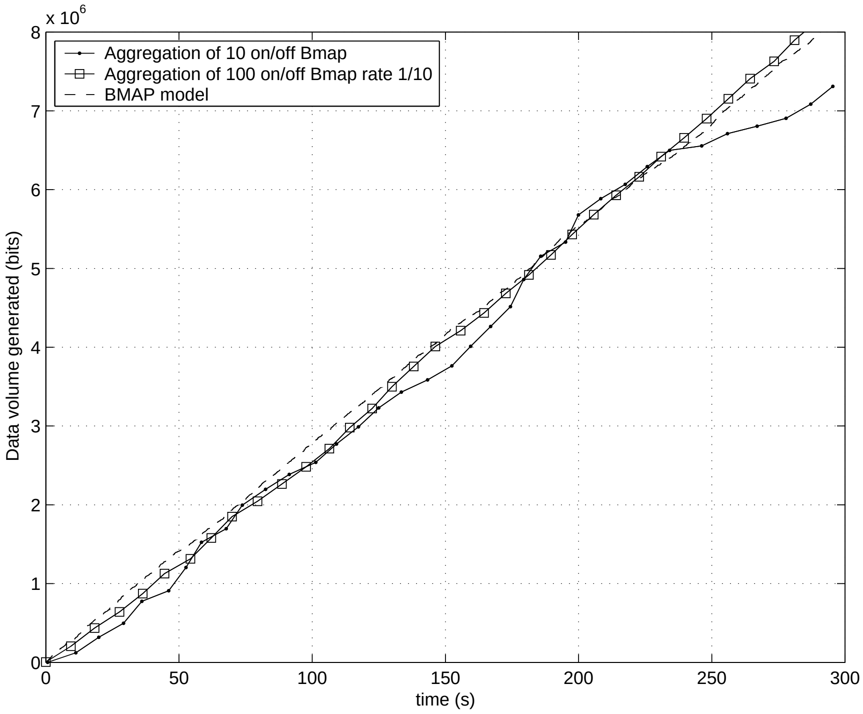


Figure 6.6.: Aggregation of an On/Off BMAP model and comparison with BMAP model - analysis of trafic volume

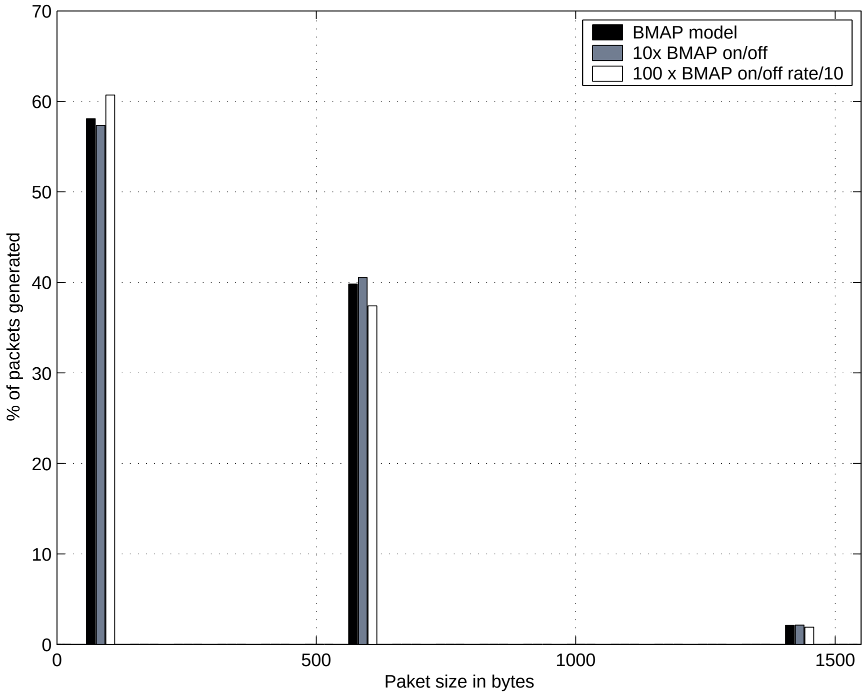


Figure 6.7.: Aggregation of an On/Off BMAP model and comparison with BMAP model - Analysis of packet size

In figure 6.7, a comparison of the size of the packets for the different size is included. A similar repartition is obtained. Recalling the results obtained in (6.1) for the repartition of number of packets theoretically presents using the parameters of the models, 57.87%, 40.05%, $2,08\%$ of packets should be of the first, second and third category. These percentages are similar as the one obtained in the figure 6.7.

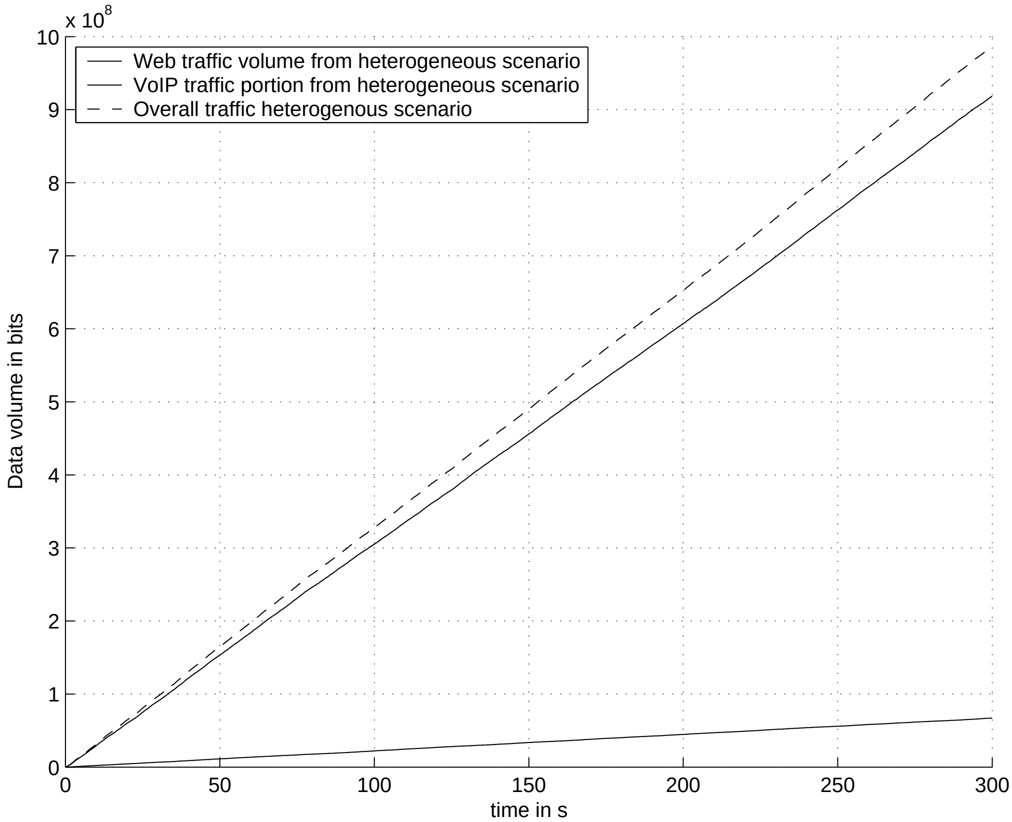


Figure 6.8.: Data volume generated vs time with the two traffic component(web and VoIP)

6.5.2. Heterogeneous case

Recalling the results obtained in (6.2) for the repartition of number of packets theoretically present in the system using this equivalent model.

For the heterogenous traffic, using the BMAP(3) model that was derived previously in the case of VoIP and web, a aggregate model at node level was obtained. On figure 6.8, the data volume generated by the aggregate model is compared to the data volume generated by the precise.

The volume rate was evaluated using a mean square procedure for a model of type $'a * x'$ (polynom starting at origin). We got the following estimates

- web only: $V(t) = 2.2362e5 \times t$, 95 % confidence interval $[2.2361e5, 2.2363e5]$,
- VoIP only: $V(t) = 3.048710^6 \times t$, 95 % confidence interval $[3.048710^6, 3.048710^6]$
- all (sum of the two): $V(t) = 3.276610^6 \times t$, 95 % confidence interval $[3.276510^6 3.276610^6]$

Moreover the repartition of the packet size is compared in figure 6.9.

$$p_V = \begin{pmatrix} 0.9580 & 0.0368 & 0.0052 \end{pmatrix} \; p_N = \begin{pmatrix} 0.9925 & 0.0071 & 0.0004 \end{pmatrix}$$

Figure 6.9 shows the repartition of packets in the system. The repartition of the smallest packets is of 99%. This corresponds to the model value of $99.25\%(p_N(1))$

6.5.3. Influence of Transport level settings

Here each subscriber is using two applications that are voice and web surfing. At the application level, two streams $\Phi_1^{(1)}$ and $\Phi_2^{(1)}$ are generated. Because the web surfing model is not so easily individualized, the model is generating traffic directly at the node level from a traffic stream $\Phi_{\text{www}}^{(3)}$

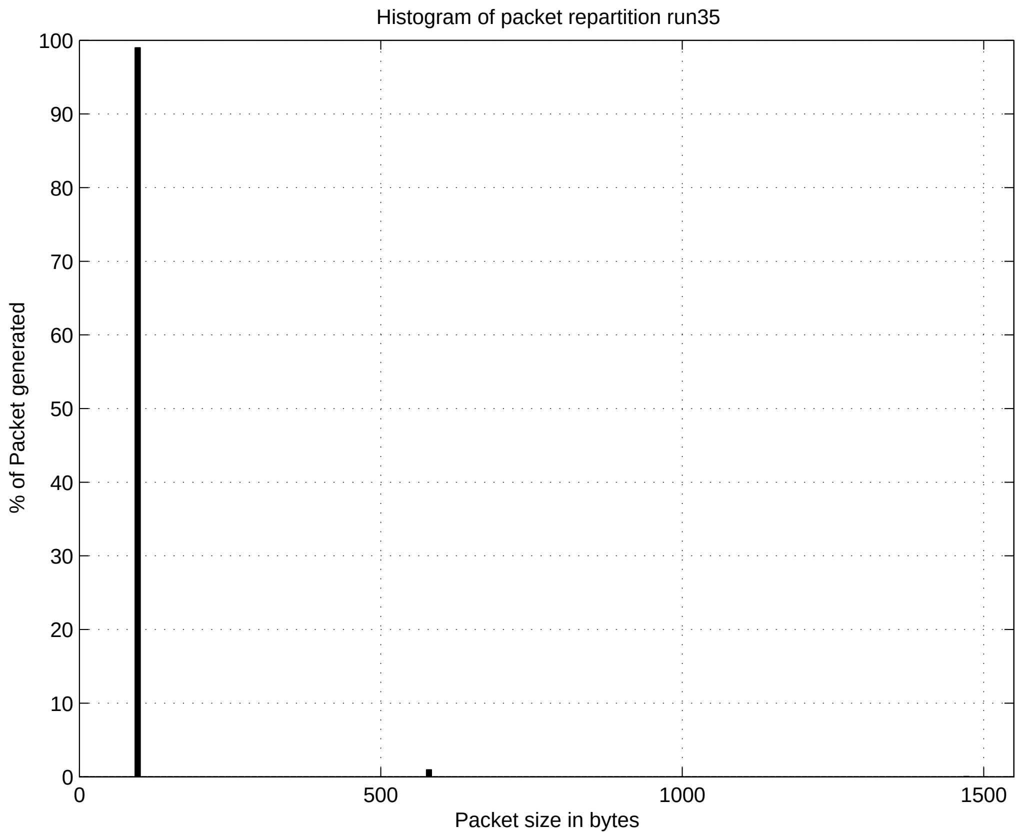


Figure 6.9.: Packet size repartition in the heterogeneous case

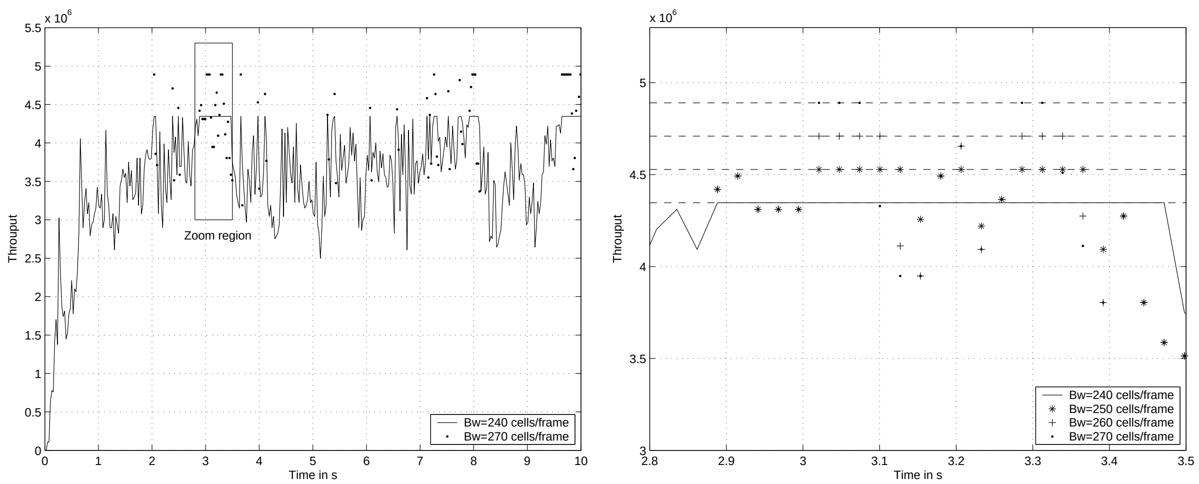


Figure 6.10.: Throuput on the satellite link for a single node

Figure 6.10 shows a comparison of the throuput measured on the transport level for a generated traffic stream composed of 100 subscribers with VoIP application and one web server. The effect of the bandwidth allocated per node can be seen: in the zoomed part 4 different bandwidth were allocated BW=240,250,260,270. When the generated traffic is over the allocated bandwidth, a plateau appears in the throuput. When the bandwidth increases, it can be seen that the occurrence of frame with the maximum number of packets is reducing. The practical consequence is that the delay is reduced with higher capacity.

6.6. Summary

Equivalent models at node level have been investigated in this chapter. For this purpose, heterogeneous traffic models were necessary. Tentative models were introduced and a user-oriented model was suggested in order to model web access transmission. The procedure of model derivation for this new traffic composition was repeated and an analysis of the properties of these model was conducted.

7

Full size system investigations

The results presented in this chapter concern the transport level of the aggregate pyramid. First, the scenario is presented where aggregation gains are investigated. Secondly, the results of node investigation from previous chapter are verified at transport level and a synthetic model at the transport level is proposed. Finally, investigations are performed to analyze the results obtained with aggregate models and the benefits that can be gained with their use.

7.1. Overview of presented results

7.1.1. Considered case

Figure 7.1 shows some of the nodes that can be considered as sources. The first case corresponds to a ”precise” node and is depicted on the left side where 100 passengers are considered with each two models of an application.

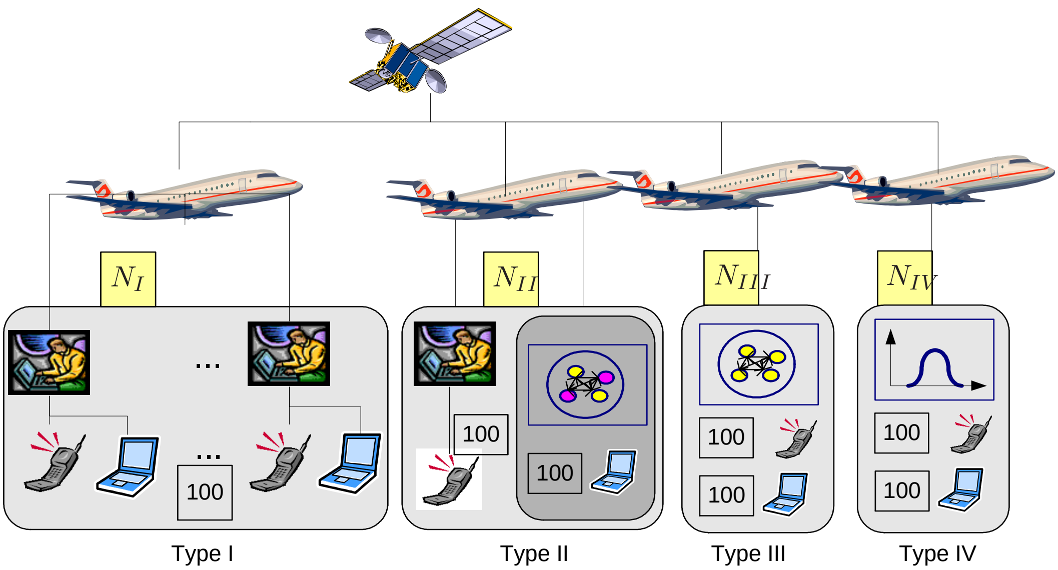


Figure 7.1.: Elements considered in the multiple mode case

The different type of node are then:

- type I (precise model). Each of the application is using its single model. The subscribers are homogeneous (each of them is using the same applications). The number of subscriber then characterize the volume of traffic generated.
- type II (realistic model). A synthetic model is used for the web traffic (with a specific traffic volume) and a certain number of VoIP users can be selected.
- type III (aggregate model). In this case the node is using the BMAP D' model that was described in last chapter.
- type IV (gaussian model). In this case the number of packet generated at level 3 is following a gaussian distribution. If two QoS classes are considered, the bi-variate gaussian model is used otherwise a single dimension gaussian model is used.

For the purpose of system evaluation other node profiles are introduced later on, based on a similar ”base” load to which a test application is added.

The scenarios are then composed by a combination of the different node type. It requires the description of the number of nodes of each type that are considered. The variables N_I N_{II}, N_{III}, N_{IV} characterize these numbers. Of course, other variations could have been performed: a fixed model per aircraft could have been used with a variable number of aircraft or the bandwidth of the satellite could have been reduced for a given traffic load. In the influence of the bandwidth will be investigated at a given load. For more bandwidth, the delay shall decrease. For less bandwidth, the shared bandwidth effect shall be attenuated, so the benefits in term of performance for web traffic shall be reduced.

7.1.2. Data collected at transport level

During the simulations, following results were gathered:

- Analysis of volume of data generated vs time. (using records of the instants of packet generation, the packet size).
- Density of transport level cells repartition (frame composition: record of number of N_1 and N_2 packet transmitted per frame)
- Performance indicators from the simulator. In the study performed, the session duration and the session volume in term of transmitted bytes during the ON period are calculated.

This data is derived from the stream $\Phi^{(4)}$, measured at the ”output” of the satellite. Since in the considered system the satellite is working with a constant clock ($T = 26.5$ms) and manipulates packets of constant size and is allowing a certain number of $\mathcal{P}_4$ cells to be transmitted following the described policy for quality of service, the stream can be described by the number of cells $\{N\}$ that was transmitted at $t = nT$. This number can be decomposed by cell categories, this constituting two series $\{N_1\}$ and $\{N_2\}$ such that $N = N_1 + N_2$.

7.2. Validation of node equivalents at transport level

In this paragraph, the results obtained for single node study performed in the previous chapter are confirmed by an analysis of the records obtained at the transport level. A comparison of transport level frame occupancy (the number of cells of type 1 and 2 N_1 and N_2) that were calculated at the level 4 (from traffic strem $\Phi^{(4)}$ is performed.

The estimation of frames occupancy parameters was previously performed for the following cases:

1) Single node with 100 subscribers with single VoIP application
2) Single node with aggregate equivalent for previous traffic
3) Single node with web server
4) Single node with VoIP and Web server
5) Single node with aggregate equivalent for previous traffic conditions
6) Gaussian equivalent for heterogeneous traffic

#	Scenario	$E[N_1]$	$E[N_2]$	$E[N_1^2]$	$E[N_2^2]$	$E[N_1N_2]$
1	Voice (precise 100x)	193.73	X	639	X	X
2	Voice (MMPP(100))	193.31	X	386	X	X
3	Web only (BMAP)	X	19.31	X	505	X
4	Voice + Web	188	14.3	910	322	-11
5	BMAP(D')	197.37	X	460	X	X
6	Gaussian(2)	188,4	17,8	871	322	-15

Table 7.1.: Transport level analysis of traffic measured for different scenarios

Remark for the scenarios 1,2,3 and 5 we have multiple instances of a single model or a single aggregated model that don't allow distinct quality of service class, hence only a model for N2 and N1 or N1+N2 is possible.

Table 7.1 summarizes the results. The mean number of cells, the variance and, if applicable, the cross-correlation for each cells categories was evaluated.

For the voice case (scenario 1) and 2)) a good agreement between both models is obtained. Both mean amount of cells match precisely. The MMPP has a lower variance that the original model, because the aggregate model was not derived strictly from the second order property but from the queueing properties. A tentative explanation for the reduced variance is that the aggregate model has lost some information because of the fit of autocorrelation function was used, the fit is not done only on the basis of the first order statistics. The constraint on the variance is hence implicitly released.

The other scenarios are concerned with the heterogeneous cases. The web traffic, the complete traffic and an equivalent are compared. The N1 cells correspond to VoIP traffic and the N2 cells to web traffic. The values in the table show that the similar first and second order statistics are obtained. The distribution of the correspond cell numbers is analysed afterwards. In particular the scenario 4 and 6 are compared. Here again, it can be noticed that the BMAP (an aggregated equivalent) has a reduced variance. For the voice and web case, the global number of cells has a mean at 202.3 cells / frame. The BMAP model emits in average of 197.37. It represents a difference of about 2.5 % in mean. The gaussian models is transmitting on average of 206.2 cells per frame. That represent a difference of about 2% with the original case. These are good agreements from the point of view of traffic modelling.

7.3. System description for investigations

7.3.1. Investigation of performance metrics

Figure 7.2 shows the nodes with a test application considered for the multiple node scenario. Two approaches for this node can be considered: the one of the left, considers a node composed of the test application, of N times the instance of the VoIP model and the BMAP on/off model; the node on the right is composed of an aggregate model derived for 100 VoIP applications and 100 users browsing the web.

Two type are then introduced:

- type Ip (precise model with test application). It is the type I where a supplementary application which performance are investigated has been added
- type IIIp (aggregate model with test application). Similarly, it represent the type III node with an application for performance investigation

The typical full size simulation are performed with 50 nodes accessing the satellite. Between them, 49 nodes are of type IV and are generating traffic according to the bi-dimensional Gaussian model. I

n the simulations the number of Gaussian sources was fixed but the parameters of the model can be chosen in the list from a single node derived in previous paragraph. The number of passengers per aircraft

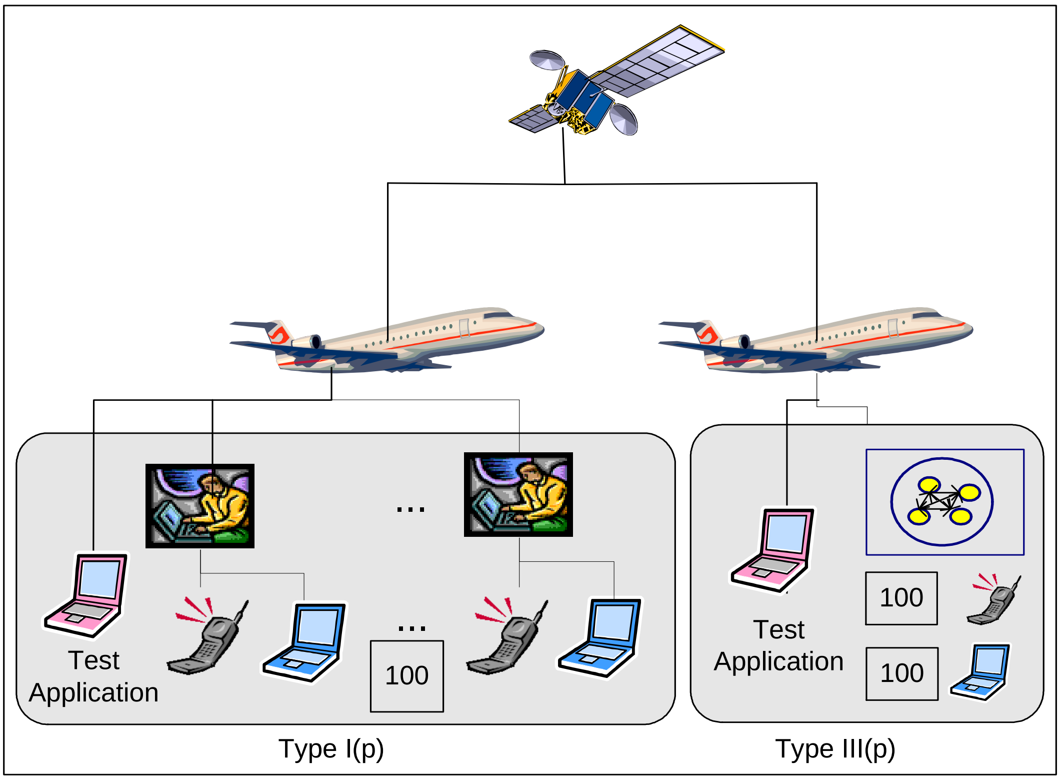


Figure 7.2.: Case considered for performance evaluation

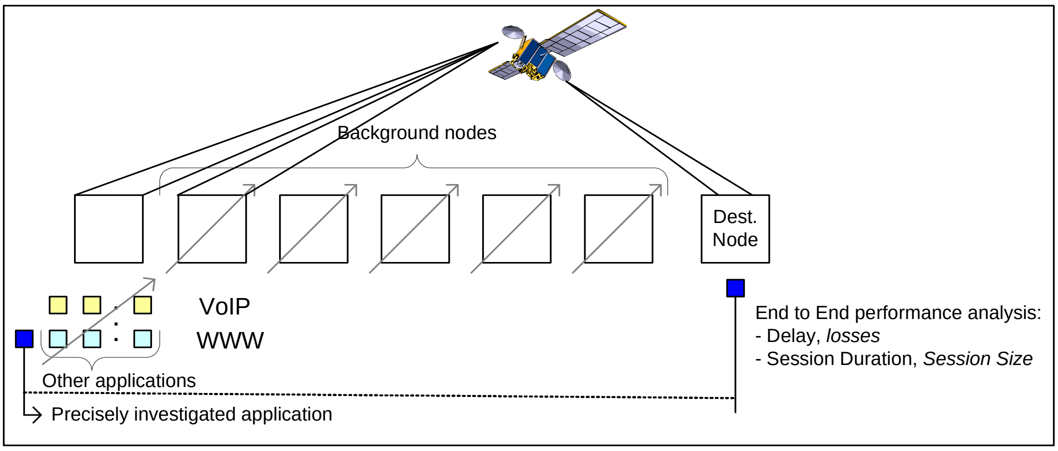


Figure 7.3.: Configurable parameters for performance evaluation case

(`N_sub_per_node`) is then the parameter that sets the volume of traffic (and the ratio of N1 and N2 cells). This parameter is then used as the parameter setting the number of passengers is the aircraft with the investigated application, and constituting the load of the node. Hence, the load of the system is fixed by this number of passenger per aircraft. The capacity of the satellite is fixed by the number of cells per frame and per node (BW). The overall offered capacity is then 50*BW cells per frames.

7.3.2. Parameters of the problem - Description of traffic streams

Because in the description of a full size system requires different parameters in order to size the system, the parameters having an influence on system dimension are described hereafter.

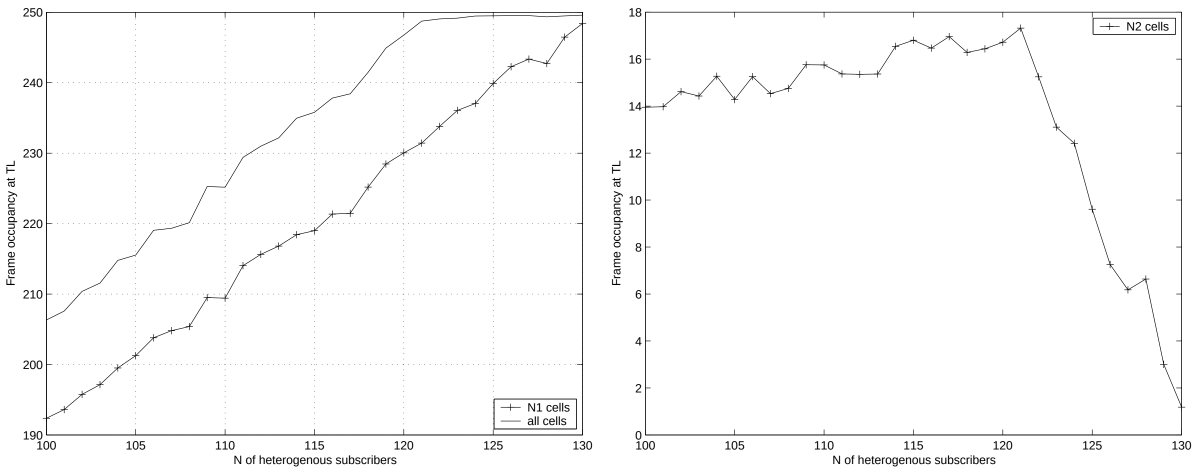


Figure 7.4.: Mean value of N_1 and N_2 for the single node case under varying load

Figure 7.3 is representing the elements involved in the performance investigation study. The key element is the test application. In the node where the test application is located, some generators representing other subscribers and applications are necessary. For the other nodes, aggregate models can be used to generate the traffic of the aircraft. The parameter ”number of passengers per aircraft” sets the number of ”other” subscribers in the node with test application on one side and it sets the parameters of aggregate model for all the other nodes on the other side. The aggregate node models are derived from a database with the values from table 6.2. By proceeding this way, the number of passengers is modelling the complete volume of usage of the system. The load in the node with the test application and the other nodes could be decoupled, but it seems better to have a single parameter conditioning the complete traffic volume. Of course, investigating the influence of the load in a particular node could be an interesting other issue, but the results will be similar to the result obtained with a single parameter.

Hence in the following system investigations, the following entities were considered: the system is studied with one test node (either of type Ip or type IIIp) and optionally (for full size investigation) with 49 nodes of another type (I,II,III or IV).

7.4. Investigations of system behavior

7.4.1. Influence of traffic volume

In this section, modifications of the traffic load are analyzed. By proceeding this way, the system is driven to a higher load, up to its maximal capacity. To monitor this effect, the frame occupancy was observed. Two cases have been distinguished: the first one corresponds to a system with single node, the second one corresponds to a system with 50 nodes.

7.4.1.1. Single Node

Figure 7.4 shows the repartition of N_1 and N_2 cells for increasing load (number of subscribers) for an unique node. The maximum bandwidth is set to 250 cells per frame. On the left side, the number of N_1 and $N_1 + N_2$ is represented. On the right side, the number of N_2 cells is represented. Since the N_1 have the highest priority, as long as the overall amount of traffic is smaller that the bandwidth limit, there is a linear increase of the volume.

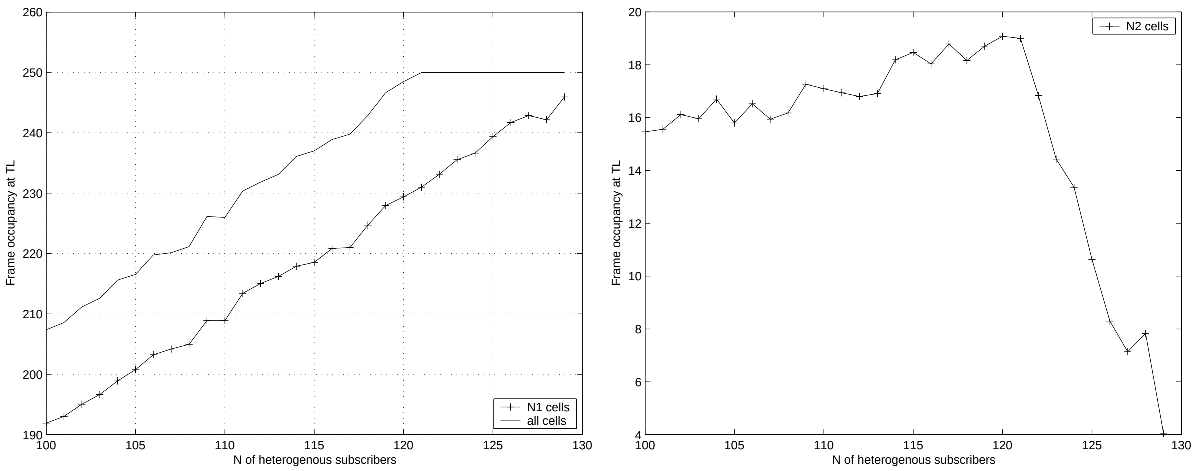


Figure 7.5.: Mean value of N_1 and N_2 for the multiple node case under varying load

7.4.1.2. Multiple Node

Figure 7.5 shows the repartition of N_1 and N_2 cells for increasing load (number of subscribers) for 50 nodes. The maximum bandwidth is set to 250 cells per frame. On the left side, the number of N_1 and $N_1 + N_2$ is represented. On the right side, the number of N_2 cells is represented. The results here were divided by the number of node considered. Hence they can be compared with single node volume.

It can be verified that the maximum bandwidth is never exceeded. A dramatic loss of capacity transferred for the type 2 cells is obtained in the last part [120,130] passengers/node. With these volume of traffic, type 2 cells can not be served at an acceptable rate. Beyond this limit capacity, any user of web traffic will not any more be served with a satisfactory level. It would be better to design the system in order to prevent that such a state is reached either by limited the usage of the application by alerting the users or by limiting the admission of new users when the capacity is close (with a given margin) of this maximal limit. It must be kept in mind that the users close their session after some time, so this limit must be mixed with a factor taking into account the session closing rate in order to reach maximal system capacity.

7.4.2. Node to transport level traffic modifications

Figure 7.6 shows the generated data volume observed at the satellite level. It can be seen that the both generated traces have a common tendency. To confirm this visual impression both density functions of the volume rate computed over a period of T=26.5 ms was displayed. The results presented correspond to the comparison of the generation of web and VoIP traffic like in the case investigated earlier (heterogeneous sources). For the precise generation the node is composed of 100 VoIP sources and a BMAP model generating the traffic for the web surfers.

The volume at node level is obtained from the record of packet size $V(t) = \int_0^t S(u)du$. The volume at transport level is derived from the number of N_1 and N_2 cells recorded per frame. It comes $V(nT) = (N_1 + N_2) \times \mathcal{S}_4$ where $\mathcal{S}_4$ is the size of cell at level 4. In the detail, the packets measured at node level are generated slightly early that they are transmitted on the transport level. The overhead for the transport level (7 bytes added to the 53 bytes of content) conduct to a increased overall volume. The consideration of the information content in the transport level has a good agreement with the node volume.

So far, the simulation approach seems to be suitable for investigation of the system behavior. This will be confirmed in the following when investigating the match of the model and their proposed equivalent using different measures (based on volume and other performance metrics) in the next paragraphs.

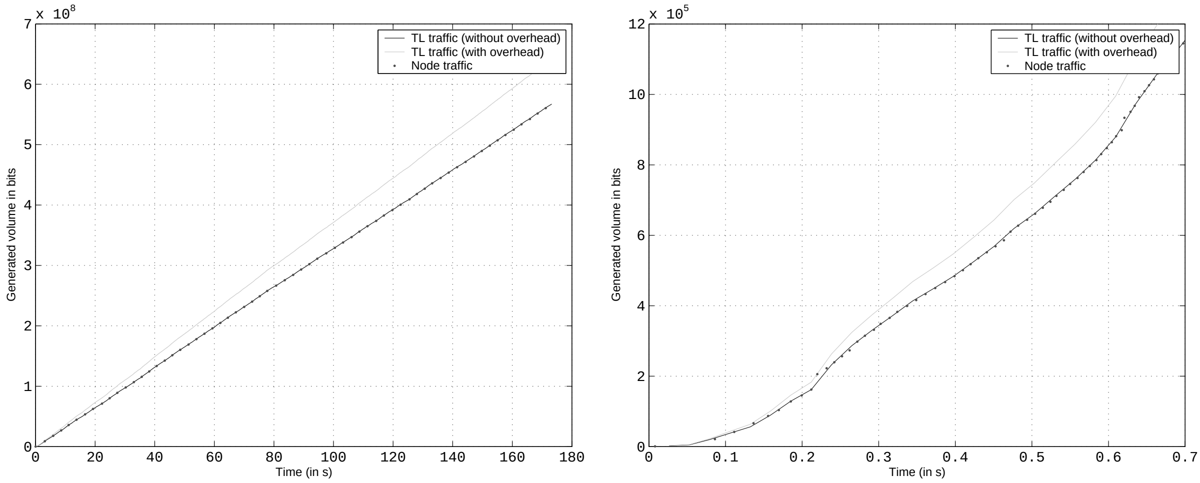


Figure 7.6.: Throuput on the satellite link for a single node

7.5. Validation of equivalent models

This section shows that the use of equivalent models produces results similar to the ones obtained with the initial traffic flows.

7.5.1. Analysis of frame occupancy

A first parameter to be analyzed in order to compare models in the frame occupancy at the transport level. Equivalent models shall deliver a similar load of the N_1 and N_2 frames, because equivalent models would imply an identical distribution of N_1,N_2. For this comparison, the frame occupancy was monitored. Two cases have been distinguished: the first one corresponds to a system with single node, the second one corresponds to a system with 50 nodes. The purpose of having single node result is to be able to analyze the results at the node point of view to check that no mismatch occurs. Furthermore, once validated, the results with 50 nodes need to be analyzed again in order to justify, that the full size system study furnish similar conclusions.

7.5.1.1. Single Node

First, results considering only one aircraft in the system are presented. The advantage is that the effect of limited bandwidth can be observed earlier and that simulations are executed quicker than full size simulations. With respect to the description of the scenario, nodes of the type I, III and IV are considered, each time with a single element.

At this point it need to be compared with aggregate models. First, the distribution of N_1 and N_2 for different models is compared by plotting the density of the corresponding frame occupancies for both type of packets. Figure 7.7 shows the corresponding results. The node traffic exhibits a Gaussian distribution that is also followed by the transport level net traffic. At the transport level, the effect of the bandwidth constraint per node can be evidenced. No more than 250 cells can be transmitted per frame. In order to fulfil this constraint, cells are delayed. This can be performed since the average traffic is below this limit (around 200 cells/frames). Both gaussian distributions exhibit a good agreement, a part for the maximum size of the $N1$ cells (type 1), and the small size packet of the $N2$ cells. This can be easily corrected using bounded gaussian distribution. Refer to annex D.2.9 for a description of such distributions.

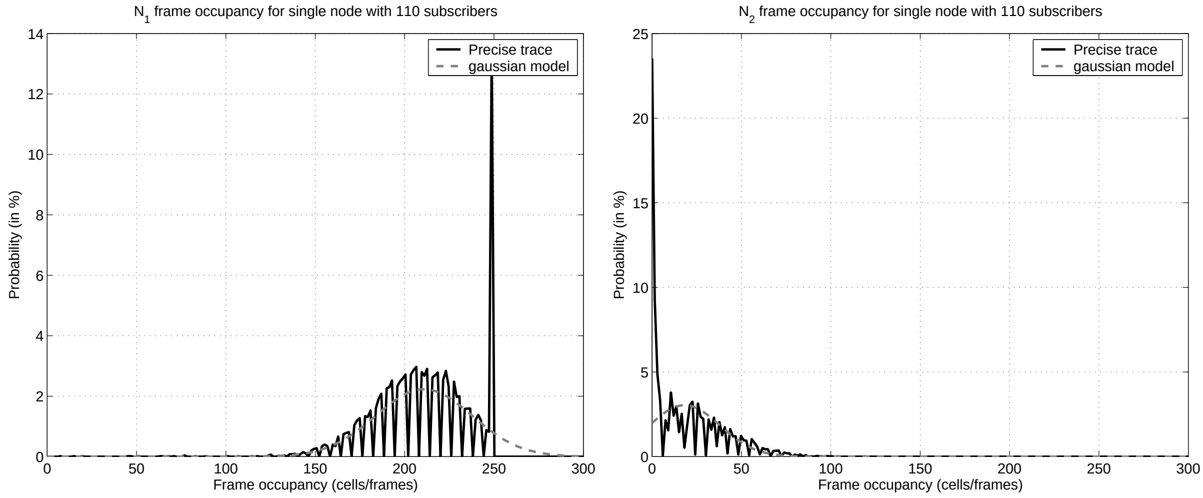


Figure 7.7.: PDF of N_1 and N_2 for the single node case

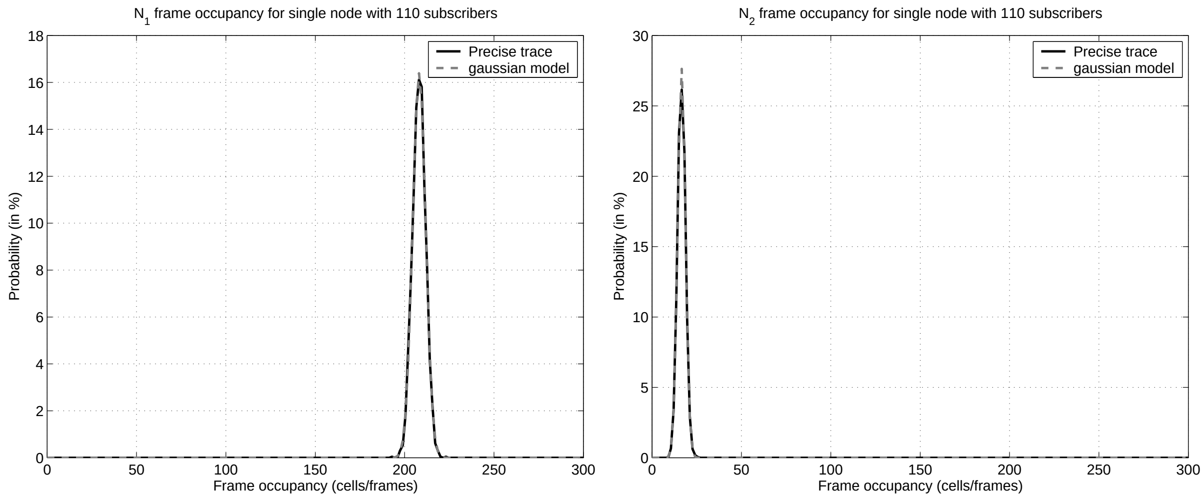


Figure 7.8.: PDF of N_1 and N_2 for the multiple node case

7.5.1.2. Multiple nodes

With respect to the description of the scenario, nodes of the type I, III and IV are considered. Three cases are distinguished. First the precise case with $N_I = 50$, the aggregated case with $N_{III} = 50$, a synthetic one with $N_I = 1$ and $N_{IV} = 49$.

Figure 7.8 shows the repartition of N_1 and N_2 (divided by the number of nodes of 50) for the case of multiple node with 110 subscribers. There is very little difference between the distribution for the precise model and a global gaussian estimate. The mean value of the gaussian distribution is then $m_1 = \mathbb{E}[N_1/50] = 208.9$ and $m_2 = \mathbb{E}[N_2/50] = 17.3$ is this case. It is very close to the value in raw 110 of the table 6.2 that were used as input for the type IV nodes.

An agreement in the distribution of the number of packet is the highest guarantee that both models matches. Nevertheless, the volume will be investigated to check from another point of view that a good match is obtained.

7.5.2. Volume analysis

In figure 7.9 the traffic generated by 50 nodes is depicted. Different composition of the traffic generated by the node have been simulated. First each node was composed of 100 users of voice over IP, then each node of web traffic, thirdly both traffic were considered and finally an aggregate model was used for each

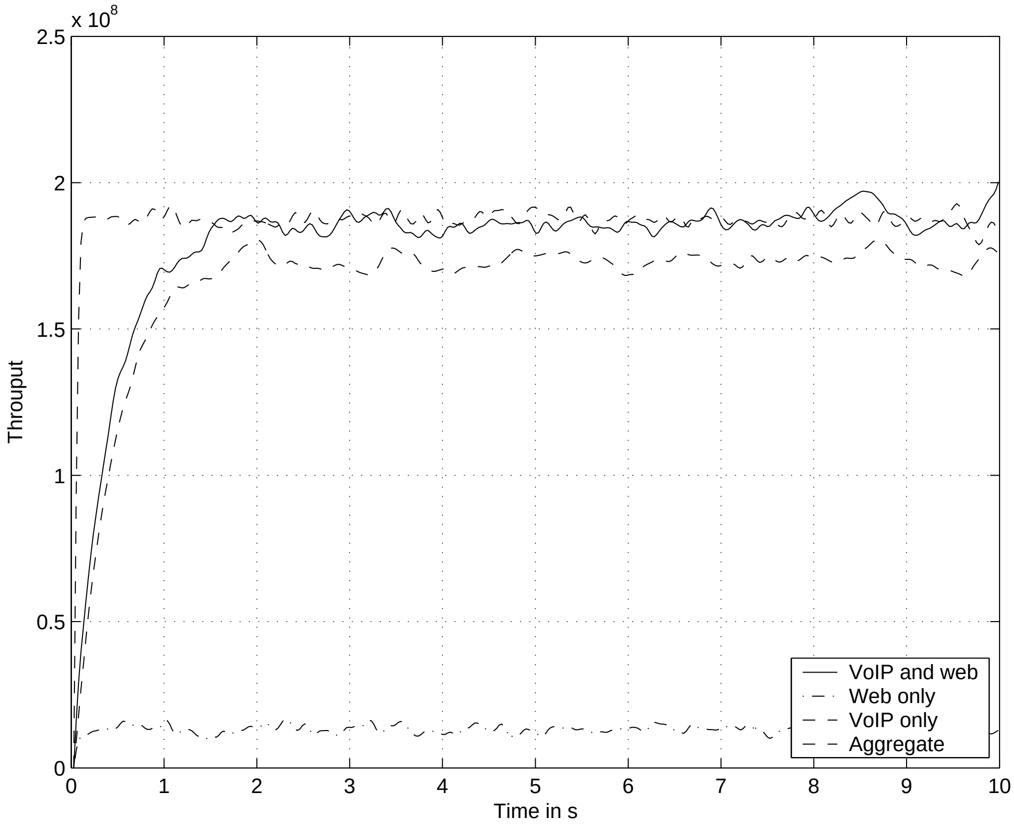


Figure 7.9.: Comparison of throughput for 50 nodes with different traffic composition

node.

A good agreement can be seen between the aggregated model and the VoIP and web composite model because they generate a similar throughput (around 1.8510^8bit/s).

For 100 subscribers, it as been derived in table 6.2 that the voice volume had a rate of 192 cells per frame and the web traffic of 14 cells per frame. It implies a volume of $V_1 = 192 * (\mathcal{S}_4)/T_f$ where $\mathcal{S}_\triangle = 60 \times 8$, $T_f = 26.5ms$ it comes $V_1 = 1.7310^8$ bit/s and $V_2 = 14 * \mathcal{S}_4/T_f = 1.2610^7$. In total, $V_1 + V_2 = 1.8710^8$ bit/s. This corresponds to the observed means.

7.6. Investigation of Web session duration

The objective of these investigations is to perform simulations with the equivalent models that were discussed previously in order to obtain indications that could help system design. The selected objective was to determine which share of surfing services are acceptable within the satellite system for aeronautical communications and to derive limits that would limit a good service provision of the considered services.

7.6.1. Single Node

Figure 7.10 shows the evaluation of the web session estimation for the reference scenario with a single node. This case considers only one node of type Ip. It can be seen that the number of users has an influence on the session duration. When the load increases (because of more users), the session duration increases: the web traffic with quality class two has more difficulties to send its packets through the system. With increasing load, the available service remaining free for web traffic is decreasing and simultaneously the load is increasing Hence, the ser

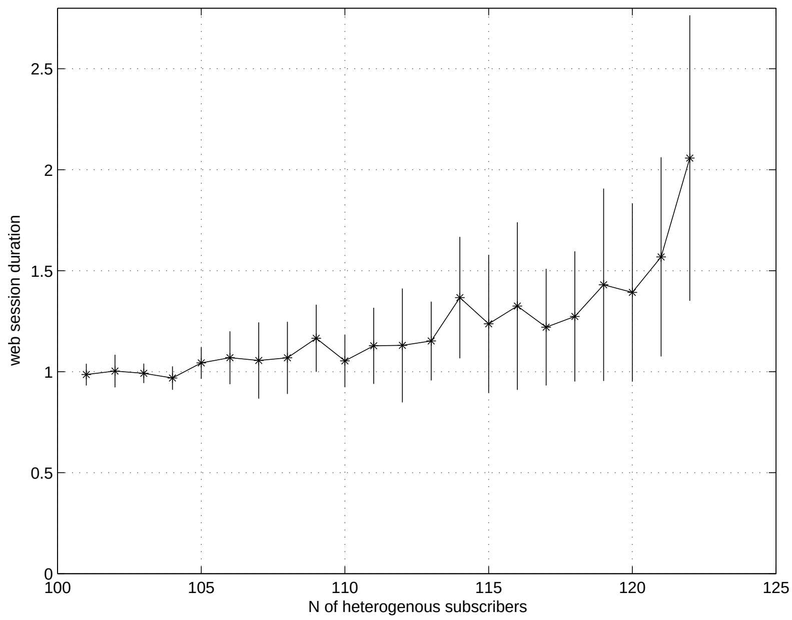


Figure 7.10.: Evaluation of web session duration (Allocated BW=250)

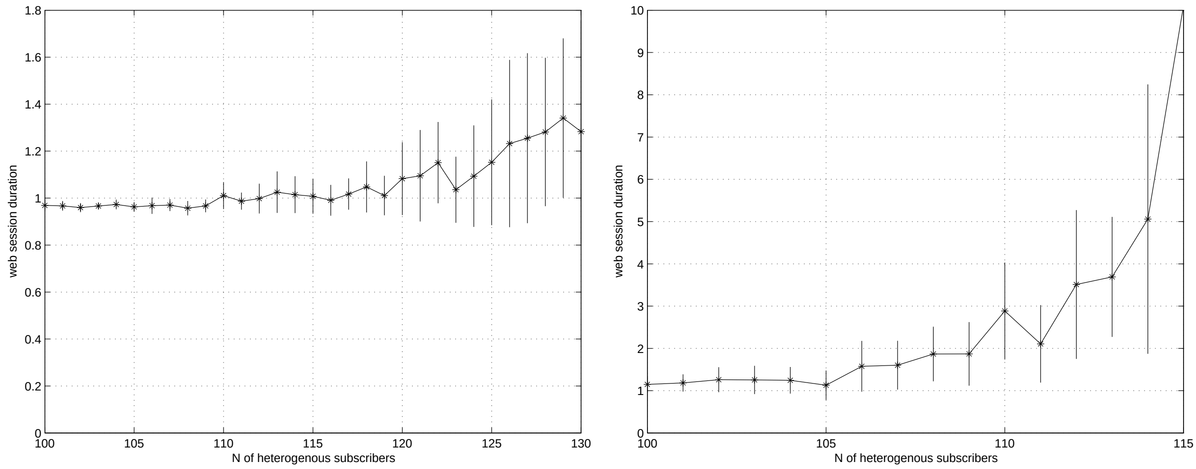


Figure 7.11.: Evaluation of web session duration(Allocated BW=275 on the left BW=225 on the right)

Furthermore the effect of different bandwidth allocation was considered: in particular the cases for 10% more bandwidth ($BW = 275$) and 10% less bandwidth ($BW = 225$) were simulated, with the same parameters as in the previous cases.

Figure 7.11 shows the evaluated session length for increasing number of subscribers. Both cases were considered: the one on left side represents a greater bandwidth allocation, the one on the right a reduced bandwidth. For $BW = 275$, it can be seen that the session length increases with the load, what means that when the utilization increases, it gets more and more difficult for web traffic to be transmitted over voice traffic. In comparison with the bandwidth allocation of 250, the increase of the session length is anyway slower. For $BW = 225$, the increase is quicker, with a attained limit where the session length reaches an unacceptable limit.

7.6.2. Multiple Nodes

A similar analysis was performed as previously but with synthetic nodes of type IV in order to obtain a load of type 1 and 2. These node model was making use of the parameters that were determined for the

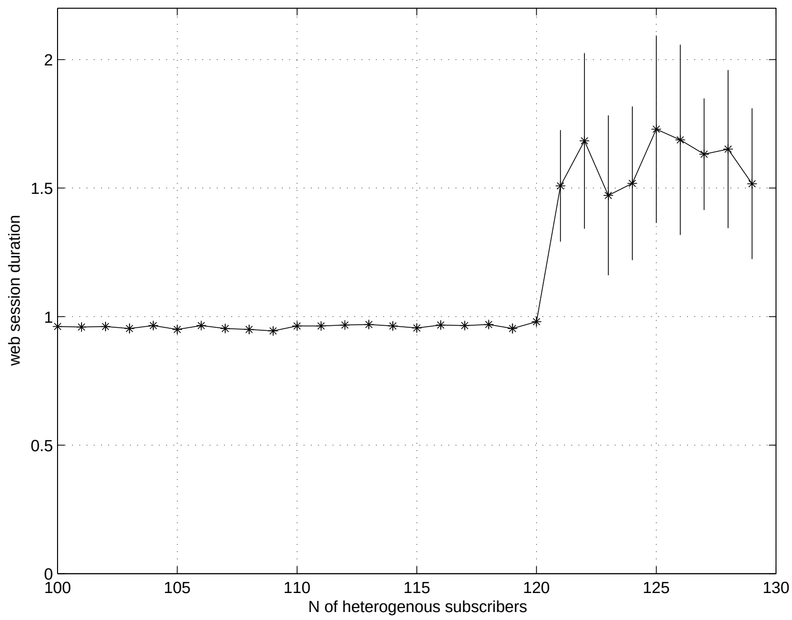


Figure 7.12.: Evaluation of web session duration (Allocated BW=250)

gaussian model (that were determined in table 6.2 in previous chapter).

Figure 7.12 shows the session length evolution for increasing number of users. This is quite different from previous results, because a plateau in the interval $[100, 120]$ can be seen. In this range the web traffic can be served without increase of the average session duration. This effect is due to the share of resource that occur thanks to the collective usage of all class 2 traffic between all users. This scheme was used in order to provide such benefits to the second category of traffic. In the loaded case, the estimate of session length is shorter than in the non-shared case. In the some cases with too much load, the number of estimates for the session length was reduced, telling that the sessions were so long that the last sessions were not completely received because their packets are stalled in a queue in the system. Another point to mention, is that the use of the gaussian ”load” model was derived with a bandwidth constraint set at 250 cells. The output per node as such a model is depicted on figure 7.5. For high traffic load, a decrease in the amount of the second category cells is included in the model. For better results, a linear model could have been used, but it places the study in another scenario.

Similarly, for a reduced bandwidth allocation, similar results are obtained (plateau), the limit at which performance start to be reduced is attained at $N = 107$. This shows a straightforward dependency between the allocated bandwidth and the size of this plateau: when more bandwidth is shared among the different web users, an higher benefit in term of kept performance can be obtained.

7.7. Methodology for system dimensioning

In this section, a method for system design is described. Figure 7.13 shows a methodology for investigation and dimensioning of a complete aeronautical system. The system is supposed to be similar with the considered numerical case: different aircraft are served by a satellite system that have different traffic profiles. For the design of the satellite system it is important to know which traffic is transmitted by the satellite.

For the analysis of this problem, the following approach is proposed. It was derived by analogy with the investigated case and make of course use of aggregate models. It is assumed that the number of aircraft types is not so high, so that an aggregate node model can be derived for every aircraft considered in the system (for example, 5 type of aircraft could be possible).

The first step consists in the derivation of equivalent node model. For this purpose, a precise simu-

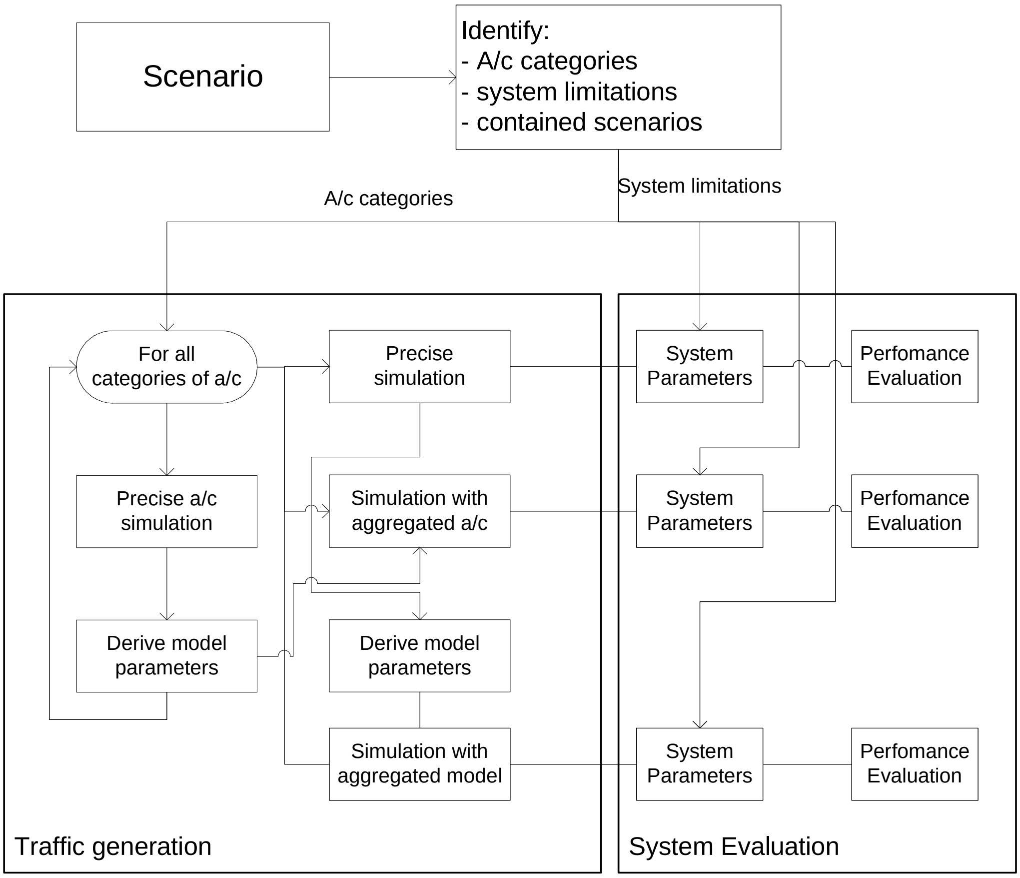


Figure 7.13.: Simulations principles

lation is required of each single aircraft. From the recording for such simulation (of reduced size), the parameters of an aggregate model for each aircraft can be derived. A comparison may be performed in order to check that both model produces similar outputs. If different aircraft types are considered, an aggregate model is required for each of them.

The second step consists in the derivation of an aggregate model for all aircraft, using the models derived in step 1. From these data, a global model is available (at transport level) for a complete system investigation. The results obtained at system level form the reference results.

In order to validate the model, precise simulation could also be conducted in order to check that both approach match. Since in general, these results are costly to obtain. The goal of the aggregate model usage is to obtain the same results than in the reference case. First a first round is dedicated to the derivation of aggregate models for nodes (aircraft in that case).

The following aggregate models are then required

- aircraft aggregate model (one per type)
- global traffic model

The derivation of models can be performed using similar methods as the one that were derived in the previous paragraphs. The methods for the investigation of equivalencies between model can also be used again.

7.8. Performance analysis

In this section, the focus is given to the analysis of performance metrics gathered during simulations. The results collected here are based upon a table of the number of events generated during a simulation.

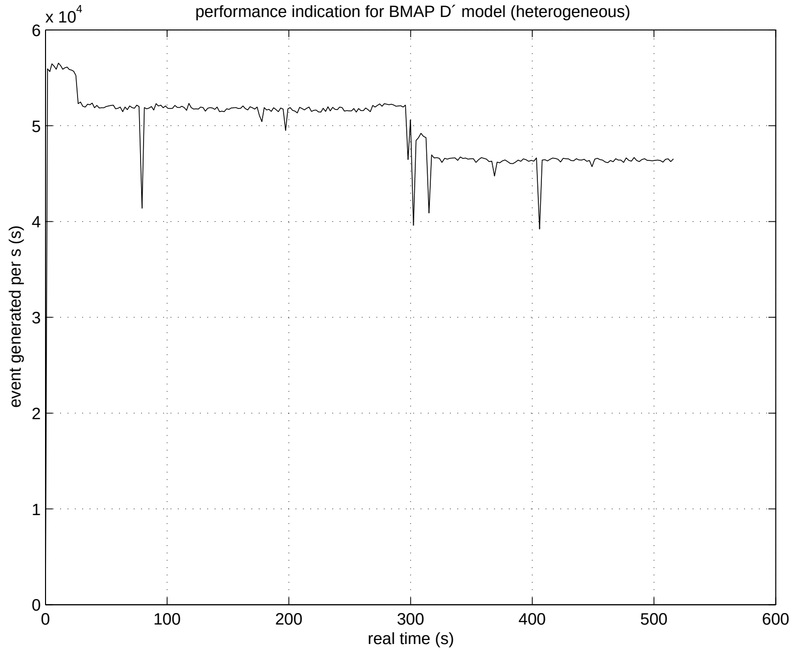


Figure 7.14.: Evolution of simulator performance for single node case

First, the optimization of simulator parameters that have an influence on the system parameters is shown. The purpose here is to present some results showing the performance obtained and concerning some optimizations that were performed in order to increase the simulation speed by comparing the execution speed of different models.

7.8.1. Simulator optimization

In this paragraph, three simulations are presented with the focus on the analysis of simulator execution time. The following timing are used: i) the simulated time is the time that is used in the simulator and corresponding to the time managed by the simulator ii) the real time is the time that is used on the computer on which the simulator is executed. The first one is a single node simulation with parameters that enables fast results. The second case is a simulation with 50 instance of the previous node with identical parameters that was only executed at a very low speed and where poor performance was suspected. The performance monitoring has highlighted this problem and a solution has been proposed. With the modified case, the performance are analyzed again, with a simulation executing at acceptable speed indicating that the problem has been solved.

Figure 7.14 shows the number of events generated for the precise simulation of 100 voices sources. This number is almost constant with a small deterioration over time. It means that the number of events generated over time is constant. It is in phase with the model that with an high number of users, should have a stationary behavior (”constant” number of sources in the constant bit rate ON phase). The deterioration of performance is probably due to an internal saturation of the simulator (for example increasing access time to the log file as time increase). When the relation between real time and simulated time time is monitored, the linear behavior is not affected, this decrease has not big influence on the simulator execution time.

When the simulator is used with the same parameters, for a scenario with 50 nodes, the simulator is taking more than 50 times more for execution and gets very slow with increased simulation duration. This is confirmed by an examination of figure 7.15 where the number of events treated by the simulator against real time is monitored. There, it can be seen that there is a dramatic reduction of the number of events. It means that the simulator is then busy by performing other tasks as processing events. By

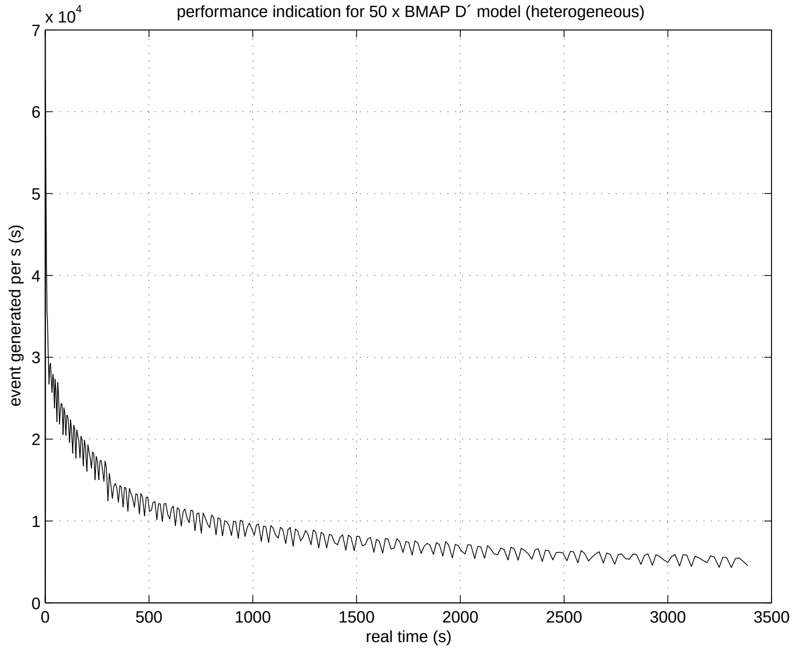


Figure 7.15.: Evolution of simulator performance for 50 nodes (with same settings as in previous case)

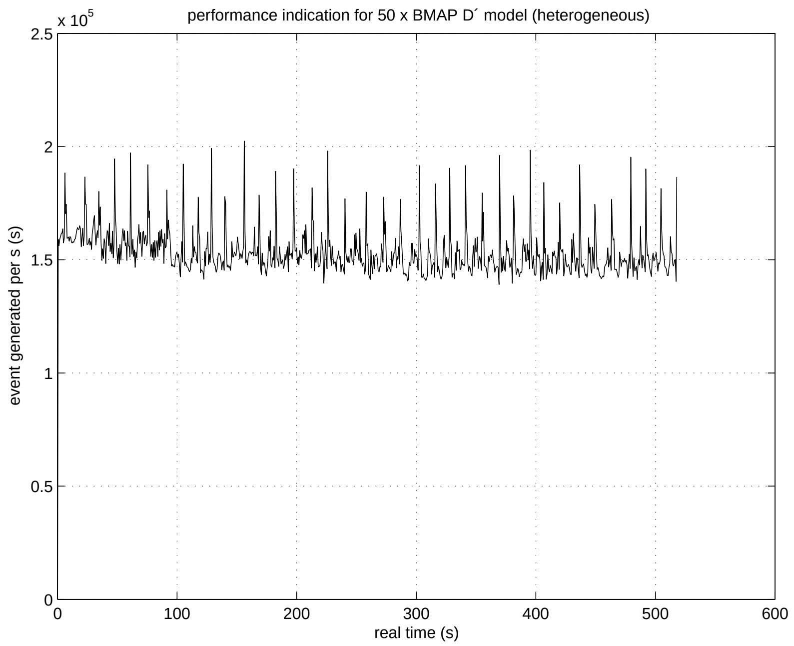


Figure 7.16.: Evolution of simulator performance for 50 nodes (with optimised settings)

monitoring the sizes of the queues in the simulator, a increasing (high) size was seen for some of them. This evidenced that proper settings for the queue processing time is necessary otherwise the queue is unstable.

Figure 7.16 shows the same graphs as before when the simulation parameters were adjusted. For example, at node level and transport level, the queue service time was reduced by a factor 50 because the load is expected to be increased by a factor 50. With this proper settings, the simulator behavior is more stable. The burstyness observed on the picture is a consequence of the BMAP models, that have states with the transmission of more packets. These events are visible here. The real time for simulation is of

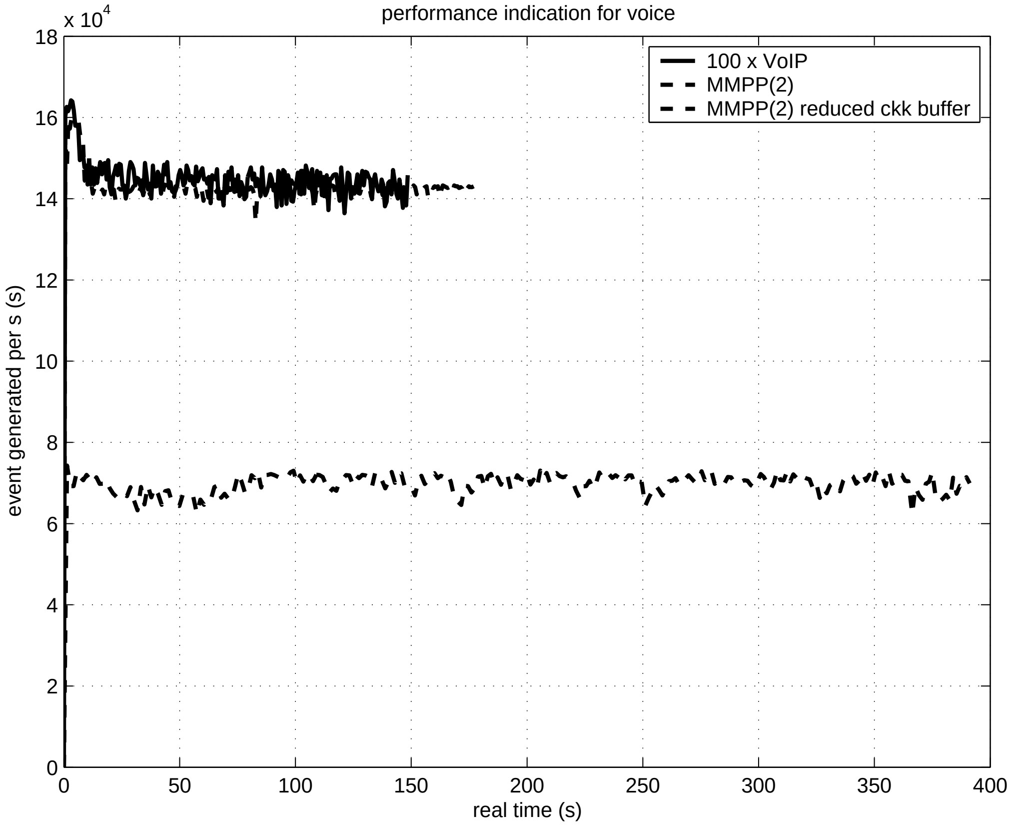


Figure 7.17.: Evolution of number of events manipulated by the simulator for different models

about 500 s, it was of more than 3500 s in the previous case.

A second improvement of the simulator execution time was obtained, by reducing the amount of metrics logged. By proceeding like this, less delays due to hard disk access are obtained. But of course less data after simulation execution are available... So a careful selection of the collected material is necessary.

7.8.2. Comparison of performance measure

In the following, the execution speed of different models is compared. Figure 7.17 compares the number of events generated versus real time for the execution of a simulation scenario with 100 precise voice sources, a MMPP(2) model and a MMPP(2) model where the autocorrelation estimator was disabled (set to a very small buffer). When comparing the time spend in the different module, it was found that the simulator was spending a lot of time in this analysis module. When this module was suppressed a similar speed for the MMPP(2) that for the precise model was achieved. This brings to the conclusion that the complexity of aggregate models, making them generating some more events for the change of state without packet transmission, is not penalizing the simulation speed since the difference in time is negligible. All three simulations correspond to the same simulation duration, the real-time execution duration of each can be seen for each of them: the precise simulation has required 150s, the MMPP(2) without analysis was executed in 180s, and the first MMPP(2) lasted about 400s. No big loss of execution time are due to the use of the aggregated models.

For the models at the transport level, the gain is even clearer, because the model used is simple and the generated traffic is injected directly at the node level, so logically less monitoring effort is needed. Moreover, the traffic is directly in a format compatible with the level $\ell = 3$, so no operations are required for the packet conversion.

It must be noted, that when these aggregated models were implemented as an equivalent source model, they will never generate less ”events” that a precise model. If they will act so, it will mean that less packet are send, and then it will be an unrealistic representation of the traffic sources.

7.9. Summary

In this chapter, equivalent models were used in a full size system with heterogeneous applications. Most of the results are related to the investigation of how good the equivalence of the proposed models are to validate the obtained equivalents. After these evaluations, a short study evaluates the deterioration of a web session duration when the traffic increases to demonstrate that this service can be accommodated with the other applications. From this example, guidelines for capacity analysis with traffic models are described. Finally, performances of the equivalent models are shown to prove that their usage can be realistically considered and is competitive with the original modelling.

Mon Quatuor comporte huit mouvements.
Pourquoi ? Sept est le nombre parfait,
la creation de six jours sanctifiée par le sabbat divin;
le sept de ce repos se prolonge dans l'éternité
et devient le huit de la lumierère indéfectible , de l'inaltérable paix.

O. Messiaen à propos du Quatuor pour la fin du temps.

8 Conclusions

8.1. Summary

In this work, the following investigations have been performed.

- Definition of an abstract model to describe hierarchically the traffic models.
- Review of traffic models for the different levels
- Definition and validation of aggregate models (first a simple case, then the aggregation at node level and finally complete pyramidal aggregation)

The main results will be summarized in the next subsections.

8.1.1. Aggregate traffic model pyramid

In Chapter 2, a hierarchical model called the aggregate traffic pyramid has been introduced.

A number of definitions that enable the description of generic traffic scenarios have been introduced. The hierarchy of the different levels is represented with a pyramid with four levels: on the base the applications are the origin of the traffic, then the subscribers and nodes levels are the above levels providing access to the transport level.

First, the four levels of the pyramid have been described: the application, the subscriber, the node and the transport levels were introduced and are the basis for a hierarchical modelling. The natural applicability to a general communication system and in particular to the aeronautical case investigated numerically was shown. The requirements for traffic models considered on these different levels are investigated: models representing the traffic streams at different levels shall be used in order to provide scalable models. Prior to use of equivalencies, a careful investigation of validity is necessary. Moreover, the procedure for the derivation model parameters will need to be defined. Simultaneously this pyramid can also be used to describe the data entities manipulated inside each level. Packets manipulated at each level were noted $\mathcal{P}_1$,$\mathcal{P}_2$,$\mathcal{P}_3$,$\mathcal{P}_4$. Through the transmission over the different levels, these packets go across inter-level interfaces that modify their sizes and are concentrated over the next level. These mechanisms can be described in the framework of the pyramid.

Then, it has been shown that the pyramid model is a good tool to describe the aggregation of the sources towards the final level. The framework is refined by a characterization of each of the levels. Then the traffic models definition are given. For this purpose, the abstract units are introduced, that enable to describe traffic streams $\Phi = (\mathcal{S}, \mathcal{I})$. With these definitions the problem of finding equivalent traffic models is reformulated by the necessity to derive independent generators for some dependent generators that are built using the aggregation operator. Furthermore three operators where introduced in order to describe the modification of size and time at the different levels. Each of the operators corresponds to a simple operation: addition, level packet conversion and level conformation. The characteristics of these operators are reviewed. Finally, the relationship between the pyramid model and the OSI model and the protocols is addressed.

Finally, the pyramid model can be used to get performance metrics. The first goal of the metrics is to enable the comparison of two distinct traffic streams. The different evaluations of the metric are discussed: instantaneous, time varying, sequence, mean. The points for evaluation of the metric. The details of metrics for stream comparison (to evaluate how far two streams are) and for evaluation within the pyramid (either evaluated at a single point or at packet) are then exposed. All definitions for these metrics are given with indication on the evaluation procedure. With the information provided, the evaluation of performance can be conducted for any system.

8.1.2. Traffic models in the pyramid model

In chapter 3, the mathematical models for traffic modelling have been introduced with some of the details provided in the Appendix A. For the models selected in this work, the focus was then given to the derivation of the parameters required for the aggregated models.

First, the single streams traffic models are reviewed. These models could be used as independent generator at any level. Models with gradual complexity are exposed: Renewal based models are first presented (a class including Poisson and PH-distributions), then the Markov renewal process which enables to derive the MMPP, BMAP models and their circulant version. This gives a good understanding of the traffic models. The description begins with the presentation of the renewal processes. The renewal process arises from renewal theory and represents a generalization of the poisson process. The derivation of their renewal properties like the distribution of time between events are derived. The superposition of renewal process can also be described theoretically. The observation of a renewal process can be facilitated by the use of the index of dispersion of intervals (IDI) or the index of dispersion of counts (IDC). Finally the alternating renewal process is described. This first family of models is quite interesting for the modelling of traffic streams of applications with single size models. Their implementation is quite easy (only a distribution of the inter-arrival is needed), the theoretical exposure shows what kind of characteristics of the models can be derived. Afterwards, the PH-distributions are explained. These distributions have a particular form for the inter-arrival distribution, in this case more simple expressions for the characteristic functions can be obtained. As an example, the model for a voice source was derived. After this, Markov renewal processes are presented. A big number of models belongs to this family and are reviewed afterwards. The principal theoretical formulas useful for different computation with these models are recalled. An interesting result on superposition of such process is recalled that explain the states extension for the consideration. After having given a broad spectrum of potential mathematical models for the modelling, three particularly suitable models are introduced: the BMAP, the MMPP and the cMMPP models seemed to be good candidates for traffic modelling. First, the BMAP is introduced: this model is a Markov renewal process, that is best written with matrix notation. It enables different size of packets to be modelled, hence can be used in many cases. Moreover, the queueing characteristics of such processes can be analytically derived. To avoid too much formulas, these results are presented in annex C. In chapter 6 the theoretical results are compared with the simulated measurements. Similarly, the MMPP process is introduced, a particular case of BMAP with single size packets. The formulae for the computation of characteristic values are exposed. The waiting time function can also directly be calculated, these being the basis of one method for the derivation of parameters exposed later on. The cMMPP is a particular form of MMPP, for which many statistical properties can be derived in particular in the spectral domain and are particularly suitable for the study of aggregation of models because it was shown that the aggregated spectrum is the addition of the individual spectrum.

Secondly, models of potential interest in the pyramid modelling are considered from the practical point of view. First, models with adequate parameters for the single applications are presented with the typical values that could be used in order to model properly the traffic generated by the application. At this level models are suggested for voice, video, web models. The VoIP model is a constant bit rate on/off model. The video model, which is Gamma/Pareto model including self-similarity, is described even if the investigations concerning self-similarity are not presented in this work. The web modelling is presented with the ETSI model, the BMAP model, and results obtained after measurements in a particularly interesting case. At the subscriber and node level the potential models for aggregation are mentioned: here the model referred in the previous section will be intensively used as potential equivalents. At the transport level, a gaussian model is presented as a potential ("light weight") equivalent to the measured trace.

Thirdly, the procedures for model parameters derivation are tackled. The best model would be useless, if its parameters can not be derived handily. First, the focus is given to two different methods for the derivation of parameters that are extensively described. The first method is based on the measurements of the number of arrivals versus time and some derived graphs (derivative and second derivative estimates). The validity of the method is hence dependent of these estimates. The second method uses different inputs: autocorrelation, queueing asymptotics, mean arrival rates. It is an iterative method, that finds a solution matching different criteria. A condition for convergence is that all initial estimates need to be available, so the method can be used only at high load. After this two methods, the derivation of parameters for aggregated trace for video superposition is explained and the results for different movies are presented. Finally, the EM (expectation maximization) method is also explained as a powerful algorithm for the derivation of parameters. The principle of the method is first explained in general. Its usage for the MMPP and BMAP models is then particularized and two procedures for initialization (initial guess) are presented. They have an impact on the speed of convergence of the algorithm.

Fourthly, some theoretical results related with the aggregation of traffic models are recalled. The drawback of these theoretical formulae is the state explosion that occur with the number of sources. The case of on-off models is mentioned because the effect here is more limited. Afterwards different possibilities for the simplification of the model are reviewed.

Finally, a classification of the models is indicated to show how a model can be extended in order to increase its modelling capacity. As a conclusion, the hierarchy between the different traffic models described before is presented and the corresponding models use is justified. This chapter has then given an overview of different traffic models and their usage within the aggregated traffic pyramid has been shown.

8.1.3. Description of a practical case with the pyramid model

In chapter 4, a scenario for aeronautical communications is introduced and the operators and levels of the aggregate traffic pyramid are described for this case with realistic values. The numerical values of importance for the simulation are presented and justified. The background ideas justifying a simulation approach are summarized and some critical modules of the simulator are presented.

First, the case of study is described more precisely: an aeronautical communication system is studied which furnish internet and voice communication to passenger of an aircraft. The necessary entities are described within the pyramid model. The data manipulated are described with the vocabulary introduced previously. The parameters for the data size are fixed, and the different mechanism considered in the transmission process are particularized. An overview of QoS enforcement policies and of resource allocation mechanisms used typically in satellite systems is given. This is terminated by an indication of the number of actors considered for a standard case, that is forming the reference for numerical investigations.

Secondly, the operators introduced in the theoretical part are explained for the case considered. A proper description of the addition, level conversion and level conformation operators, as they are really operating at the subscriber level, node level and transport level. The most "influent" operator is the level conformation for the transport level: it modifies the streams because the bandwidth constraints are significantly modifying the traffic stream. The role of inverse pyramid operators is emphasized because it is necessary to compute the end to end performance metric.

Thirdly, the choice of a simulation approach is justified: studying a system by simulation is a useful method in order to investigate telecommunications systems. The principles for the design of the simulator are explained. The different levels of definition of a simulation are detailed. Then, the benefits that are expected to be demonstrated in the simulator are listed. Afterwards, the simulator is presented, the architecture of the simulator for traffic investigation is shown: the different levels are implemented and the transmission of traffic streams from level to level occurs in accordance with the pyramid module. The complementary part dedicated to result post-processing and model parameter estimation is also shortly described. After this, the simulator is described in relation with the pyramid level. Here, level by level the different operators are described, so that the construction of the stream at transport level from the streams at application level is clearly identified. The cost in terms of supplementary bit rate for the conversion within the different layers is shown. Then, the principle of traffic equivalencies are explained by combining the theoretical notations with solutions that would be practically usable.

Fourthly, the main modules implemented in the simulator are presented: the modules for traffic generation are presented to highlight the principle for implementation and the modules that collect performance metrics. On the generation side, a simple model is first shown and afterwards the BMAP model implementation is detailed. On the performance analysis side, the principles of the models for queuing estimation, end to end analysis and packet recordings.

Finally, a summary of the data collected during a simulation is provided. The raw data obtained after a simulation run is described and the data meaningful for the assessment of system performance are listed. The measurements necessary for simulator performance evaluation are also described. Afterwards, an analysis of expectable performance gains is given. This analysis shows that the complexity can be reduced with aggregate models, but that by choosing cross level models, the number of events can be substantially reduced at the end. For equivalent models having their input at the lowest level, this gain can not be attained.

8.1.4. Validation of aggregate models

The purpose of the Chapter 5, 6 and 7 is to show the equivalence between the aggregate traffic models and their original counterparts.

Fist case: voice only In Chapter 5, the results of aggregation for a first case are explained when only VoIP sources were considered. The goal here is to give a first view of how aggregation can be considered, to show the different models that can be used and to justify some of the equivalencies obtained.

First, the case considered is explained in details to show what are the original models and what kind of equivalent models are considered. This case is of course described within the pyramid model.

Then, the particular models at each level are emphasized. At the application level the behavior of the VoIP model is shown. At the node level the properties of the aggregated flows are investigated: the distribution of data throughput per source in the aggregated stream is investigated to show the averaging effect and the autocorrelation shows that the superposition of different number of sources conduct to the same shape for the autocorrelation function. The procedure for the parameters derivation is discussed again showing the results that were obtained at node level. Two of the three methods were efficiently used for the derivation of parameter of an MMPP(2) for variable number of passengers. Then at transport level the results are presented for the reference case. At this level, a gaussian model derived at node level is used. The match between these models and the measured traffic at transport level is good and is visualized by the coincidence of their probability distribution functions. The investigations are summarized by a review of the different traffic models used in this first example.

Afterwards the emphasis is given on the justification of model equivalencies. The focus is first given to results collected at node level and then at transport level. At node level, the traffic volume and the queuing properties of the traffic stream are shown to demonstrate that very close estimates for the original and the suggested equivalent are obtained. At transport level, the overall throughput and its distribution are compared to show that a good agreement is obtained.

Finally, the performance results that were obtained in that case are discussed. First the performance of simulation in terms of number of events and ratio between real and simulated traces are obtained. Secondly, the distribution of delay of packet are were measured for different models is compared demonstrating a very comparable distribution.

Also, this chapter is presenting the first investigations of traffic models considered at different levels that were articulated following the principle of aggregate traffic pyramid. In the next chapter, similar results are exposed for more complex traffic scenario. The aggregation will be done progressively in two levels.

General aggregation at node level In Chapter 6, the results of aggregation at node/aircraft level (level 3) are explained. In this chapter and the following one, different traffic sources are considered (a mix of voice and web traffic). The difference with previous case is mainly that different packet sizes are transmitted and that the two traffic stream with their own quality of service category are considered. In particular, the derivation of node models parameters that furnish aggregate equivalents is investigated. The equivalency of these models with their ”precise” equivalent is investigated.

First, the different investigated scenario are presented to show which kind of results are collected. The material collected during a simulation is summarized.

Secondly, the consideration of additional services requires of course appropriate models. For the addition of video service, MMPP(2) models could be derived using an approach similar to the one used for voice models. For web service, a dedicated BMAP(3,3) is devised . This model is first discussed as an aggregated model (or as a global traffic stream), it is then modulated by an On/Off process in order to be usable as a single model. Aggregation of these single models are also evaluated to prove, that equivalency are obtained. For the global model, queueing characteristic estimates can be obtained that conform quite exactly with theoretical estimates.

Thirdly, a BMAP model for the heterogeneous traffic is derived. After classification of the different packet size, using the EM algorithm, the model parameters could be derived.

Fourthly, gaussian equivalent models are derived for different node composition in term of number of passenger per aircraft. Two different bandwidth limitations have been considered. These results are inputs for the study performed in the next chapter.

Lastly, an analysis of the generated traffic volume is performed in order to justify the equivalencies of models. To show the influence of the bandwidth settings (corresponding to the level conformation operator for level 3 to 4), the node throughput with different bandwidth settings are compared.

General aggregation at transport level In Chapter 7, the aggregation of nodes was investigated to prove that the method could be repetitively used at the transport level. At this level, the final justification of the method can be given. Moreover, a scenario where system dimensioning can be done using the now familiar models is exposed. The results here have a practical nature, since estimations of future system performance metrics are obtained, like for example the analysis of the influence web traffic volume on a loaded system and the evidence of a point at which the application will not be served well any longer.

First the case investigated is presented to present succinctly the entities and the model considered. The material collected during a simulation is summarized.

Secondly, results from previous chapter (node equivalent) are analyzed at the transport level. The frame composition is analyzed in detail both from the statistical and from the repartition point of view. The goal there is to check that the equivalencies at node level are also valid at transport level.

Thirdly, the parameters that will be modified during the different simulations are explained. This is done by enumerating these parameters. In particular, the number of passengers is an important parameter for the description of the considered scenario and will be used in many of the presented results.

Fourthly, for a full system, some characteristics of the generated traffic are analyzed. This illustrate the behavior of the system with different size and illustrate the case under study.

Fifthly, the equivalencies of the models are studied. The purpose here is to justify the replacement of precise models by other more macroscopic models. In focus here is a gaussian bivariate model that can well describe the frame occupancy. The equivalence of the model in volume is also checked.

Sixthly, a performance metric is investigated in order to give estimates of system performance with aggregated traffic models. The evolution of web session duration increasing when the load is increasing, could be estimated with the aggregate node models. Two cases have been analyzed: the first for single node to prove a logical duration increase and the second for multiple node where the benefit of resource sharing could be shown.

Afterwards, an iterative methodology for traffic dimensioning of an aeronautical system is discussed. This approach is based on the aggregate traffic pyramid and should enable the design of systems similar to the case investigated previously. Aggregated models are derived to progressively model the overall traffic.

Finally, an investigation of the number of events generated by the simulator representative of the simulator general performance was performed. A first case presents the optimization of the simulator settings performed through the monitoring of the simulator performance. A second case compares precise and aggregated models and how the performance of an equivalent model could be increased.

8.2. Interest and relevance of this work

8.2.1. Summary of achievements and findings

In this research work, the following results have been obtained. First an abstract baseline for the description of traffic generation scenarios has been proposed. Then some theoretical traffic models were reviewed and implemented in a simulator tool. Different procedures for the derivation of the parameters have been compared. These theoretical developments have been completed by a numerical analysis.

The abstract representation of the traffic models was for the first time represented using a pyramid model. This organization is able to handle traffic model at different level of abstraction. It represents an alternative between high level simulation using just one packet level and a high precision modelling including all protocols required in the traffic transmission.

The theoretical presentation of the traffic models gives to the interested reader a good picture of models with different complexity. They can be of interest for modelling of other type of traffic or more generally for the representation of any time series. The focus is given to heavily tested methods for the determination of the parameters. The most applicable procedures are described step by step.

Completing these investigations by a numerical analysis is a good mean to switch to the practical implementation of the traffic models in a (discrete event) tool for simulative investigations of the models. The overall framework for the investigations is presented with the most important details concerning the architecture of the simulator and the implementation of key modules. First, a simple case was investigated that is able to justify the usage of different models for different simplifications. Afterwards, more general cases were investigated with more diverse application mixes, at node and at transport level.

8.2.2. Relation with project work

In the following section, the work of this thesis is related with corresponding work carried during the activity at DLR. The project work was the background (sometimes quite time consuming) that has fruitfully put the seeds for the developments of some ideas. Moreover these projects were sometimes a field where the ideas developed in this thesis could be applied and tested in practice. For these both reasons, the activity conducted in those projects will be succinctly highlighted with a particular focus on the relationship for this work.

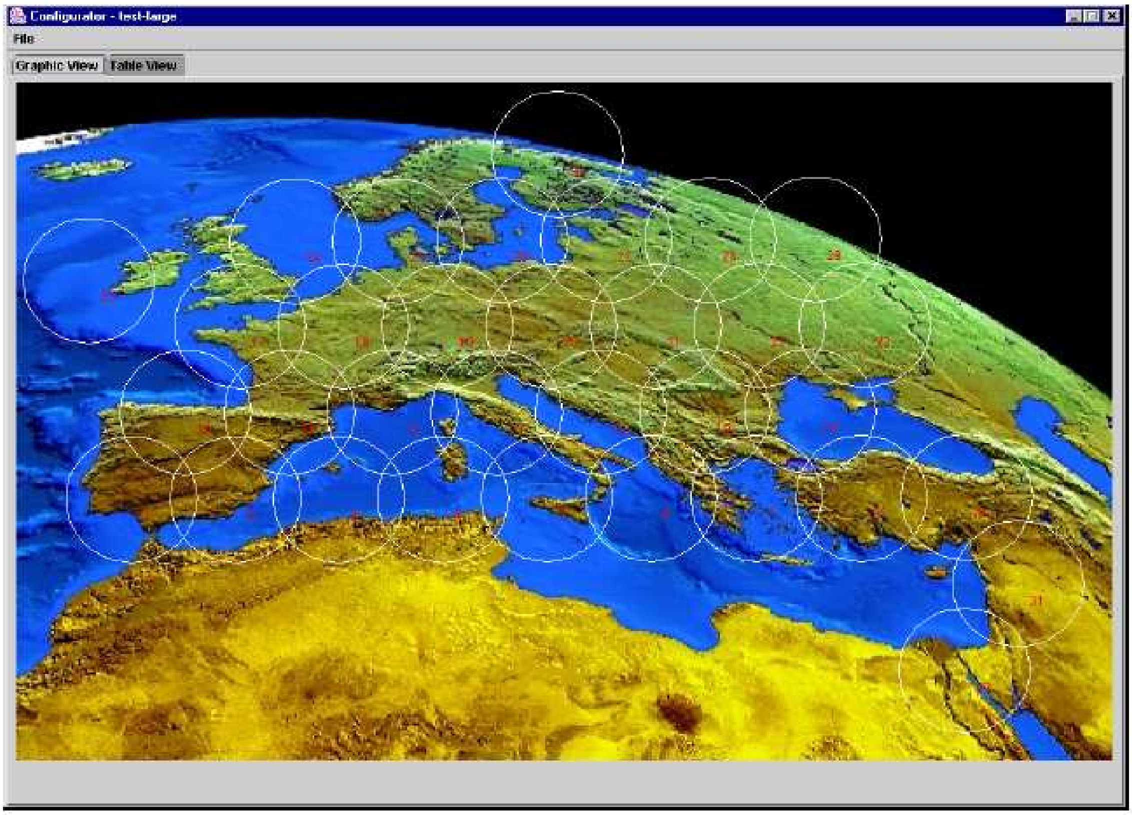


Figure 8.1.: Top level interface of the ESW GUI

8.2.2.1. Euroskyway Phase II: GTS

The goal of the EuroSkyWay project was to design an European multimedia satellite system. DLR was implied in the project and had to realize the GTS (Ground Traffic Simulator) that was an element of the EVA (EuroSkyWay VAlidator) responsible of the ground testing for the system [NGJV02]. The GTS was responsible of generating the traffic equivalent to a loaded system. Its output was at the input of the EPS (EuroSkyWay Payload Swicth) and had to be compliant with the real format of the system at physical level.

During this project, a review was conducted on traffic models and the implementation of some of them was performed. Meanwhile for the design of a graphical interface in order to set the scenarios to be used for traffic generation, the basis of a hierarchical organization was conceptually derived.

Figures 8.1 and 8.2 show two screen-copies of a graphical user interface that was conceived by DLR to configure the traffic generation of the GTS prior to operation. The first picture 8.1 is the main picture, where the transport level configuration is fixed. Each of the circle is a carrier-group. In the pyramid model, each of them corresponds to a node. When the user is clicking on each spotbeam, a window appears for the configuration of the node. Later, in the hierarchy, the window of kind of the one present in 8.2 enables to configure the type of applications used by the Euroskyway user. Clearly, it corresponds to the configuration of the applications used by the subscribers like in the pyramid. These examples show that the pyramid is a good tool to describe the entities responsible of the generation of traffic in the GTS.

As a conclusion, the traffic generation of the GTS was configured (and implemented) following the principles of the aggregated traffic model pyramid. This shows that this model can be used for effective generation of traffic in a dedicated system.

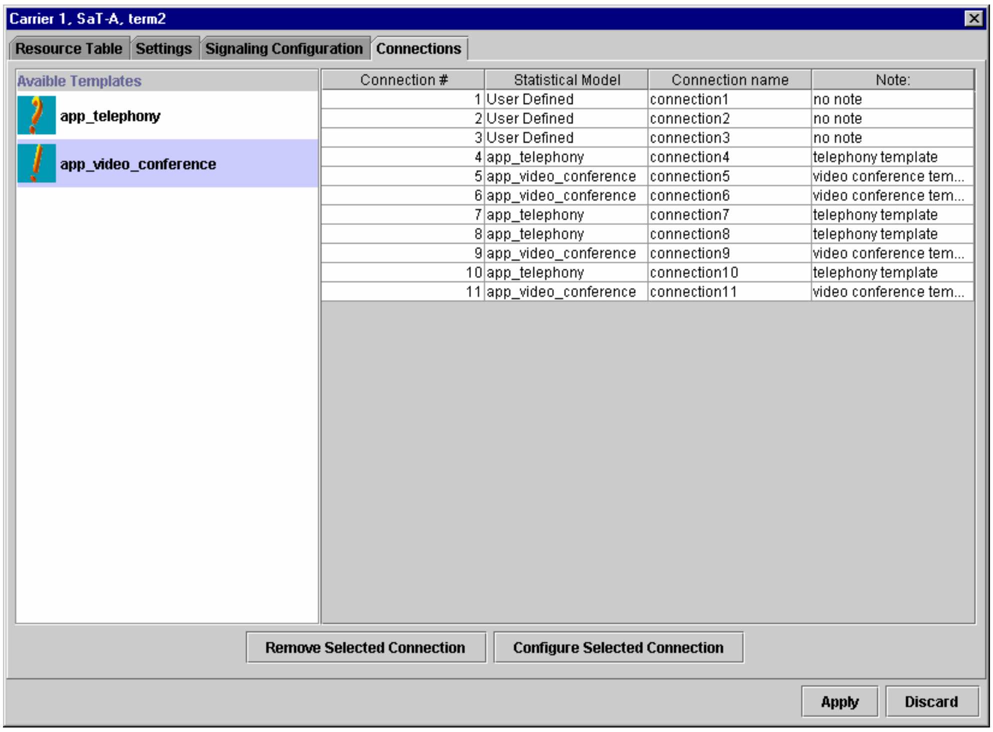


Figure 8.2.: ESW GUI: configuration of the subscriber level

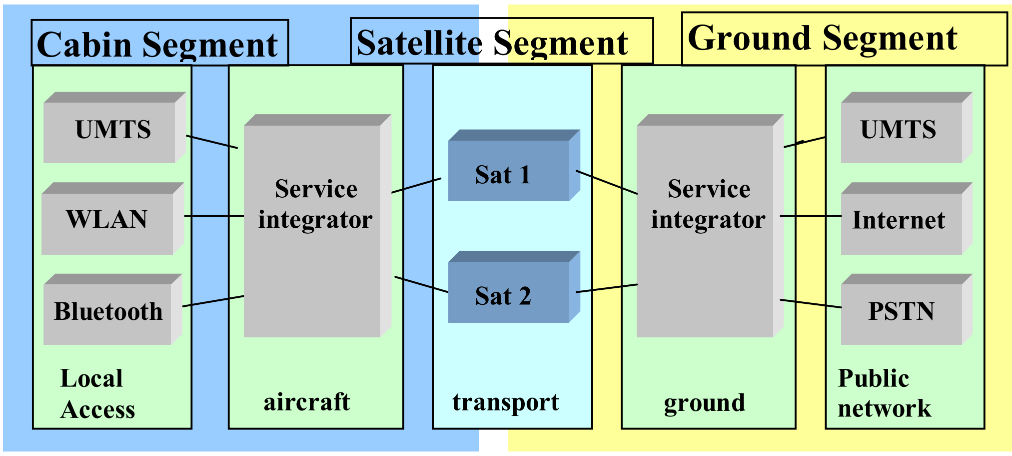


Figure 8.3.: Architectural organization for Wireless Cabin

8.2.2.2. Wireless Cabin

The project has given birth to the scenario that was studied intensively during this study. In the project, I participated to the definition of system deployment for different phases (demonstrator, first commercial system, next generation). A study was also performed to analyze the amount of traffic volume generated by the communication services. A market study based on airliners, aircraft manufacturer and passengers surveys showed that some services had a good expected acceptance.

Figure 8.3 shows the architecture of the Wireless Cabin target system. In this description, different domains have been introduced: the local access domain, the on board service integrator domain, the transport domain, the ground service integrator domain and the public network domain. But it could also be seen with the view of the aggregate traffic pyramid model: the transport domain is the transport level, the service integrator is the node level. The generation of traffic on the transport level from the different aircraft can be viewed as a pyramid. For the consideration of the system with the ground counterparts, the inverted pyramid necessary for performance evaluation appears again, this showing that even system architecture are well described within our pyramid model.

Moreover, a demonstration flight was performed (on board of an Airbus A340-600 on 29/11/2004) where the different services of the WirelessCabin system were shown (GSM calls, Bluetooth application and wireless LAN Internet access). Access was granted to the recordings of traffic during this flight and they allow us to compare our theoretical assumptions with real life data. This data were presented in chapter 3.

8.2.2.3. Other projects

The results presented in this study could be also used in two other studies performed by DLR. The first one is the ESA ACM Modem (European Space Agency Adaptive Coding and Modulation Modem) project where DLR was responsible of the implementation of traffic generator to be integrated within a demonstrator for DVB-RCS satellite with advance coding and modulations. Within this project, the single traffic models that are described here were implemented. This tool was configurable via SMTP. The configuration of the traffic generation was done using a hierarchy quite similar to the one developed within this thesis.

Similarly, in the project ULYSS (Ultra Fast Switching), where DLR was responsible of the realization of the simulator of a satellite switch (the kernel element of the transport level), it was necessary to determine which application were responsible of the longer frames (that are critical for a good switch operation). This analysis and another similar study more focussed on HTTP traffic ([Bou05]) were used to measure this information and gave valuable input for the investigation of switch performance.

8.3. Perspectives for further research

This work provides a large and in-depth overview of the usage of traffic models for the design of multi services satellite system.

The thematic is rich since it covers telecommunication and mathematics. The construction of a theoretic model for the description of traffic sources was performed and then applied to a case of interest. Of course, not all questions have been addressed and there is room for further work. For example os tjere a medd for other type of traffic models ? Also a category of traffic models that showed interesting properties but which could not be integrated in this work is introduced as a potential direction for further work.

8.3.1. Reaching new traffic models objectives

As pointed in the introduction, the number of available traffic models in literature is very high. Because of that, in this thesis, not all models could be evaluated, compared and optimized. This gives a feeling of limitation in the exploration of a quasi-infinite field. But, the objective of traffic modelling is to mimic the behavior of a recorded traffic trace: the mimicry will always be limited because even a perfect fit to the recorded data, no guarantee can be given for the fit to another record. On this point, this work intended to show that the proposed models have good agreements with the original traces (or original models) and that they have the same properties as the original flows.

In order to remain accurate, traffic models will need to fit to the new applications that will appear in the next years. For example, the growth of Digital Subscriber Line (DSL) connections for home users and the birth of file sharing applications has caused a modification of the composition of web traffic. This requires a modification of the terrestrial models used for network design. If these applications are considered to be used also on satellite networks, the BMAP model used in this thesis could still be used to model the traffic because it is modelling the packets sizes (peer traffic is essentially composed of two big categories of traffic: ”mices” for very small packets and ”elephants” for the very big packets), so in order to follow the birth of new services, the models should be adapted within the same framework.

So although the work was limited to a few applications, the methodology exposed here can easily be extended to any other application.

8.3.2. Simplifications for aggregated models

A possibility that was not explored in this thesis is the use of circulant MMPP for the simplification of the models. The spectral density of these models can be easily derived for aggregated streams (the spectrum is additive and has the same shape for every identical component)

In the derivation of the parameters of the c-MMPP(101) from a known spectrum, two steps are required: first the vector of eigenvalues of the process are required λ and then the vector $\mathbf{a}$ of the circulant matrix Q (first row). Difficulties occurred in the implementation of the second step, so that model parameters could not be derived properly (because may be of misunderstanding of the described procedure). The MMPP(3) generator was used with the parameters of cMMPP(3) in order to check the statistical properties. The implementation of a circulant MMPP(n) is straightforward from the MMPP generator; only a dedicated handling of parameters is necessary.

9

Synthèse du travail

Résumé

Cette thèse s'intéresse à la modélisation du trafic de données transitant sur des systèmes de communications par satellite. Les services concernés sont multiples (voix, surf web, courrier électronique, diffusion audio/vidéo...) et le lien satellitaire est caractérisé par une capacité limitée.

Les recherches suivantes ont été menées :

- Définition d'un cadre général hiérarchique permettant la description du trafic pour les systèmes étudiés avec plusieurs niveaux d'abstraction,
- Revue de modèles de trafic et définition de modèles adaptés pour chaque niveau,
- Détermination des paramètres de ces modèles, étude par simulation d'un scénario ,
- Étude des équivalences des modèles agrégés.

Dans le premier chapitre, des notions sont introduites dans le domaine de l'ingénierie de trafic et des systèmes de communication par satellite. Ensuite, le cadre de l'étude est précisé et les problèmes à résoudre sont posés. Puis l'état de l'art est étudié : les résultats essentiels sont les principales études de modélisation du trafic dans les réseaux terrestres (qui demandent une modélisation de la dépendance à long terme), les méthodes matricielles géométriques (qui permettent l'étude théorique de certaines files d'attente).

Dans le second chapitre, un modèle hiérarchique est introduit. Ce modèle est basé sur différents niveaux d'abstraction dénommés applications, souscripteurs, nœuds et transport. Dans ce cadre, des flux de trafic sont définis et leur modification à chaque niveau est décrite par des opérateurs. Ces opérateurs manipulent des paquets dont le format est spécifique à chaque niveau. On définit aussi la comparaison de modèles et les métriques observables dans le système. Finalement, l'architecture d'un simulateur conçu sur cette organisation hiérarchique est définie.

Le chapitre 3 est dédié à l'étude des modèles de trafic. En premier lieu, les modèles les plus appropriés sont présentés théoriquement (on considère notamment les processus de renouvellement et les modèles d'arrivées poissoniens Markoviens MMPP et BMAP). Les modèles des services sont ensuite présentés qui serviront de base pour la détermination de modèles agrégés pour les niveaux supérieurs. Diverses méthodes pour la détermination des modèles sont envisagées

Dans le chapitre 4, un scénario pour les communications aéronautiques est présenté qui servira de cadre pour les études par simulation ultérieures. Pour ce cas, le modèle hiérarchique en pyramide est utilisé et les opérateurs sont décrits de façon précise. Ensuite les principes essentiels de certains modules sont décrits (générateur de trafic, analyseur type file d'attente, analyseur des temps d'arrivée, analyseur de session) synthétiquement.

Les chapitres 5, 6 et 7 présentent les résultats obtenus respectivement aux niveaux souscripteurs, nœuds et transport. Au niveau souscripteur, les propriétés des applications sont vérifiées et des modèles agrégés sont dérivés. Les résultats sont étendus au niveau nœud. Les équivalences entre modèles agrégés et précis sont démontrés à différents points de vue (volume de donnée, distribution des files d'attentes, ...). De même, au niveau transport, des estimations de charges sont effectués et généralisent les résultats précédents. Finalement, une estimation de performance est effectuée à l'aide des modèles développés.

Le dernier chapitre résume les résultats obtenus, évalue les futurs besoins en terme de modélisation du trafic et envisage divers prolongements de cette thèse (par exemple l'utilisation de modèles circulants pour la simplification des modèles agrégés ou les extensions probabilistes (basées sur la théorie du *network calculus*)).

9.1. Introduction

Ce chapitre d'introduction fixe l'objet du travail, les définitions de base, la formulation de la problématique traitée, la méthode de résolution choisie et l'organisation du travail.

9.1.1. Les concepts d'analyse du trafic de données dans les réseaux de communication par satellite

9.1.1.1. L'ingénierie du trafic

L'ingénierie du trafic est une discipline répondant à trois objectifs principaux : la conception, l'analyse et l'optimisation des réseaux de communications. Pour arriver aux résultats, trois actions sont nécessaires : la mesure du trafic, la caractérisation du trafic et la modélisation du trafic. Les deux derniers thèmes sont abordés dans ce travail.

9.1.1.2. Systèmes satellites

Le développement des systèmes par satellites a débuté en 1960 pour la couverture de larges zones, l'avantage des systèmes par satellite étant leur adaptation à la diffusion d'information. De nombreux systèmes ont été envisagés pour fournir des services multimédias et les protocoles possibles ont été standardisés. Les principales caractéristiques de tels systèmes sont résumées dans ce chapitre.

9.1.2. Formulation du problème et cadre du travail

9.1.2.1. Scénarios étudiés

Ce travail vise à étudier les échanges de données dans un système de communication entre deux applications paires. En particulier, ce système peut être un satellite de télécommunication permettant diverses applications. On parle alors de satellite multiservices.

9.1.2.2. Objectifs

Le thème du travail est l'élaboration d'un modèle pour la description hiérarchique de la génération de trafic au sein d'un réseau satellite. Une modélisation du trafic via un cadre hiérarchique à travers différents niveaux d'abstraction a été conduite. Des études par simulations ont été menées pour démontrer et valider l'intérêt de la modélisation hiérarchique. Les objectifs du travail sont :

- i) Définition d'un cadre général hiérarchique permettant la description du trafic pour les systèmes étudiés avec plusieurs niveaux d'abstraction,
-ii) Revue de modèles de trafic et définition de modèles adaptés pour chaque niveau,
iii) Détermination des paramètres de ces modèles, étude par simulation d'un scénario,

- iv) Étude des équivalences des modèles agrégés. Extraction d'indicateurs de performance du réseau pour ces modèles.

9.1.2.3. Méthode de résolution

L'approche choisie est basée sur un modèle hiérarchique en structure pyramidale qui sera présenté au chapitre 2 (ou en 9.2 dans ce résumé) et précisé pour un cas pratique. Une partie de l'étude sera réalisée par simulation numérique pour ce qui concerne la mise en évidence des équivalences entre les modèles.

9.1.3. État de l'art

Une fois les objectifs du travail fixés, les résultats de travaux de recherche utiles pour ce travail sont présentées. En particulier on revoit tout d'abord des résultats sur le télétraffic et ensuite sur le trafic de données transmises par satellite :

9.1.3.1. État de l'art des domaines de l'analyse du trafic de données et les réseaux de communication par satellite

Systèmes satellites Le développement des systèmes par satellites a débuté en suivant une idée d'Arthur C. Clarke. Il a montré que des relais extraterrestres pouvait être utilisés pour fournir des services de communications à de vastes régions. Un état des lieux des systèmes en cours de développement a été conduit.

9.1.3.2. État de l'art pour le télétrafic

Une analyse de la littérature concernant la modélisation du trafic dans un réseau de télécommunication a été menée. Les principaux résultats sont les suivants :

Résultats pour les réseaux terrestres
Les principales études traitant du télétrafic concernent les réseaux terrestres. Le premier concept répandu est la notion d'auto-similarité détecté sur des échantillons de trafic web. En effet, de nombreuses études on montré que le profil du trafic web avait un caractère fractal que l'on modélise par des modèles auto-similaires. L'utilisation de modèles poissoniens est mal adaptée car elle conduirait à un sous dimensionnement de ces liens.

Théorie des files d'attentes
Un grand nombre de résultats sont issus de la théorie des files d'attentes. En cherchant à étendre les formules d'Erlang pour le dimensionnement des liens basés sur la voix, les propriétés statistiques des files d'attente ont été calculées pour des modèles plus complexes permettant de décrire des applications plus évoluées. Les principales avancées dans ce champ sont rapportées.

Revue des modèles de trafic
Ce paragraphe commence par lister un certain nombre de modèles de trafic classés par la catégorie à laquelle ils appartiennent. Ensuite les différences entre une approche aléatoire ou déterministe pour la modélisation sont explicitées afin de justifier l'emploi de modèles aléatoires. Finalement, le principe de modèle multiservice est introduit avec les paramètres adaptés pour représenter différentes applications.

9.1.3.3. Modélisation du trafic pour le satellite

Des protocoles spécifiques ont été développés pour adapter développés pour fonctionner dans un système satellite. Les protocoles ATM et DVB-S sont par exemple dédiés aux applications multimédia par satellite. Dans le cadre du développement de ces protocoles, les performances du système ont été étudiées. Différentes architectures peuvent être envisagées, mais la capacité limitée du lien implique une limitation des performances possibles. Pour décrire la sporadicité des applications, certains concepts comme la bande passante équivalente ont été proposés. Finalement, les services considérés sont les applications qui devraient être les plus populaires sur les réseaux à base d'IP. On considère donc les applications voix, d'envoi d'email, de visualisation vidéo, et le web browsing.

9.1.4. Organisation du travail

Le plan du travail est le suivant.

- Le chapitre 2 (résumé en 9.2) décrit notre modélisation hiérarchique du trafic : la pyramide de trafic agrégé. L'ensemble des définitions servant de cadre à l'étude sont précisées ainsi que l'organisation structurale du modèle.
- Le chapitre 3 (résumé en 9.3) décrit les bases de modélisation du trafic de données. Les définitions du chapitre 2 donnant un cadre à la modélisation du trafic, les modèles les plus pertinents pour notre étude sont expliqués. En particulier, les modèles de la famille des MAP (Processus à arrivées Markoviennes) sont décrits, pour présenter les modèles On/Off , MMPP et BMAP qui sont très utiles pour modéliser différentes applications. Ensuite, les méthodes de détermination des paramètres convenant à une modélisation sont décrites. Comme il n'existe pas de méthode universelle, on précise pour chaque cas la méthode adaptée pour chacun des modèles.
- Dans le chapitre 4, on expose le système étudié pour lequel on modélise les trafics échangés et on précise la nature des opérateurs. Le lien entre les deux précédents chapitres théoriques se fait au travers du système que l'on étudiera par la suite avec un simulateur. Le scénario de cette étude est donc présenté en utilisant le formalisme exposé dans le chapitre 2.
- Le chapitre 5 présente les résultats d'une étude d'agrégation de sources de type voix dans les différents niveaux de la pyramide. Cet exemple particulier permet de construire des agrégations à différent niveaux et, en particulier, d'étudier les équivalences entre les différents niveaux. La généralisation s'effectuera au cours des deux prochains chapitres.
- Les chapitres 6 et 7 présentent les résultats obtenus tant au niveau nœud et transport. Les premiers gains sont illustrés au niveau nœud où l'agrégation est plus importante qu'au niveau souscripteur. Une généralisation est effectuée au niveau transport. Les modifications des flux de trafic sont modélisés en tenant compte des différents niveaux de la pyramide.
- Dans le chapitre 7, une étude portant sur le dimensionnement de systèmes est réalisée en utilisant les modèles précédemment développés. Dans ce cadre, des résultats numériques sont obtenus par exemple pour estimer les métriques de performance d'un futur système. La validation des équivalences de modèles est conduite.
- Le chapitre 8 résume les principales réalisations de ce travail et conclut en indiquant les directions pour de futures recherches.

9.2. La pyramide de trafic agrégé (chapitre 2)

Ce chapitre introduit la notion de pyramide de trafic agrégé qui sert de cadre à l'ensemble de l'étude. Cette pyramide est définie sur plusieurs niveaux qu'il s'agit de caractériser. Ensuite, sur cette base, on développe un cadre théorique pour décrire la génération du trafic ainsi que la mesure des performances permettant de comparer les caractéristiques des flux de trafic.

9.2.1. Principe de modélisation hiérarchisée

On modélise les sources d'informations d'un système de télécommunications de manière hiérarchique. Le principe d'organisation est celui d'un modèle hiérarchisé bâti de la façon suivante :

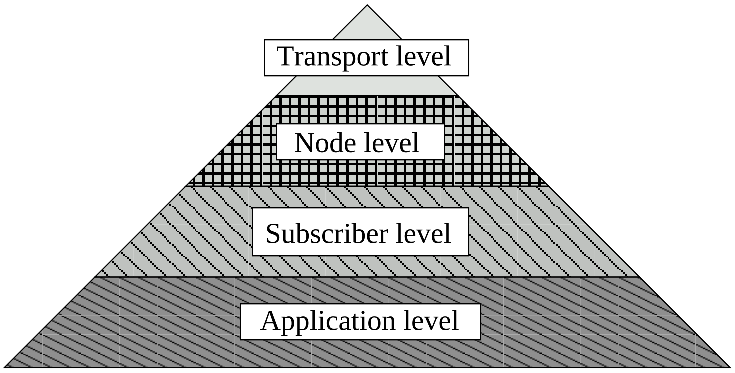


FIGURE 9.1.: Le concept de pyramide de trafic agrégé

i) Chaque souscripteur est supposé utiliser plusieurs applications. Ces applications sont la base de la génération de trafic.
ii) Chaque nœud est constitué d'un certain nombre de souscripteurs.
iii) Le niveau de transport est constitué d'un ensemble de nœuds.

Cette représentation définit ainsi la pyramide de trafic agrégé. La Figure 9.1 illustre le concept avec les différents niveaux. Un scénario qu'on désire observer au niveau transport est donc composé d'un certain nombre d'applications, de souscripteurs et de nœuds.

Même si le modèle en pyramide est défini de manière générale, nous l'appliquerons à des systèmes utilisant le satellite comme composant du niveau transport. Le cadre de l'étude est précisé au paragraphe 9.4.

9.2.1.1. Exemple d'utilisation du modèle

Cet exemple illustre le modèle en pyramide. Dans le cas de l'accès d'un utilisateur consultant des pages d'Internet par l'intermédiaire d'un système satellite, l'application est le logiciel de visualisation des données (web browser par exemple). Symétriquement, l'application fournissant ces données situés à l'autre bout de la chaîne de transmission est aussi constituante du niveau applicatif. Le niveau souscripteur est la machine de l'utilisateur et le niveau nœud est constitué du terminal satellite permettant l'accès au service. C'est un scénario typique pour un système de communication et le modèle en pyramide s'y applique bien.

9.2.1.2. Définition des besoins du modèle

Modéliser le trafic au sein d'un système ne peut se faire qu'au travers d'une certaine abstraction des éléments constitutifs du système. Il faut donc définir le niveau d'abstraction avec lequel on décrit le système, afin de définir les exigences auxquelles les modèles devront répondre.

Par ailleurs, la comparaison entre deux modèles de trafic doit être possible. L'évaluation des caractéristiques de différents modèles doit être possible même si certaines simplifications ont été effectuées. En particulier le volume total doit être conservé, l'espace disponible pour d'autres applications doit être décrit et les performances comme le retard ou des critères de qualités de service doivent rester mesurables. Finalement, les modèles doivent aussi permettre d'obtenir des mesures caractéristiques des performances du système pour estimer par exemple : le volume des paquets échangés, la charge disponible pour un service supplémentaire, certains critères de performance relatifs à la gestion de la qualité de service.

9.2.1.3. Le modèle des paquets : un organisation par niveau

Les données échangées au sein du système (paquets) sont associées à un niveau de la pyramide. De manière générale, les données échangées sont décrites par des formats génériques que l'on précise dans le cadre hiérarchique. Les paquets pour le transfert de données sont décrits par leur taille qui est propre au niveau considéré. A cela s'ajoute la décomposition des paquets en un entête et une charge utile (qui correspond à la portion du paquet correspondant au transfert effectif d'information). Les opérations de transfert de niveaux sont ensuite décrites. Pour faire transiter les paquets dans la hiérarchie, les opérations de fragmentation, d'agrégation (regroupement) et transmission inter-niveaux se produisant doivent être décrites et leur action est formalisée au moyen des définitions (2.1.4, 2.1.5 et 2.1.6).

Ces opérateurs doivent en particulier permettre de décrire les opérations de stockage temporaire (buffering) et d'agrégation-concentration entre diverses sources et une destination commune. Un exemple est donné pour les types de paquets qui sont manipulés lors de la transmission de paquets IP dans un système satellite.

9.2.2. L'architecture en pyramide : l'agrégation par niveaux

9.2.2.1. Niveaux et entités considérées

Tout d'abord, le vocabulaire suivant est défini : réseau (ensemble d'entités et de liens), lien (médium capable de transmettre des données), entité (élément du réseau), niveau (caractéristique de la position de l'entité dans la pyramide, variant de 1 à 4), le niveau applicatif (où se trouve la source et la destination des données), le niveau du souscripteur (où se trouve l'équipement communiquant, le niveau du nœud (terminal d'accès au système final) et le niveau transport (système central que l'on cherche à observer).

9.2.2.2. Modèles de trafic

La notion de modèle de trafic est définie : en distinguant le modèle simple (ou primaire) qui est l'origine d'un flux de trafic et un modèle composé résultant de l'étagement et de la composition de diverses sources.

Modèles simples : Pour décrire le trafic généré par une source simple dans un niveau, les définitions de l'unité abstraite (composé de deux variables aléatoires indiquant l'instant de passage d'un paquet de manière relative ou absolue et sa taille) et du flux d'unités abstraites (qui est l'ensemble des unités abstraites vues en un point du système) sont introduites. Le flux d'unités abstraites représente ce qu'on désire obtenir en sortie d'un modèle de trafic. Des opérateurs génériques sont ensuite introduits pour décrire les modifications qui affectent un flux entre l'entrée et la sortie (on précisera plus tard trois d'entre eux ainsi que leur rôle dans le cas étudié en simulation numérique). Finalement le modèle indépendant qui représente la source simple au sein d'un niveau est introduit.

Opération d'agrégation : L'agrégation de flux est une opération importante qui nécessite une description détaillée pour modéliser les différents effets qui interviennent ainsi que pour être à même de reproduire ses effets sur un flux de trafic synthétique. Elle est tout d'abord explicitée pour l'ajout de deux flux de trafic et se généralise par la suite. Cette opération est divisée en trois étapes qui sont l'addition de flux, la conversion de niveau et la mise en conformation avec le niveau supérieur. Chacune de ces opérations est décrite par un opérateur. Ces opérateurs permettent de passer du flux d'un niveau inférieur à un niveau supérieur. L'opération d'addition (premier opérateur élémentaire) consiste à obtenir le flux de sortie d'un certain nombre de flux d'entrée qui se superposent. Deux modes sont considérés : l'addition absolue et l'addition en salve. Pour chacun des cas, les règles de construction des flux sont explicitées. Dans le cas en salve, un seul paquet dans la salve est considéré dont la taille est calculée en sommant les tailles de chacun des paquets en entrée dans la salve. Ce paquet sera émis à un instant plausible durant la durée de la salve. Si la capacité du lien est insuffisante, on fera appel à l'opérateur de conformation qui sera décrit ultérieurement. Dans le cas absolu, les instants d'émission de chacun des paquets des flux d'entrée sont conservés avec leurs tailles initiales. La conversion de niveau (second

opérateur élémentaire) consiste à convertir le format des paquets reçus de l'opérateur d'addition au format en vigueur au niveau supérieur. La conformation de niveau (troisième opérateur élémentaire) vise à modéliser les opérations nécessaires pour respecter les contraintes imposées par le niveau final.
En pratique, les opérateurs permettent de décrire toutes les caractéristiques du flux de sortie, comme dans le cas de deux flux poissoniens, dont la loi superposée n'est plus poissonienne, car les caractéristiques du flux de sortie ont été modifiées par l'opération d'agrégation.

Modèles multi-niveaux : L'architecture des modèles nécessitant d'incorporer plusieurs niveaux, le modèle dépendant est introduit à cet effet. Il permettra plus tard de donner un cadre pour la comparaison de modèles utilisés à des niveaux différents. Cette comparaison s'effectuera en utilisant les critères de performance définis dans la section suivante dédiée à l'évaluation des performances.

9.2.2.3. Relations entre le modèle OSI et les niveaux du modèle en pyramide

Les relations entre le modèle standard ISO 8478/OSI et le modèle en pyramide sont évoquées afin de mettre en lumière leur interactions. Une correspondance entre les 4 niveaux de notre modèle et les sept couches du modèle OSI est établie.

9.2.3. Évaluation des performances

L'hypothèse est faite d'un système en fonctionnement normal (c'est à dire que les phases d'établissement de connections ne sont pas considérées). Pour un tel système, des mesures de performances sont définies et leur rôle est également précisé.

9.2.3.1. Principes pour la comparaison des flux et l'évaluation des mesures

Comparaison des flux d'unités abstraites : Les caractéristiques qu'un comparateur de trafic doit posséder pour distinguer numériquement deux flux sont définies. La réponse fournie permet de juger dans quelle mesure un flux équivalent est semblable ou éloigné du flux initial. A l'aide du comparateur, la validité de modèles équivalents pourra être évaluée.

Définition et propriétés des mesures : Les différents types de mesures sont inventoriées. Celles qui sont instantanées, séquentielles, en moyenne temporelle, en distribution statistique, basées sur un extremum sont distinguées. Ces mesures se distinguent ensuite en fonction des points où elles sont évaluées : elles peuvent être sont évaluées en un point unique, ou entre deux points, ou entre deux sorties de générateurs (éventuellement virtuels pour la comparaison dans deux scénarios), etc. Finalement, des exemples de mesures telles qu'elles peuvent être recueillies dans un système réel sont données.

Mesures issues de la comparaison des flux : Dans le cas où deux flux sont observés en deux points distincts, ces mesures indiquent si les flux sont similaires ou peu similaires. Le comparateur intégral (basé sur un calcul d'intégrale) qui généralise le comparateur de différence est introduit.

9.2.3.2. Mesures disponibles dans le modèle en pyramide

Mesures dérivées d'un unique flux d'unités abstraites : Ces mesures sont évaluées en un point donné unique. Cela correspond à une sonde mesurant une grandeur caractéristique du trafic passant en ce point. On décrit en particulier l'évaluation du volume de données (mesure instantanée), du volume de transfert moyen (mesure instantanée), du taux d'arrivée (mesure de moyenne), du taux de transfert net. Par la théorie des files d'attente on peut aussi calculer la répartition théorique d'une série de mesures (comme par exemple la taille de la file d'attente à un instant quelconque, sa taille à l'instant de l'arrivée d'un paquet ou à la fin du service d'un paquet etc). Son évaluation (par la mesure) renseigne donc sur l'état du système.

Mesures orientées paquets : Les types de relations existants entre paquets originaires et destinations sont caractérisées (selon qu'ils sont considérés avant leur transmission ou après leur réception). Grâce à ces relations, on peut définir des mesures supplémentaires comme les pertes de paquets et la mesure du retard subi par un paquet. Le taux de perte est une grandeur couramment utilisée pour fixer des objectifs de qualité de service que le réseau doit maintenir. Certaines de ces mesures demandent à ce que source et réception soient considérées au sein d'un même niveau (car sinon l'évaluation est imprécise ou n'a pas de valeur significative).

9.2.3.3. Mesures dédiées à l'étude des performances

Recommandations de l'ITU : Des documents de référence indiquent les paramètres significatifs pour l'évaluation des performances de réseaux de télécommunication. Les définitions essentielles sont indiquées comme le taux de cellules insérées incorrectement. Avec le modèle en pyramides des mesures de performance sont obtenues qui sont similaires à celles des références normatives.

Lien avec les concepts de qualité de service : La corrélation entre l'approche issue du modèle en pyramide et les définitions traditionnelles de la qualité de service est démontrée.

9.2.3.4. Évaluation des performances aux différents niveaux de la pyramide

L'évaluation des mesures exposées précédemment est mise en œuvre dans cette section pour des cas courants.

Au sein d'un niveau : En particulier, pour deux flux, le délai de transmission pour différents types d'applications est évaluée à l'occasion de la combinaison de deux flux.

Au travers plusieurs niveaux : Les difficultés d'une telle approche et les erreurs qui peuvent être introduites sont expliquées. La pyramide inversée, nécessaire pour les fonctions de réception, est également décrite. Elle complète si nécessaire la pyramide pour l'évaluation des performances.

Dans cette section, le modèle de la pyramide de trafic agrégé est décrit. Cette organisation permet d'agencer les modèles de trafic hiérarchiquement et de décrire l'évaluation des performances d'un système de télécommunication.

9.3. Modèles de trafic organisés en pyramide (chapitre 3)

9.3.1. Modèles de trafic pour les flux élémentaires

9.3.1.1. Modèles mathématiques pour les générateurs indépendants

Le générateur indépendant, ainsi que les définitions nécessaires à la définition d'un modèle de trafic ont été données au chapitre 2 (section 9.2). Cela ouvre le champ à la description des paquets dans le temps, puisque le modèle de trafic permet la description des paquets de données (unités abstraites) en un point donné.

Les modèles de trafic peuvent avoir des natures différentes puisqu'ils peuvent être déterministes ou probabiliste (par exemple les variables que sont le temps entre les arrivées et la taille des paquets peuvent être considérées comme fixes ou aléatoires), avoir une référence cadencée ou une référence absolue (en fonction de la référence temporelle choisie pour la description des paquets dans la trame).

Les modèles indépendants suivants sont présentés :

- les processus ponctuels et de renouvellement : Ces modèles simples (et aux capacités de description limités) permettent de comprendre ensuite des modèles plus complexes construits sur leur base : un certain nombre de leurs propriétés sont rappelées car elles faciliteront l'exposition des autres modèles. En particulier, la définition des processus ponctuels et de renouvellement, du temps de récurrence progressif ou rétrograde est rappelée. La fonction de renouvellement (et sa transformée de Laplace) qui se calcule aisément pour certains types de processus standard est également présentée. Le comportement asymptotique peut en être déduit. En outre, la superposition des tels processus est considérée pour montrer que le comportement asymptotique est poissonien. Finalement, certaines méthodologies liées à l'observation de tels processus sont présentées.

- le processus de renouvellement PH : Ce processus qui généralise le cas précédent est introduit. La notation matricielle facilite certains calculs. La matrice de phase T et le vecteur α décrivent complètement la distribution des temps d'inter-arrivées et de la fonction de renouvellement. La combinaison de deux modèles se calcule aussi aisément par des opérations matricielles.

- le processus de renouvellement Markovien : Ce processus, qui est une nouvelle généralisation du cas précédent, est également introduit car il offre l'avantage de pouvoir calculer la superposition exacte de flux. En revanche, le nombre d'états nécessaire pour décrire cette superposition explose rapidement, ce qui rend son utilisation difficile.

9.3.1.2. Le Processus markovien d'arrivées par salves (BMAP)

Ce processus permet des descriptions très générales des flux de paquets. En effet, il repose sur un processus markovien à plusieurs états (version en temps continu d'une chaine de Markov discrète) pouvant décrire différents *régimes* de génération auquel on adjoint une distribution de taille de paquets. Il repose donc sur une matrice décrivant le processus de Markov et sur un jeu de paramètres décrivant les différentes tailles de paquets. En outre les propriétés des files d'attentes pour ce générateur sont calculables analytiquement.

9.3.1.3. Le Processus de Poisson Markov modulé (MMPP)

Le MMPP est un cas particulier de BMAP qui mérite une analyse à part entière. C'est une simplification puisqu'une seule taille de paquet est considérée et que la matrice du processus de Markov sous-jacent est décrite par une matrice stochastique caractérisant le temps de séjour dans chacun des états du processus et un vecteur décrivant le taux d'arrivée dans chacun des états. Beaucoup de résultats analytiques exposés pour le BMAP sont simplifiés pour le MMPP et certains résultats supplémentaires sont obtenus. Certains se simplifient encore davantage lorsqu'on considère un MMPP à deux états appelé MMPP(2) qui n'a que 4 paramètres.

9.3.1.4. Le Processus de Poisson Markov modulé circulant

La catégorie des MMPP circulants est une classe particulière pour laquelle la structure de la matrice de saut du processus de Markov qui module le processus a une forme spécifique dite circulante. Un certain nombre de simplifications sont alors possibles. En particulier, l'étude du spectre des processus peut être réalisée aisément. La forme particulière de la fonction d'auto corrélation et de densité de puissance d'un tel processus est indiquée. Toutes les opérations d'équivalence peuvent ensuite se faire sur la base des fonctions approchant les spectres.

9.3.2. Générateurs considérés dans notre étude du modèle en pyramide

Après la présentation théorique des modèles, les modèles qui seront utilisés dans le reste de l'étude sont décrits plus en détails en incluant les paramètres des modèles utilisés dans cette étude.

9.3.2.1. Niveau application

Les modèles utilisés pour les applications sont les suivants :

- le modèle voix on/off : Il s'agit d'un modèle à deux états on et off correspondant à une phase active et une phase inactive. Durant la phase active, les paquets sont générés de manière constante. Durant la phase inactive, il n'y a pas de génération de paquet.

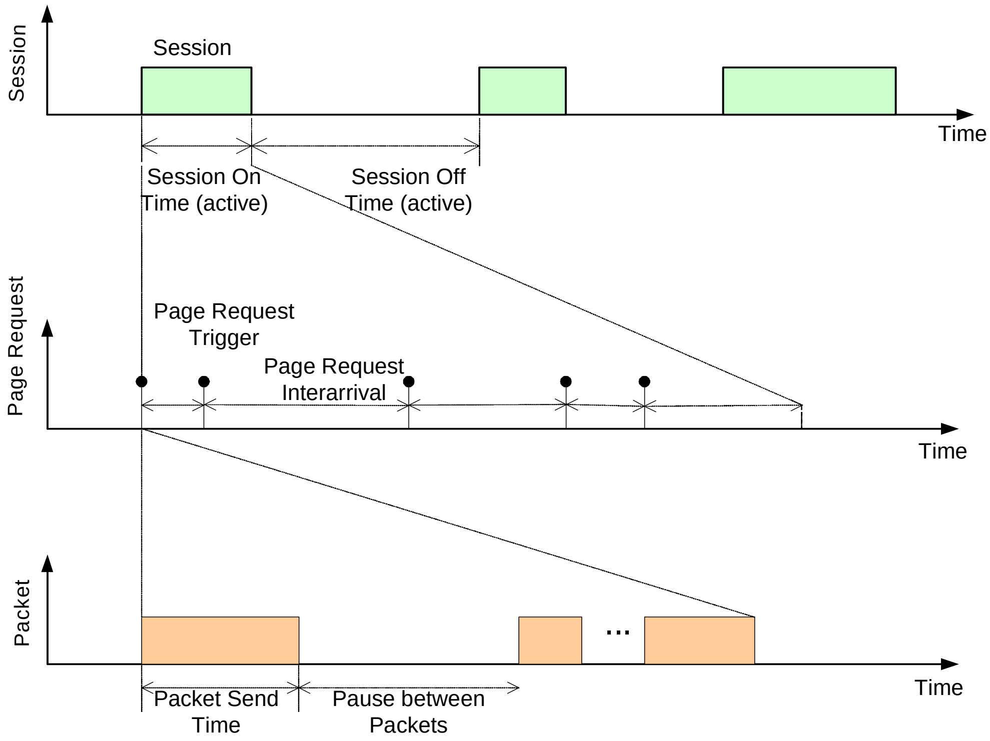


FIGURE 9.2.: Modèle ETSI pour les sessions web

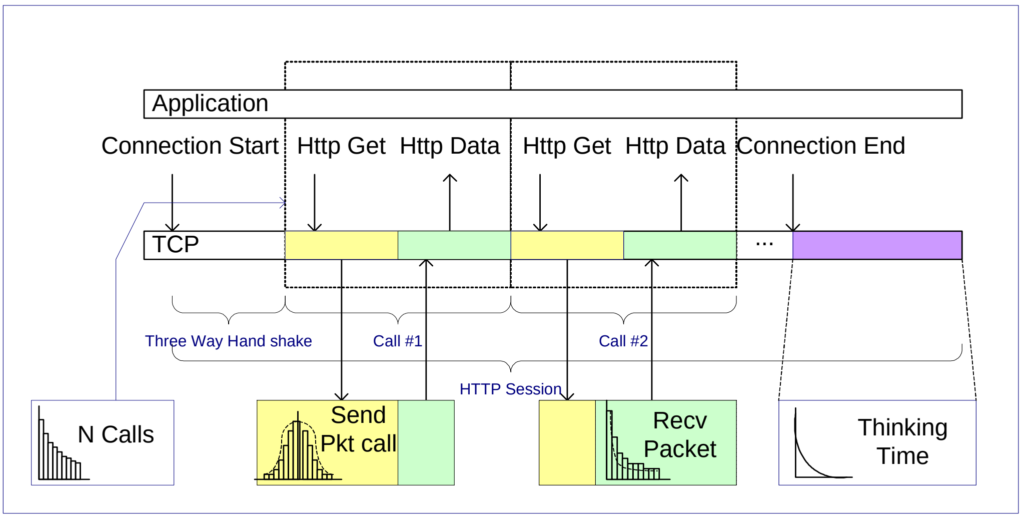


FIGURE 9.3.: le modèle web du DLR

- le modèle vidéo : Ce modèle a la propriété d'être auto-similaire. Ce type de trafic peut avoir des conséquences pénalisantes pour le dimensionnement puisque de gros paquets peuvent saturer les liens. En effet, il repose sur une taille de paquets décrits par une distribution à queue lourde (heavy tailed) issu d'une distribution Gamma-Pareto. Les échantillons initiaux sont issus d'une séquence auto-similaire obtenue à partir de l'algorithme énoncé (algorithme de Hosking). La distribution des tailles de paquet peut être adaptée pour correspondre à un film particulier. Ce générateur, utilisé dans le cadre de [FM98], a été utilisé pour les résultats énoncés au paragraphe 6.1.

- le modèle web : Une première modélisation du trafic web est issue du modèle de l'ETSI [ETS98].

La liste des paramètres qui caractérisent le modèle est établie. Ils servent à décrire la durée et le nombre d'événements se produisant durant la visualisation d'une page web. Ces paramètres comprennent par exemple le nombre de paquets contenus dans un datagramme ou le nombre de requêtes par pages. Des valeurs typiques sont suggérées pour ces paramètres. La Figure 9.2 précise l'architecture du modèle. Le DLR a aussi dérivé son propre modèle (ou une simplification) où les paramètres sont basés sur des mesures comme décrit dans [BG05b]. Ses principes sont exposés dans la Figure 9.3. Une seconde modélisation reposant sur les modèles BMAP est envisagée.

9.3.2.2. Niveaux Souscripteur et Nodal

Il n'existe pas de modèle spécifique pour ce type d'utilisateurs regroupant plusieurs classes qui soit couramment utilisé. La proposition est faite d'employer le BMAP pour du trafic hétérogène (issu de différentes applications) et le MMPP. Le résultat de l'agrégation des flux de trafic issus des niveaux inférieurs sera également considéré et consistera en un modèle composite issus du trafic des niveaux inférieurs.

9.3.2.3. Niveau Transport

Pour élargir la gamme des modèles présents à ce niveau (issus de différents types d'agrégation et de combinaison des autres modèles), un modèle simple a été considéré pour décrire le trafic d'une façon très macroscopique. Dans ce cas, le modèle utilisé repose sur une distribution gaussienne dont les éléments essentiels sont rappelés et dont l'utilisation est très répandue. Les méthodes de simulation de la distribution sont également décrites.

9.3.3. Détermination des paramètres des modèles agrégés

9.3.3.1. Détermination des paramètres du MMPP(2) basé sur les statistiques de comptage

Cette méthode vise à faire correspondre les statistiques de d'ordre 1 2 et 3 (moyenne, variance, etc) du processus agrégé avec celui du processus en entrée. Les équations sont indiquées qui permettent de passer des paramètres mesurés sur la fonction de comptage aux paramètres du modèle. On arrive au final à un jeu de paramètres caractérisant le processus agrégé. Les paramètres obtenus sont cependant très dépendants de la manière dont la fonction de comptage a été obtenue (taille des intervalles élémentaires, durée de l'évaluation). Pour obtenir plus de stabilité, d'autres approches ont été envisagées.

9.3.3.2. Détermination des paramètres du MMPP(2) basé sur les statistiques des files d'attentes

Cette méthode repose plutôt sur les propriétés des files d'attentes associées. On fait se correspondre les transformées de Laplace des processus agrégés. Par une méthode itérative, liée au calcul de la matrice G (caractérisant les sauts du processus d'attente en dehors des instants d'arrivées), on obtient les paramètres du processus agrégé. Cette méthode fonctionne correctement si l'on peut donner un caractère asymptotique à la distribution de la taille de la file d'attente. Pour cela il faut obtenir des tailles d'attente de l'ordre de 30 40 paquets en choisissant un taux de service adéquat. Il faut donc éventuellement modifier la valeur du temps moyen de service par paquet utilisée par le module de mesure.

9.3.3.3. Détermination des paramètres pour les sources vidéo

Dans le cas des modèles vidéo, il faut déterminer les paramètres de la distribution des paquets modélisée par une distribution Gamma Pareto. Pour cela, on détermine les paramètres d'une distribution de Gamma

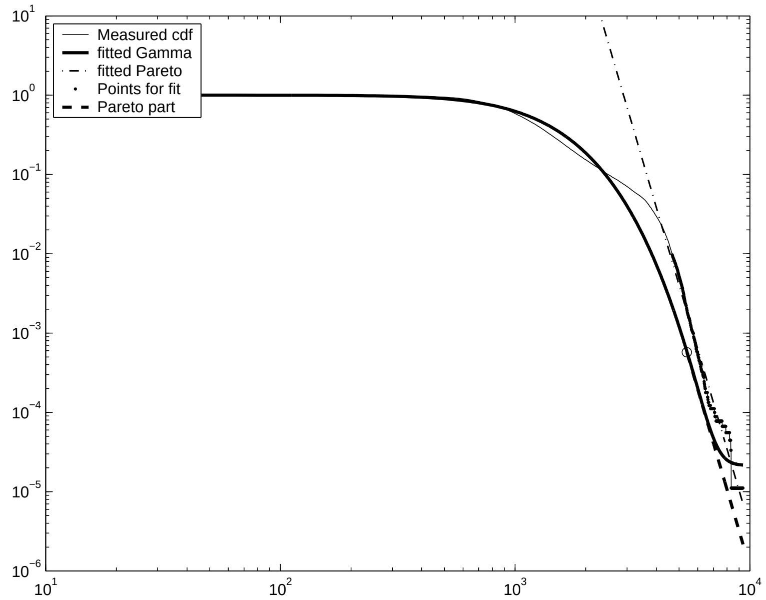


FIGURE 9.4.: Distribution Gamma Pareto adaptée à la distribution de la taille des paquets (modèle vidéo)

représentant la distribution des tailles par les statistiques d'ordre 1 et 2, ensuite on ajoute une partie de Pareto ayant la même pente que la distribution asymptotique. La Figure 9.4 montre cette procédure.

9.3.3.4. Méthode EM

Principes. La méthode EM est basée sur la maximisation d'une fonction de vraisemblance. La fonction de vraisemblance est issue de la distribution de densité de probabilité objectif dans laquelle les paramètres inconnus du modèle sont inclus. Un exemple de la méthode EM est donné où une distribution gaussienne ayant le maximum de vraisemblance pour un nuage de point est recherchée. Les inconnues sont les paramètres du modèle objectif et se composent de la moyenne et de la variance de la distribution gaussienne. Dans ce cas, la solution maximisant la vraisemblance est la gaussienne de moyenne la moyenne des points du nuage et pour variance la variance des points du nuage. Quand l'évaluation est moins simple, on doit faire appel à une maximisation itérative. La méthode EM estime la fonction de vraisemblance quand certaines variables sont inaccessibles (comme les instants de saut du processus de Markov d'un modèle MMPP). On calcule donc une fonction de vraisemblance L_c complète à partir des données accessibles. C'est la phase E (Expectation). On cherche ensuite à obtenir le maximum de cette fonction, lors de la phase M (Maximisation). La fonction de vraisemblance étant partielle on réitère la méthode jusqu'à atteindre un maximum. Dans le cas d'un modèle simple à deux états, on peut mettre à jour facilement les paramètres de modèle. Par itération successives, un maximum est atteint progressivement (cf Figure 3.9 pour le lecteur intéressé par la visualisation du chemin de convergence).

Principe pour le MMPP et le BMAP. Dans le cas du MMPP, les paramètres du modèle maximisant la fonction de maximum de vraisemblance peuvent être déterminés. Les observables sont les arrivées de paquets. Les équations de mises à jours du modèle sont indiquées ainsi que la forme des distributions de probabilité pour la fonction de vraisemblance complète. Les données manquantes sont les instants de saut du processus de Markov sous-jacent. Le cas du BMAP est semblable au cas du MMPP. Chaque observable est également une arrivée de paquet. La catégorie de taille est connue par la taille indiquée du paquet et ne fait pas l'objet d'une conjecture. La procédure est donc identique.

Procédure d'initialisation. En raison du caractère itératif de la méthode EM, la procédure d'initialisation est une étape cruciale, puisqu'elle conditionne la durée de la recherche du maximum du vraisemblance. La procédure (empirique) proposée par Ryden, pour se placer suffisamment proche des pa-

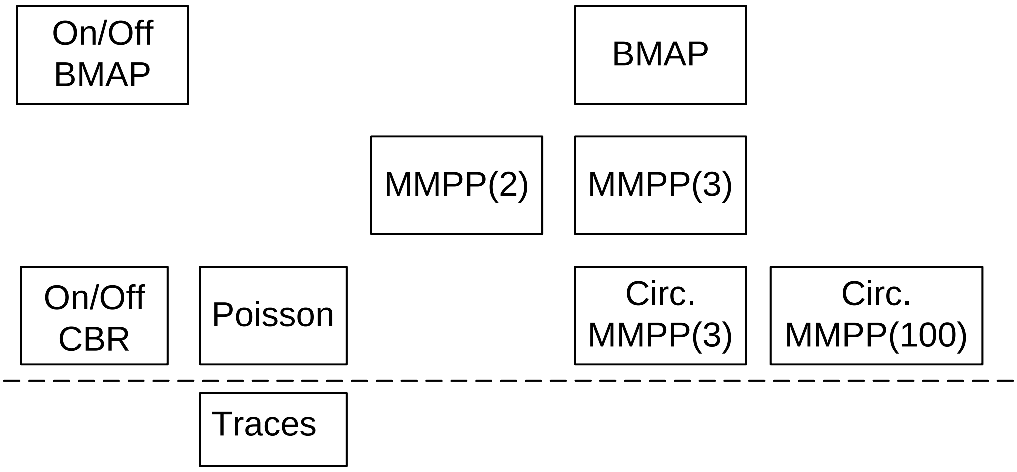


FIGURE 9.5.: Synthèse des différents modèles utilisés

ramètres idéaux, est employée. Des convergences rapides sont alors obtenues.

D'autres procédures pour l'initialisation de la méthode EM pour le cas de MMPP de dimension 2 sont envisagées, mais chacune d'elle est adaptée à un cas précis et il se trouve toujours des cas où l'initialisation est très éloignée de la solution définitive et induit un temps de convergence plus lent. Les procédures les plus adaptées sont indiquées.

9.3.4. Modèles de trafic pour les flux agrégés

9.3.4.1. Agrégation théorique

D'un point de vue formel, la superposition de deux MMPPs de dimension 2 est décrite par un nouveau MMPP de dimension 4, dont l'expression exacte est donnée. Le principe de la suppression d'états équivalents pour simplifier la gestion d'événements juxtaposés est également expliqué.

9.3.4.2. Méthodes envisagées

Des suggestions pour l'étude théorique de modèles agrégés sont données. Chacune représente une direction intéressante pour des études plus poussées. En particulier :

- **simplifications asymptotiques :** Dans ce cas, seul le comportement approché des distributions est conservé. Ce paramètre s'obtient à partir de la plus grande des valeurs propres des matrices du MMPP par exemple. Les modèles se simplifient beaucoup.

- **simplifications des matrices :** Dans [Ri02], le regroupement des états est introduit. La technique est principalement utile pour le calcul des statistiques de file d'attente mais relativement peu intéressante pour la détermination de modèles simplifiés. La méthode circulante semble capable de fournir des simplifications de modèles. La méthode a été revue mais certains résultats n'ont pu être reproduits. Ces travaux mériteraient d'être réexaminés et étendus.

9.3.5. Inventaire des modèles utilisés

Les modèles préconisés et inventoriés sont placés dans la hiérarchie du modèle en pyramide pour montrer à quel niveau ils seront utilisés. La Figure 9.5 montre les modèles étudiés et la Figure 9.6 indique leur localisation dans le modèle en pyramide.

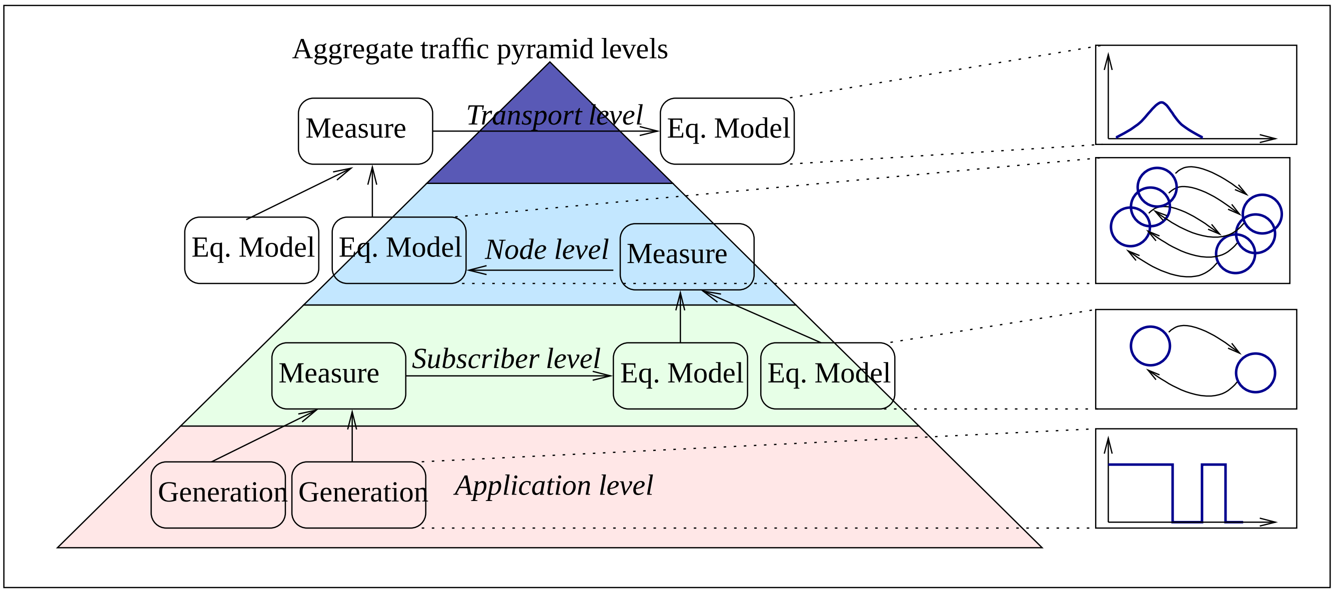


FIGURE 9.6.: Architecture des modèles dans la pyramide des trafics agrégés

9.4. Application du modèle en pyramide à un cas pratique (chapitre 4)

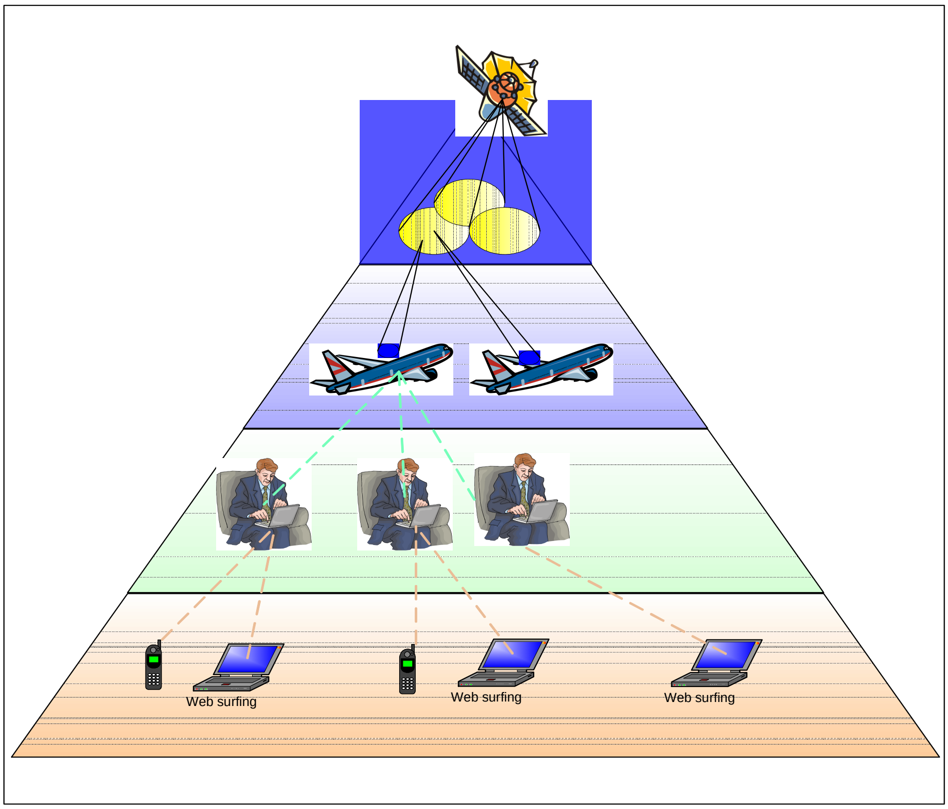


FIGURE 9.7.: Vue d'ensemble du scénario considéré

L'applicabilité du modèle en pyramide est illustré par un cas particulier, qui permettra la construction d'un simulateur afin de démontrer différentes équivalences entre les modèles. Ce simulateur est décrit dans ce chapitre et démontre l'intérêt des modèles agrégés.

9.4.1. Description d'un scénario pratique

9.4.1.1. Présentation du cas étudié

Le système considéré est un système de communication par satellite offrant aux passagers d'avions un accès au web et à la téléphonie sans fil durant le vol. Un tel système a été décrit en particulier dans le cadre du projet WirelessCabin [BCC+04a,BCC+04b]. L'élément principal d'un tel système est un satellite desservant une large zone géographique et couvrant de larges zones où l'accès terrestre est impossible. Un satellite géostationnaire (GEO) a été considéré.

Le modèle en pyramide se révèle approprié pour décrire les éléments contribuant à la génération de données. Le niveau applicatif est composé des deux applications fournissant les services d'accès web et voix (du point de vue de la description des services un modèle de voix sur IP sera préféré à un modèle GSM, même si l'architecture de WirelessCabin prévoyait le contraire). Le niveau souscripteur est composé de l'ensemble des passagers de l'avion utilisant le système. Le niveau nodal est composé de l'ensemble des avions. Le niveau transport est constitué par le système satellite. La Figure 9.7 illustre un tel système. La pyramide en fond indique les correspondances entre les modèles et les source de trafic à considérer.

9.4.1.2. Résumé des hypothèses sur le système

Il est difficile de construire le simulateur en gardant le système générique sans préciser les protocoles utilisés. La capacité dédiée aux services aéronautiques est considérée comme fixe et le but de l'exercice est de déterminer si les applications envisagées sont servies correctement.

Au niveau souscripteur, on suppose de façon standard que des paquets IP sont échangés. Au niveau du nœud, un procédé fragmentant les paquets a été choisi car la transmission par satellite s'effectue préférentiellement avec des paquets de taille fixe. La transmission par satellite est modélisée en considérant une capacité de N1 cellules par durée de trame T de 26.5 ms. Les cellules considérées sont de type $\mathcal{P}_4$ puisqu'on se situe au niveau transport. La ressource disponible est attribuée aux différentes cellules en fonction du type de cellule reçues de la couche nodale (paquets $\mathcal{P}_3$). L'affectation se fait d'abord pour le trafic de catégorie de service la plus exigeante, jusqu'à atteindre la capacité maximale pour chacun des nœuds considérés. Les cellules non affectées par cette première étape sont ensuite partagées entre tous les nœuds pour la seconde catégorie de trafic. Ce procédé introduit un retard si le trafic de première catégorie d'un nœud est en excès mais introduit un partage de ressource pour le trafic de seconde catégorie.

Les paquets $\mathcal{P}_3$ sont des cellules offrant une capacité nette de 424 bits et une en-tête de 56 bits. On s'aligne sur la taille d'une cellule ATM de 53 octets à laquelle on adjoint une en-tête spécifique pour le satellite. Le niveau nodal manipule donc des paquets de 480 bits.

Les paquets de niveau souscripteur $\mathcal{P}_2$ ont été modélisés avec une en-tête de 48 bits. En effet, TCP et IP ajoutent chacun une entête de 20 octets chacun, ce qui fait qu'en moyenne pour un paquet de données, un ajout de 48 octets doit être considéré. Cette entête est cependant compressée en utilisant un mécanisme la reduisant à 32 bits dans le meilleur cas. Un taux de compression de 1 :8 a été considéré, ce qui correspond bien à une entête de 48 bits. Cette valeur est proche de la compression d'entête maximale (32 :(48*8)).

Au niveau applicatif, les tailles réelles des paquets considérés sont pour la voix une taille standard de 800 bits et pour le web une taille variable de 752, 4600, 11712 bits (modèle BMAP).

D'autres détails numériques complétant les informations ci-dessus et concernant la construction des scénarios simulés est donnée dans les paragraphes 4.1.2 et 4.1.5.

9.4.1.3. Stratégies de qualité de service

Le standard DVB RCS prévoit 4 types de catégories de service (dénommés CRA, RBDC, VBDC et FCA). Le simulateur se limite à deux types de trafic correspondant aux deux catégories prioritaires.

L'application voix sur IP est plus contraignante au niveau des variations de délai et de gigue acceptables. On a donc pour la voix un mécanisme proche du CRA (assignement continu de ressource). Pour le trafic web, on a une gestion proche du meilleur effort possible. Cette gestion est proche du FCA (assignement de la ressource disponible).

9.4.1.4. Mécanismes d'allocation de ressources

Les mécanismes de gestion, en particulier, tel qu'implémentés dans les satellites regénératifs sont trop complexes pour être reproduits exactement dans le simulateur. En effet différentes entités, comme le gestionnaire bord de ressource ou le nœud de gestion central sont responsables de la gestion des ressources. Elles travaillent de manière dynamique en échangeant de l'information pour chacune de connections qui sont initiées ou terminées. Dans le simulateur, on considère les connections établies, le mécanisme de gestion de ressource peut être simplifié. Principalement, le simulateur doit modéliser la ressource limitée du lien satellite et la gestion de ressource donnant la priorité à certain type de trafic.

9.4.2. Particularisation des opérateurs

Les opérateurs introduits au chapitre 2 sont repris. En précisant certains paramètres (numériques) les caractérisant, leur action est davantage particularisée. Leurs effets sur les flux de trafic sont étudiés avec plus d'exactitude que lors de leur introduction générale.

9.4.2.1. Niveau souscripteur

Tout d'abord, l'opérateur d'addition est présenté. La sortie de cet opérateur correspond à l'ajout des flux de trafic appliqués en entrée. Dans le cas présent, les flux d'entrée sont des flux du niveau applicatif. La construction du flux de sortie en termes de taille de paquet et d'instant d'émission est précisée. Ensuite l'opérateur de conversion de niveau est présenté. Le passage de paquets du niveau $\mathcal{P}_1$ à des paquets de niveau $\mathcal{P}_2$ est expliqué. Dans ce cas, l'opération se limite à l'ajout d'une en-tête. Seule l'information de taille est modifiée. Finalement, l'opérateur de mise en conformité avec le niveau est introduit. A ce niveau particulier, il n'y a pas d'action sur la nature des flux. L'écriture théorique de la construction d'un flux de niveau 2 à partir de deux flux de niveau 1 est introduite, en utilisant les trois opérateurs précédemment définis.

9.4.2.2. Niveau nœud

L'opération d'addition est identique à l'opération au niveau précédent. L'opération de conversion de niveau nécessite d'assurer une taille constante des paquets considérés au niveau 3, puisque ces paquets $\mathcal{P}_3$ ont une taille fixe de 480 bits. L'opération de segmentation est donc présentée, ainsi que l'action qu'elle a sur le nombre de paquets. L'opérateur de conformation au niveau supérieur est ici à nouveau supposé sans influence particulière puisque la taille des paquets est fixe.

9.4.2.3. Niveau transport

Le niveau transport a un fonctionnement synchronisé par des salves cadencées à $T = 26.5ms$. C'est-à-dire que les paquets ne sont envoyés qu'à ces instants discrets régulièrement espacés. Ce comportement est reproduit dans le simulateur. Si les paquets à transmettre sont plus nombreux que ne le permet la salve en cours, ces paquets seront retardés jusqu'à ce que leur transmission soit possible dans une future trame. Le flux de niveau 3 subit des modifications lors de sa transmission au niveau transport. De plus, la règle de construction de la cellule de sortie limitant le volume de sortie de chaque nœud pour le trafic de type 1 perturbe les flux émis et cette limite affecte les volumes émis des flux de type 1 (servis avec la première qualité de service). La bande passante restante est partagée équitablement entre les cellules de niveau inférieur. Ces actions sont modélisés pour décrire les modifications intervenants au niveau transport.

9.4.2.4. Autres opérateurs

D'autres actions ne correspondant pas aux opérateurs introduits au chapitre 2, doivent être introduites dans le simulateur. Il s'agit des opérateurs effectuant l'action réciproque de celles des opérateurs au niveau de la réception, pour *désagréger* le trafic. Le principe de ces opérateurs supplementaire est expliqué.

9.4.3. Étude avec un simulateur

9.4.3.1. Justification d'une approche reposant sur un simulateur

L'utilisation d'un simulateur permet l'évaluation des performances d'un système avant sa mise en service. Ceci se justifie en particulier pour des systèmes de communication dont le déploiement est long et coûteux. Cela permet aussi de se concentrer sur le problème de dimensionnement sans la nécessité d'avoir chacun des systèmes prêts à être testés. En effet, les facteurs dimentionnants du système peuvent ainsi être modifiées à loisir. De plus, le simulateur préexiste au système deployé et permet d'appréhender très tôt les caractéristiques du fonctionnement du système. Ainsi, par exemple, un compromis performance et complexité peut être finalisé progressivement.

9.4.3.2. Principes de conception

Le simulateur est construit sur la base d'un système à événements discrets (DES - discrete event simulator) se nommant OMNET++. Ce système permet d'intégrer : i) des modèles en C++ implémentant les générateurs de trafic, différents éléments de l'architecture (opérateurs et éléments de mesure), ii) une description de l'architecture des différents modèles (fichier NED de description de réseau), iii) une description des paramètres en entrée du simulateur. Une fois le simulateur élaboré (implémentation et tests des modules nécessaires au simulateur), les différents scénarios sont décrits avec différent *réseaux* correspondant à chacun des cas avec leurs paramètres (le mot réseau est utilisé par OMNET++).

9.4.3.3. Gains attendus et limite de l'approche

Le but des simulations est de montrer que les modèles sont équivalents. Pour cela, on simule chacun des cas et on prouve par l'observation de certaines mesures que les deux cas se correspondent bien. Les inconvénients de l'utilisation d'un simulateur sont que ses performances peuvent rapidement baisser et que le simulateur peut devenir plus lent qu'une agrégation en temps réel des flux, si trop d'éléments sont considérés, modèlisés et mémorisés dans le simulateur. En conséquence, il faut que le simulateur n'intègre juste que le niveau de détails nécessaire à la mise en évidence de certains phénomènes. Utiliser trop de détails complexifie inutilement le simulateur (implémentation, validation des modèles) et inversement un simulateur trop simpliste ne permet pas de mettre en valeur certains effets. Puisque l'on travaille ici à différentes échelles (et avec différentes vues allant de modèles détaillés à un niveau macroscopique), considérer ces questions est particulièrement important. Finalement, on attend du simulateur qu'il mette en valeur les gains en terme de complexité de simulation (modélisation de flux semblables pour une complexité moindre) lorsque des modèles agrégés sont utilisés. Il justifiera également la grande granularité et modularité de l'approche proposée.

9.4.3.4. Architecture du simulateur

L'architecture du simulateur est présentée en Figure 9.8. On y voit des modules dédiés à la génération de trafic et des modules dédiés à l'analyse du trafic. On y retrouve (horizontalement) les différents niveaux du modèle en pyramide. Sur le plan vertical, sont indiquées les différentes instances des niveaux considérés. On exposera rapidement le fonctionnement de ces modules de génération plus loin. On inclut également des modules pour réaliser les opérations liées aux transformations requises par les opérateurs. De plus, le simulateur doit inclure une interface pour le post-processing. En général, Matlab a été utilisé pour la réalisation des analyses de post-processing.

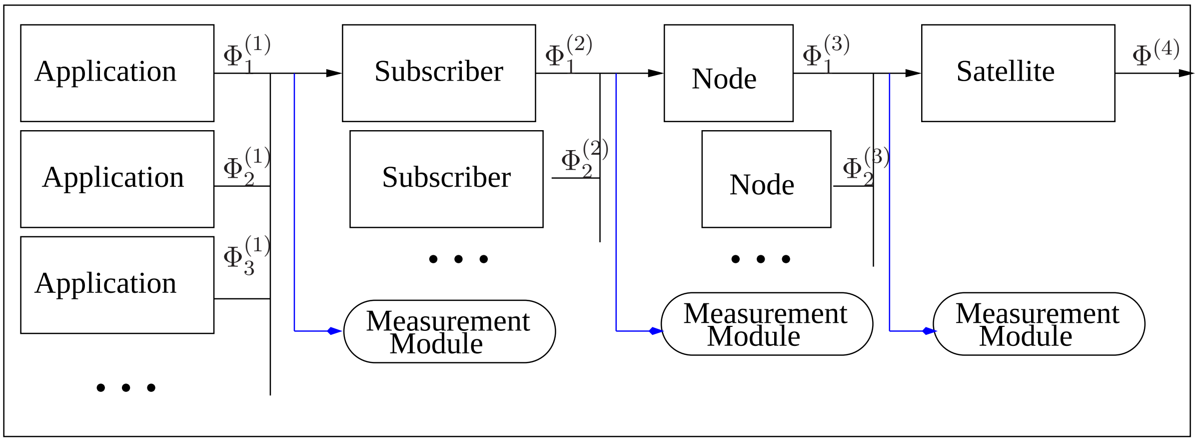


FIGURE 9.8.: Architecture du simulateur

9.4.3.5. Liens entre le simulateur et le modèle en pyramide

Comme l'objectif du simulateur est de valider numériquement les équivalences au sein du modèle en pyramide, les opérateurs sont repris pour indiquer ils sont mis en œuvre dans le simulateur.

- **Niveau application :** la conversion du niveau 4 au niveau 3 s'effectue au moyen de trois opérateurs. Dans le cas étudié, on peut se limiter à l'opérateur d'addition des flux puisque le générateur ne génère aucun paquet nécessitant une segmentation et qu'il n'y a pas de contrainte sur les paquets du niveau 3.
- **Niveau souscripteur / passager :** l'opération principale est la conversion des paquets de niveau 3 en paquets de niveau 2. Cette opération correspond à la collecte des trafics de chaque utilisateur pour les transmettre à un nœud central.
- **Niveau nœud / avion :** on a ici un nœud permettant l'accès au satellite pour l'ensemble des passagers d'un avion.
- **Niveau transport / satellite :** de même, on passe ici de flux du niveau 2 à des flux de niveau 1. Les mécanismes d'accès à la ressource y exercent leur influence sur la transmission et les propriétés des flux de trafic.

De plus, le coût supplémentaire liées aux conversions des formats de paquets est analysée. Par exemple, le coût de l'encapsulage jusqu'au niveau de transport est analysée en fonction de la taille initiale des paquets.

Scénarios pour l'étude d'équivalences de trafic Les différents cas (dans la composition des sources ou des modèles utilisés) considérés lors des comparaisons pour justifier les équivalences sont énumérés. Ces cas sont donnés tant en terme des flux ayant une équivalence potentielle dont l'écriture abstraite est indiquée, qu'en inventoriant la composition des entités composant les différentes pyramides de trafic.

9.4.4. Description des modules principaux du simulateur

9.4.4.1. Modules de génération de trafic

Les modules dédiés à la génération de trafic sont implémentés en respectant le principe des événements discrets qui facilite leur implémentation puisque la gestion du temps est assurée par un moteur interne au simulateur. En particulier l'implémentation comprend :

- un modèle voix - On-Off alterné à durée exponentielle : Pour modéliser les sources de voix, un modèle On-Off oscillant entre deux états a été choisi. Les principes de conception du module sont exposés avec des exemples en pseudo code qui illustrent les détails de l'implémentation.

- un modèle web BMAP(3,3) : Le principe d'implémentation de ce modèle est exposé. Le BMAP est basé sur une chaîne de Markov dont les sauts déterminent l'état (la phase dans laquelle se trouve la chaîne).

9.4.4.2. Modules d'analyse

Deux types de modules d'analyses sont décrits : ceux calculant certaines statistiques obtenues en mettant le trafic en entrée d'une file d'attente et ceux obtenues à partir la séquence des instants d'arrivée. Les points suivants sont traités :

- files d'attente : Comme de nombreux résultats concernant les files d'attentes sont disponibles, un module dédié permet d'obtenir ces statistiques. Sans considérer que les transmissions dans le système peuvent être décrites par la théorie des files d'attentes, ni vouloir construire un modèle de file d'attente pour l'ensemble de la pyramide, le flux d'entrée est caractérisé au travers des variables que traite la théorie des files d'attente. Ainsi, le retard subi par des paquets traité par un serveur avec un temps de traitement donné peut être obtenu par un module dédié qui caractérisera le flux. Les paramètres du module doivent être adaptés. Le retard subi par les paquets peut être décrit de diverses manières : par la distribution de la taille de la file d'attente lors de l'arrivée d'un paquet ou à un instant quelconque (distribution discrète) ou par la distribution du durée d'attente (ou de jusqu'à la fin du service - distribution continue). Des résultats théoriques sont disponibles pour de nombreux types de queues.

- temps inter-arrivée : Voir en annexe A où la description est faite d'une méthode itérative pour estimer des grandeurs basés sur les temps inter-arrivée (en particulier l'obtention de la fonction de corrélation est décrite).

- estimation de la durée des sessions : Ce module évalue la durée des sessions de transfert web en se basant sur les paquets reçus et en utilisant un identifiant ajouté lors de la génération. Il permettra d'évaluer la distortion de la durée des sessions pour différentes charges.

9.4.5. Description des sorties délivrés par le simulateur

Durant une simulation, le simulateur collecte divers paramètres qui sont ici précisés. En particulier, le simulateur permet d'obtenir :

- des mesures qui se se regroupent en deux catégories principales. La première liée à la transmission du trafic et la seconde liée l'estimation des performances des flux échangés.

- des indicateurs de performance : Ils renseignent sur les performances du simulateur lui-même, comme, par exemple, le décompte des occurrences d'événements ou le ratio entre le temps simulé et le temps réel qui c'est ecoulé durant la simulation. Ces indicateurs sont élaborés à partir de données internes du simulateur.

9.5. Agrégation pyramidale de modèles voix (chapitre 5)

Dans cette section, les résultats obtenus dans un cas simple sont présentés. Une architecture hiérarchique respectant le modèle en pyramide a été choisie, où un nombre limité de modèles est employé. L'idée est de parcourir les différents niveaux de la pyramide et de vérifier les équivalences attendues.

Les résultats présentés correspondent à des compositions telles que celle décrite dans la Figure 9.9 où l'agrégation homogène d'utilisateurs est présentée. Le système est étudié avec un nombre croissant d'utilisateurs utilisant la voix sur IP (VoIP). Chaque nœud correspond à une agrégation de modèles identiques. Ainsi, par exemple, le scénario 1.1 correspond à 100 sources voix. En partant de ce type de sources, un modèle agrégé sera progressivement construit et les équivalences entre modèles seront mises en évidence.

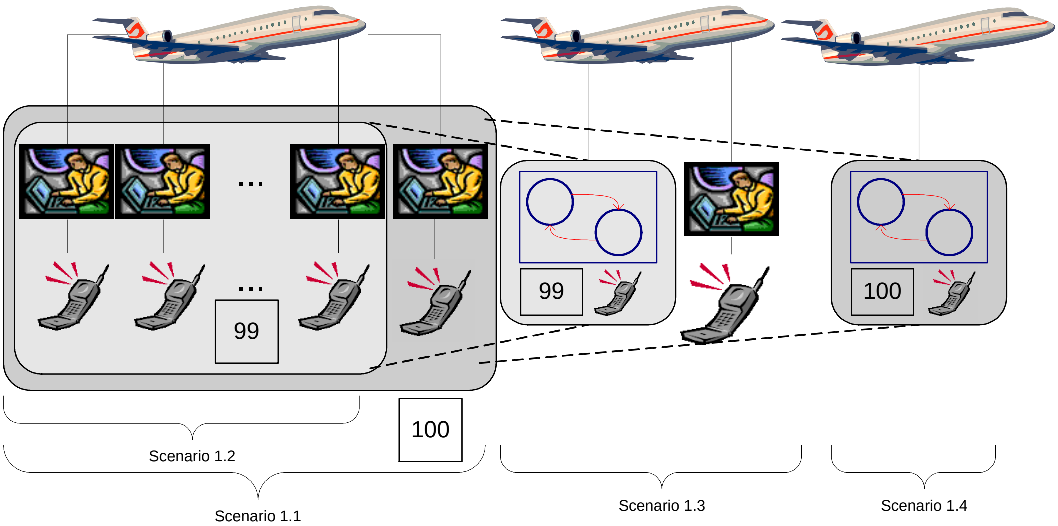


FIGURE 9.9.: Description du premier jeu de scenarii considérés

9.5.1. Représentation en pyramide du cas étudié

La modélisation dans les différents niveaux de la pyramide peut être effectuée.

Niveau applicatif Un unique modèle identique à celui décrit au chapitre 4 est employé. Ce modèle représentent une source de voix en générant un certain nombre de paquets de niveau 4 ($\mathcal{P}_4$).

Niveau souscripteur / niveau nœud Un modèle MMPP(2) comme équivalent de 100 sources voix est dérivé, en utilisant les méthodes de détermination des paramètres exposées au chapitre 3. Les équivalences seront basées pour la plupart sur la méthode de Kang basé sur les asymptotes du retard de la file d'attente (le modèle équivalent est construit pour avoir la même comportement pour une file d'attente donnée que le modèle initial). Les flux de trafic ont les propriétés suivantes :

- caractéristiques des flux agrégés
le comportement général est le suivant : au travers de la distribution de débit moyen des sources, un comportement moyen qui tend vers la moyenne est obtenu pour le modèle agrégé. Il y a perte de représentativité lorsqu'un modèle agrégé est employé plutôt que les modèles individuels initiaux. Cependant cette perte ce fait par un gain de modélisation en utilisant des modèles reproduisant le comportement global. Ainsi, la bande passante peut être économisée en groupant les sources, au lieu de réserver la valeur maximum de la source à sa valeur moyenne. Les propriétés d'autocorrélation des instants d'arrivées obtenus lors de l'emploi des modèles agrégés ont été conservées par rapport à celles des modèles initiaux. Ceci indique un comportement similaire.

- paramètres déterminés
Les méthodes utilisées pour la détermination des paramètres sont décrites. En fonction des cas, différentes méthodes sont utilisées. La méthode la plus efficace pour chaque cas a été employée. Pour les méthodes EM, la difficulté vient des procédures utilisées pour l'initialisation. Pour une plage allant de 80 à 130 utilisateurs les paramètres du MMPP(2) ont été obtenus et permettent de comparer chacun d'entre eux. Un comportement linéaire des paramètres est observé, ce qui prouve que le modèle agrégé s'adapte à la charge de trafic désirée. La proportionnalité avec la charge de trafic est donc conservée par le modèle macroscopique.

Niveau transport A ce niveau, un premier équivalent gaussien est obtenu qui prouve l'équivalence entre modèle précis et le modèle agrégé correspondant.

9.5.2. Justification des équivalences

Les équivalences ne peuvent pas aboutir à la génération de deux flux identiques puisqu'il s'agit de modèles aléatoires. En revanche on doit obtenir des mesures similaires des distributions, puisque les comportements statistiques des modèles doivent rester identiques.

Niveau Nœud

Les équivalences suivantes sont vérifiées :

Volume généré dans la pyramide : Le volume de deux modèles doit forcement correspondre pour qu'on puisse parler d'équivalent. Ce critère est vérifié.

Caractéristique des files d'attentes Au niveau transport, l'analyse ([1]) montre que la distribution des tailles de paquets des différents modèles est identique pour le modèle initial, l'équivalent MMPP(2) simulés ainsi que la distribution obtenue théoriquement. L'équivalence des modèles est ainsi démontrée.

Niveau Transport

Les analyses suivantes sont conduites :

Analyse du volume La bonne adéquation entre deux réalisations de débits équivalents pour un modèle constitué de 50 modèles agrégés et de 50 fois un nœud de 100 source voix a été vérifiée ([2]).

Analyse des retards : La répartition des retards au niveau applicatif pour deux cas équivalents est analysée ([3]). Des estimations semblables de cet indicateur de performance du système sont obtenus pour chacun des cas.

9.6. Modèles nœuds équivalent pour différentes conditions de trafic (chapitre 6)

Il y a un plus grand intérêt à utiliser un modèle équivalent au niveau du nœud qu'au niveau souscripteur, car l'agrégation est plus importante et qu'au niveau souscripteur seul un nombre restreint d'applications peuvent être considéré. L'intérêt d'un équivalent au niveau nœud est dégagé dans cette section. Cependant, l'influence des opérateurs est plus marquée puisque davantage de niveaux de transmission sont traversés. Des modèles plus complexes que précédemment sont employés, par exemple, le modèle BMAP(3) qui permet de décrire des tailles de paquets différentes.

9.6.1. Présentation des résultats

9.6.1.1. Cas d'étude

La Figure 9.10 présente les scénarios considérés pour des applications hétérogènes. Par exemple, un scénario incluant 100 sources voix et un modèle individuel BMAP(3) pour le trafic web est considéré. Ce scénario est dénommé 3.1. Le scénario 3.2 consiste en un seul modèle pour le trafic web et 100 modèles individuels pour le trafic voix. Le scénario 3.3 consiste en un unique modèle BMAP(3). L'équivalence entre chacun des ces modèles, une fois les paramètres des modèles déterminés, sera démontrée.

9.6.1.2. Données collectée au niveau nodal

Les résultats collectés durant les simulations sont rassemblés : des estimations statistiques de la répartition de la taille des paquets dans une file d'attente située au nœud d'accès et du volume de trafic généré sont obtenus. De plus, au niveau de la couche transport, le volume est mesuré pour valider les équivalents obtenus. C'est avec ces analyses que les équivalences sont démontrées.

[1] cf. figure 5.8 pour la visualiser
[2] cf. figure 5.9
[3] cf. figure 5.10

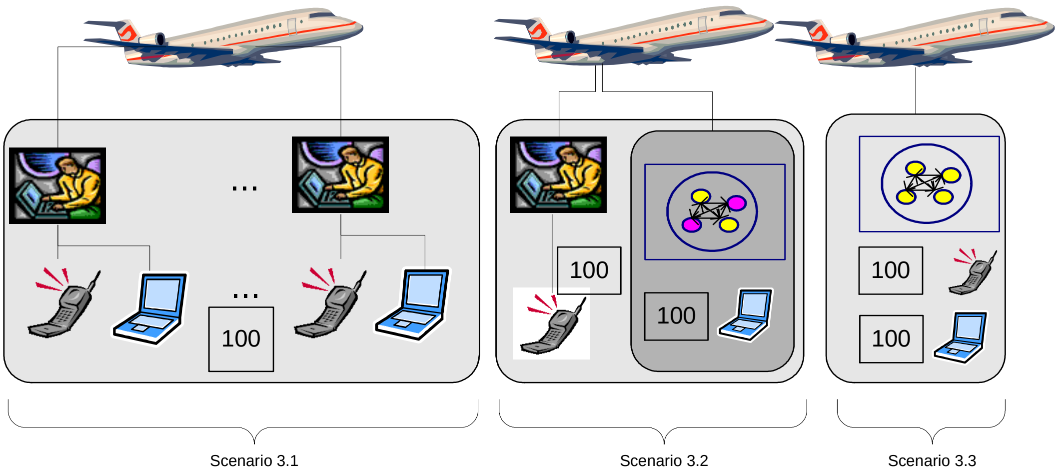


FIGURE 9.10.: Description du second jeu de scenari considérés

9.6.2. Sources supplémentaires considérées

9.6.2.1. Emploi d'un modèle pour le flux vidéo

Un modèle vidéo, utilisé uniquement à ce niveau, est décrit. Dans le cadre d'une étude indépendante, des modèles MMPP(2) ont été déterminés pour du trafic vidéo. Les paramètres du modèle déterminé pour différentes charges sont indiquées pour les différents cas.

9.6.2.2. Emploi d'un modèle de surf web

Les modèles suivants ont été utilisés :

Modèle pour le modèle de surf web sur un lien dorsal Pour modéliser un tel trafic, un modèle BMAP à 3 états et avec 3 tailles de paquets a été choisi. Des tailles de 94 bytes, 575 bytes et 1469 bytes pour chacune des catégories sont employées. Ce modèle est placé en entrée d'une file d'attente à durée de service fixe (par byte de paquet à servir) et la distribution de la taille de la file d'attente est observée. Les résultats sont représentés Figure 9.11. Une très bonne correspondance entre la distribution obtenue par simulation et la répartition théorique est obtenue (sur le graphes les courbes sont presque confondues). Le comportement asymptotique est étudié. Une correspondance avec l'analyse théorique (basée sur les valeurs propres des matrices du modèle) est également obtenue. La Figure 9.12 indique les distributions de paquets obtenues. Les paquets de tailles petites (moins de 500 bytes) et moyenne sont les plus fréquentes, ce qui est cohérent avec les répartitions dite éléphants-souris qui ont été mesurées pour le trafic web.

Modèle web individualisé Pour obtenir un modèle individualisé, le modèle précédent a été modulé par un processus on/off. Deux distributions exponentielles pour chacune des phases on/off ont été employées. Durant la phase ON, les paquets sont indiqués comme originaire une même session. Cela permet d'obtenir des indications à la réception sur les retards de transmission pour une session. En particulier, l'effet de la bande restreinte du niveau transport peut être évalué (avec une prolongement de la durée des sessions) et ce pour différentes charges.

Agrégation de modèles individuels Une première vérification a été effectuée à l'aide de multiples modèles individuels. Pour 100 modèles individualisés de durée $T_{On} = 10$ min et $T_{Off} = 50$ min, on s'attend à un trafic similaire à celui obtenu pour 16 modèles BMAP initiaux. Le nombre de sources actives à un instant donnée a une distribution binomiale.

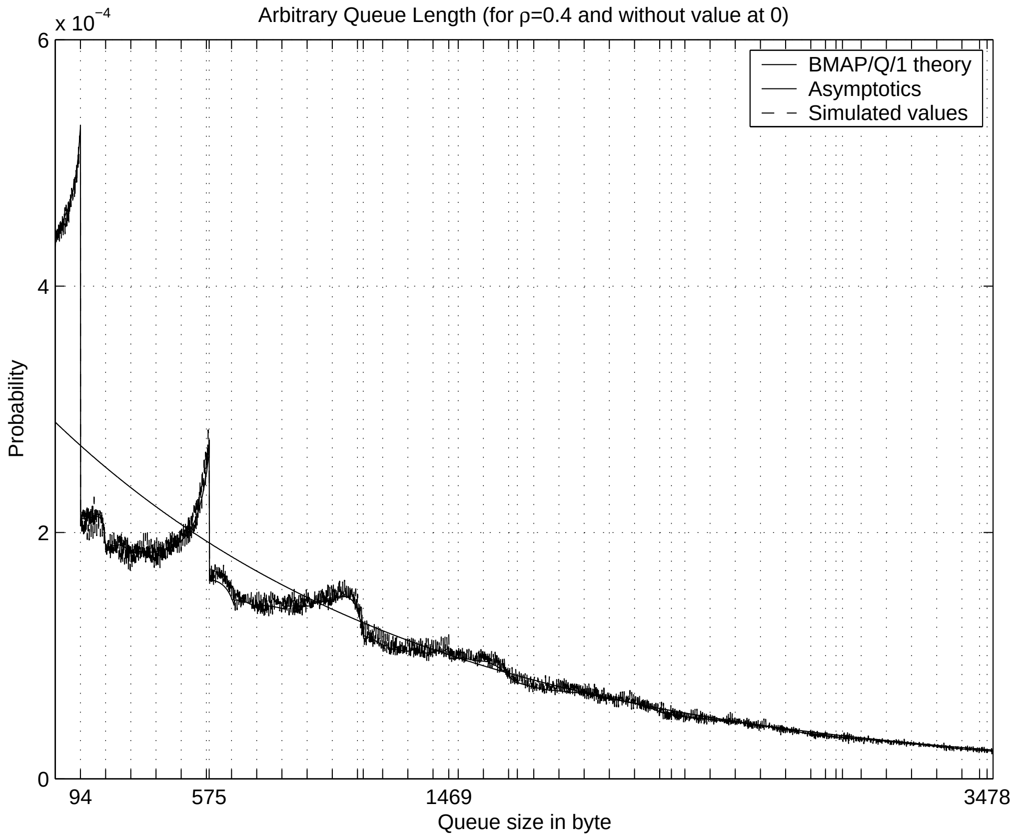


FIGURE 9.11.: Propriétes de la file d'attente pour le flux généré par un utilisateur de surf web unique

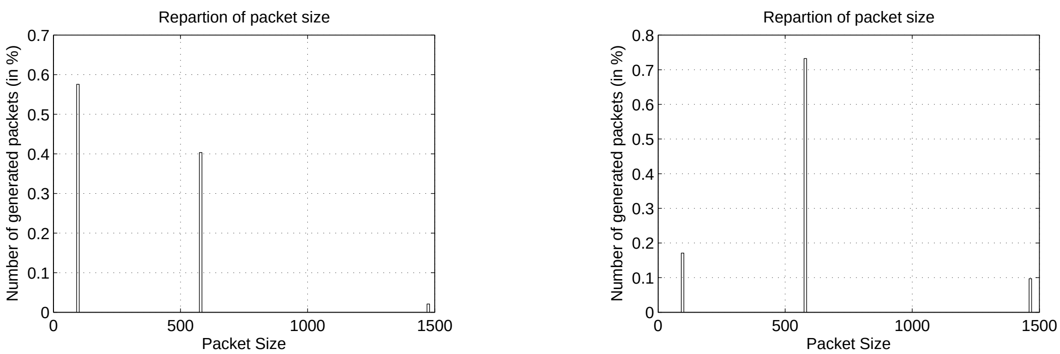


FIGURE 9.12.: Repartition des tailles de paquets générés par le modèle web (en taille et volume)

9.6.3. Modèle équivalent pour le trafic hétérogène

Les paramètres d'un modèle BMAP(3,3) obtenus comme équivalent d'une trace correspondant au modèle BMAP précédemment présenté auquel on a adjoint 100 modèles voix (modèle précis voix) est indiqué. Les tailles de paquets générés par le modèle a d'abord été déterminée. Une comparaison de la répartition pour chacune des catégories est effectuée : la taille de la catégorie inférieure a été augmentée de par la génération de davantage de paquets de 100 bytes par le modèle voix ; les tailles pour la catégorie médiane et supérieure sont relativement semblables.

9.6.4. Équivalents gaussiens pour le trafic au nœud

Des modèles au niveau 4 sont proposés pour lequel la validité de l'équivalence est vérifiée au niveau de chaque nœud.

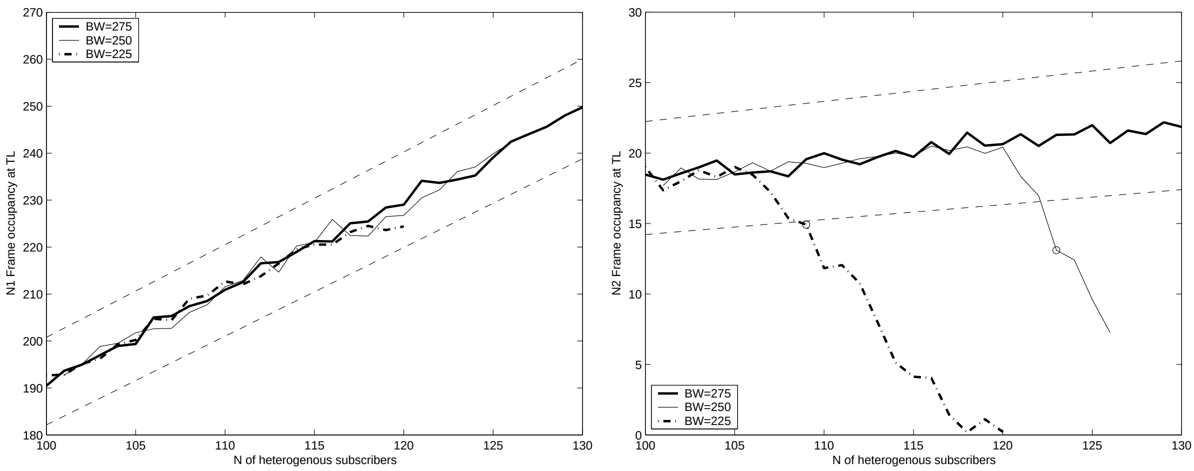


FIGURE 9.13.: Comparaison du nombre de cellules transmises (de type 1 à gauche, de type 2 à droite)

Détermination des paramètres dans le cas gaussien Les paramètres (moyenne variance et coefficient d'auto/corrélation) ont été obtenu pour différentes charges (un nombre croissant d'utilisateurs du modèle voix et BMAP individualisé). Deux catégories de trafic sont considérées : la première pour la voix et la seconde pour le web. L'hypothèse a été faite d'une capacité maximum au niveau transport de 250 trames par salve. Il en résulte que le nombre de cellules de type 1 augmente de façon constante lorsque le nombre d'utilisateur augmente. Par contre, un fois un maximum atteint, le nombre de cellules de type 2 décroît à cause des mécanismes de gestion de ressources.

Influence de la bande passante au niveau transport Différentes limites maximales de la capacité du lien au niveau transport ont été considérées. En modifiant la capacité maximale du lien (à 225 ou 275 cellules par trame), il est observé que le volume de trafic de type 1 n'est pas modifié (pour une limite de 225 par trame, l'étude s'est limitée à 120 utilisateurs, pour rester en dessous d'un trafic maximum). En revanche en déplaçant le seuil, le nombre de cellules de type 2 transmises diminue. La distinction de deux types de qualité de service permet d'affecter en premier lieu les cellules classifiées comme les moins importantes. Ceci est représenté sur la Figure 9.13.

9.6.5. Analyse du volume de trafic

Les analyses suivantes ont été conduites :

- cas du surf web : Dans ce cas, des simulations du trafic de surf web utilisant le modèle BMAP original et le BMAP modulé par un processus on-off ont été effectuées (des valeurs de $T_{on} = 1$s et de $T_{off} = 9$s ont été utilisées). Un volume total de trafic identique est généré dans chacun des cas. La Figure 9.14 montre le volume global généré. En outre, la répartition des tailles des paquets générés a été comparée. L'utilisation d'un modèle individuel est ainsi validée.

-cas hétérogène : Le volume généré par chacune des applications a été mesuré pour un trafic composé d'applications web et de l'application voix.

9.7. Étude d'un système complet (chapitre 7)

9.7.1. Synthèse des résultats présentés

9.7.1.1. Cas étudiés

La Figure 9.15 montre les nœuds où se situeront les sources de trafic. Le premier cas est dénommé type I et est constitué de modèles précis. Le second cas est de type II et utilise le modèle équivalent de nœud web décrit au chapitre précédent. Le troisième cas, dénommé type III, utilise un modèle agrégé unique.

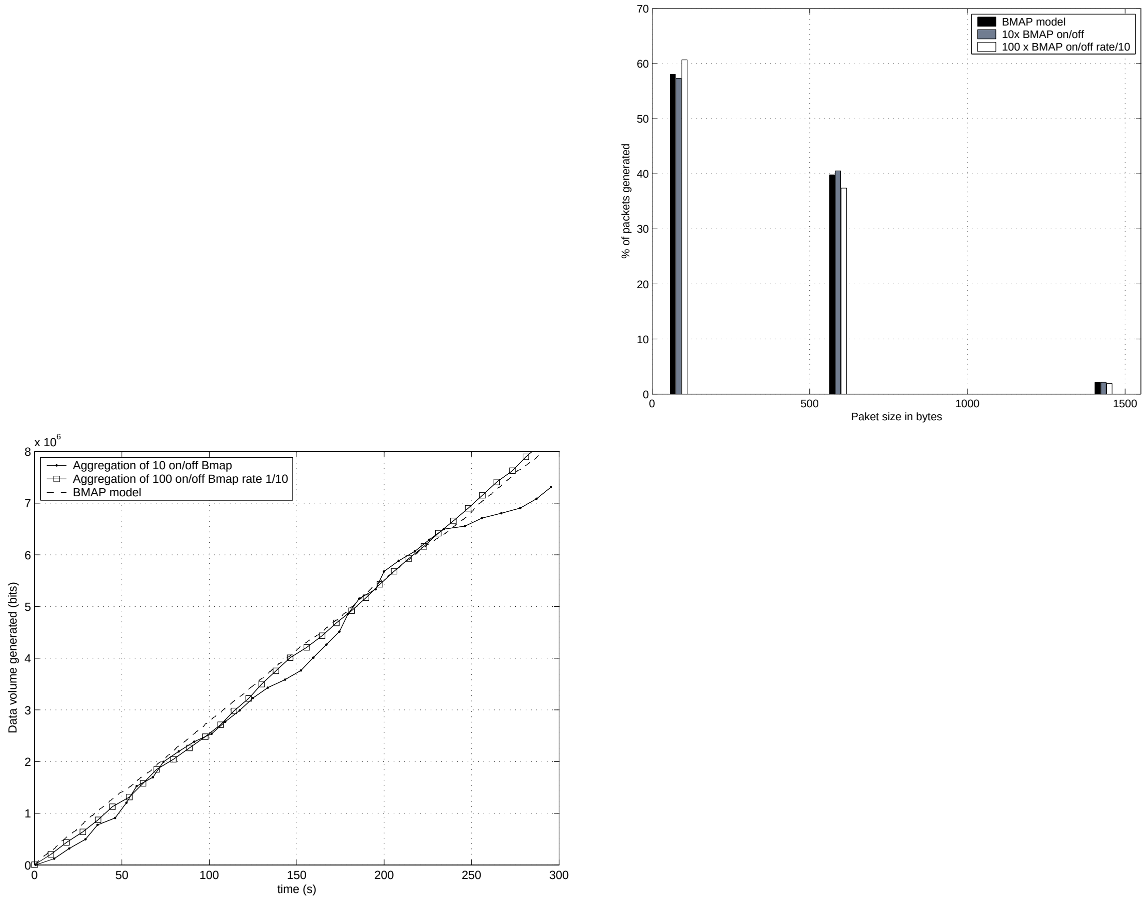


FIGURE 9.14.: Aggrégation de modèles (deux modèles On/Off and un modèle BMAP continue) comparaison du volume de trafic et de la distribution des tailles de paquet

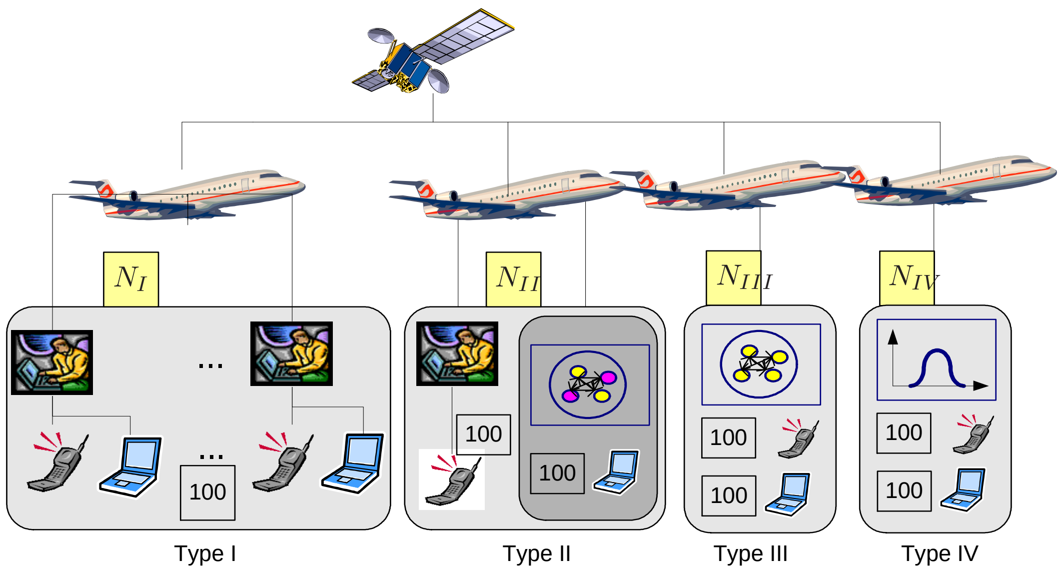


FIGURE 9.15.: Éléments considérés dans le cas d'un nœud multiple

Le type IV utilise un modèle gaussien pour simplifier le flux. Ces cas sont construits pour pouvoir établir les équivalences entre chacun de ces modèles.

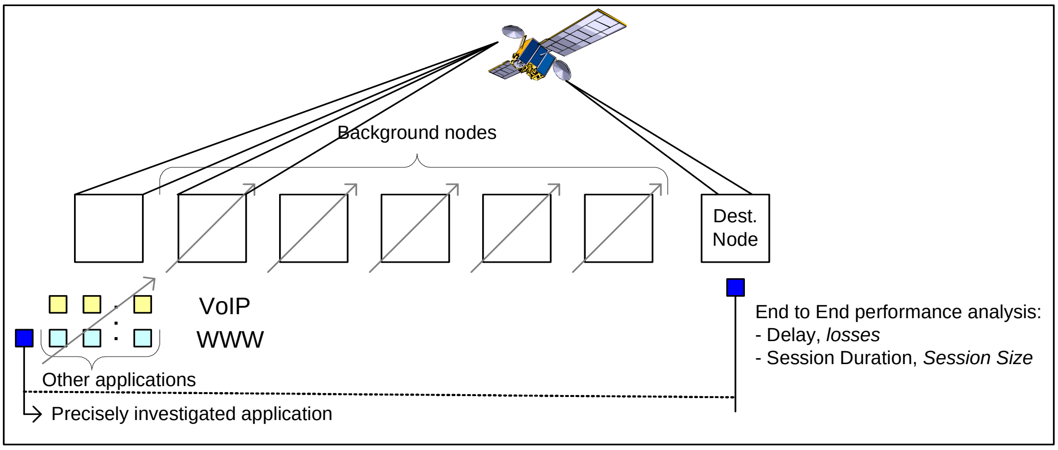


FIGURE 9.16.: Paramétres ajustables pour le cas d'évaluation des performances

9.7.1.2. Données collectées au niveau transport

Le simulateur permet de collecter le volume de trafic généré au cours de la simulation, d'obtenir la répartition du nombre de cellules transmises pour chacun des types considérés et des mesures d'un critère de performance. En particulier, le simulateur permet d'évaluer la durée des sessions web. L'étude vise bien à caractériser les flux de trafic jusqu'au niveau transport en utilisant le modèle en pyramide.

9.7.2. Validation des équivalents nœuds au niveau transport

Les résultats du chapitre précédent pour un nœud unique sont revisités et visualisés au niveau transport. En particulier le type de cellule de chaque catégorie transmise par la couche transport est analysé pour chaque trame, afin de conclure qu'une bonne adéquation entre différents types de modèle est obtenue. Si certains modèles ne sont pas capables de reproduire exactement les propriétés statistiques des flux, la cause est qu'une perte d'information s'est produite lors de la simplification ou bien que les modèles sont construits à partir de propriétés différentes que celle observées pour l'équivalence (par exemple, les modèles bâtis sur l'auto corrélation peuvent avoir une très faible équivalence lorsqu'on observe la correspondance de la variance).

9.7.3. Description du système étudié

Pour cette étude, les nœuds de type I(p) et III(p) sont décrits. Une analyse de la durée des sessions après sa transmission dans la pyramide est menée.

Indicateurs de performances Des nœuds dans lesquels un application de surf web marquant ses paquets est employée, afin d'obtenir des informations sur la transmission de bout en bout en identifiant l'appartenance des paquets aux différentes sessions.

Description des flux de trafic La Figure 9.16 précise quels modules sont considérés comme générateurs exclusifs et comme des générateurs permettant l'analyse des performances. La charge choisie pour une simulation particulière est réglée par un seul paramètre qui est le nombre de passagers par avion qui configure le nombre d'instances utilisé au niveau applicatif. Le nombre de nœuds considéré est de 50 nœuds. Les flux étudiés seront composées, dans un cas où l'on s'intéresse aux performances, d'un nœud I(p) ou III(p) et de 49 nœuds d'un type autre (I, II, III ou IV). L'emploi de nœuds similaires a été privilégiée.

9.7.4. Étude du comportement du système

L'influence du nombre de nœuds sur les résultats obtenus au niveau transport est étudiée. Les modifications subies par le flux de trafic lors de sa transition du niveau nodal au niveau transport sont également

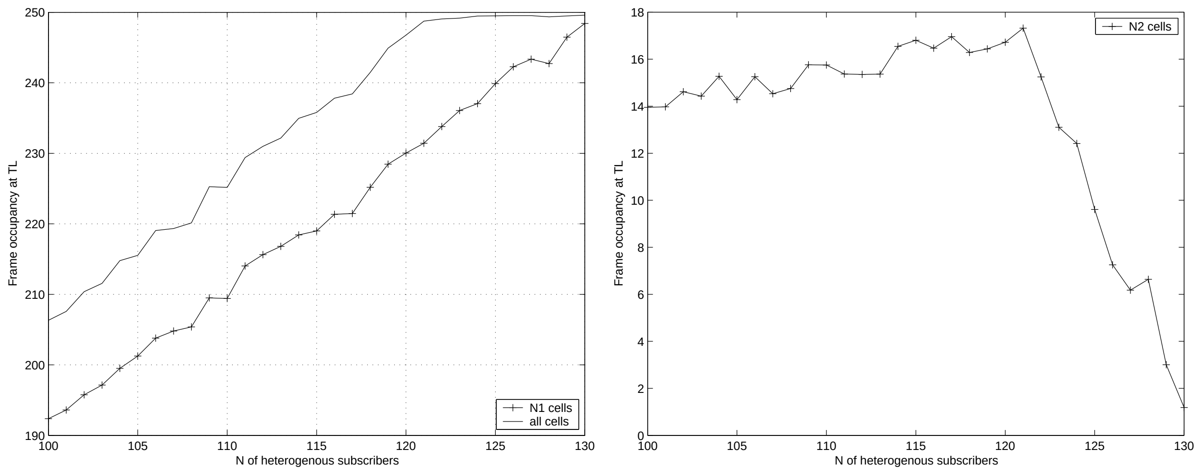


FIGURE 9.17.: Valeur moyenne de N_1 et N_2 pour un nœud unique pour une charge variable

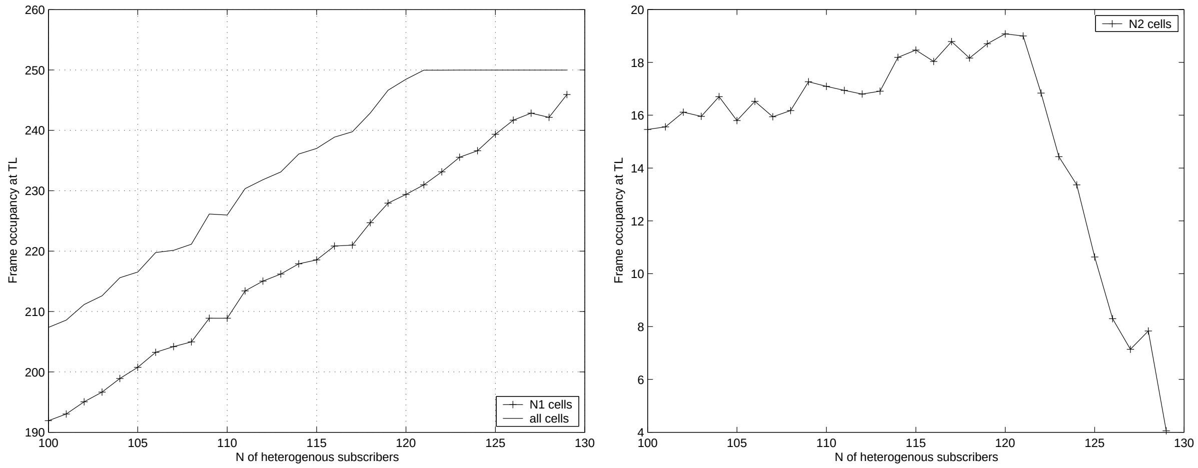


FIGURE 9.18.: Valeur moyenne de N_1 et N_2 pour des nœuds multiples pour une charge variable

considérées. Suivant les cas, le système se compose soit d'un unique nœud soit de 50 nœuds identiques. Les résultats suivant sont obtenus :

Influence du volume de trafic La conformité du simulateur au système étudié est vérifiant en le sou-

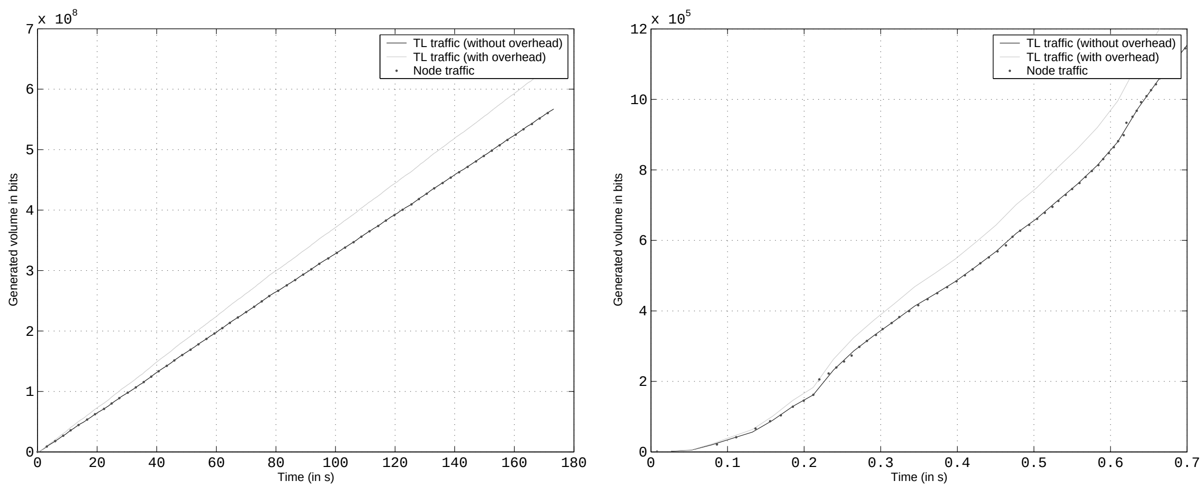


FIGURE 9.19.: Débit du lien satellite link pour un nœud unique

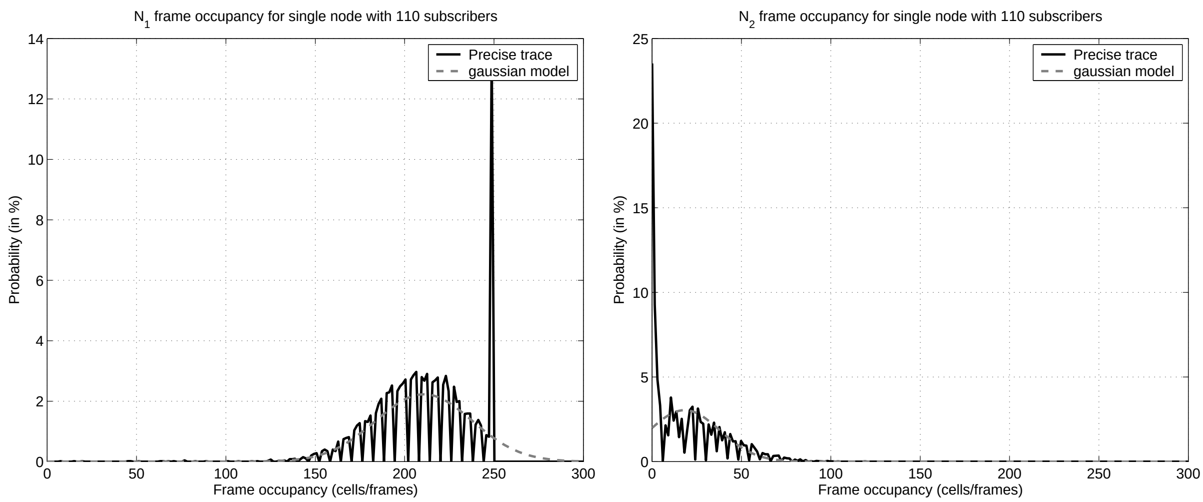


FIGURE 9.20.: PDF de N_1 et N_2 pour un nœud unique

mettant à des charges variables. En augmentant le nombre d'utilisateurs, le volume de trafic augmente. La Figure 9.17 indique la moyenne du nombre de cellules générées pour chaque type de cellule pour un nœud unique. La Figure 9.18 reprend ces distributions lorsque 50 nœuds sont considérés. Dans les deux cas, un comportement linéaire jusqu'à la saturation pour le nombre de cellules de type 1 et un comportement de croissance lente puis de décroissance soudaine dès qu'une valeur seuil est atteinte sont observés.

Effets de la conversion du niveau nœud au niveau transport La Figure 9.19 montre la variation temporelle du taux de transfert vue au niveau nœud et au niveau transport. Le volume au niveau transport est supérieur à cause du transport de signalisation supplémentaire. Un léger décalage temporel du au transfert du niveau nœud au niveau transport est également observé.

La modélisation du système dans le simulateur permet de montrer des effets caractéristiques du comportement du trafic transitant au sein du système.

9.7.5. Justification des équivalences

9.7.5.1. Étude de l'occupation de la trame

La correspondance entre deux modèles en tout d'abord analysée comparant la distribution d'occupation des trames. Ensuite les statistiques de la répartition des types de trames à chaque transmission pour chacune des deux types de cellules sont employées pour prouver l'équivalence des modèles.

Nœud unique La similarité de la distribution des deux types de paquets et pour chacune des modélisation du nœud : type I, III et IV est démontrée. La Figure 9.20 indique chacune des distributions. Elle indique qu'une bonne correspondance des répartitions entre le modèle réel et sa simplification gaussienne est obtenue.

Nœuds multiples La Figure 9.21 montre également la bonne correspondance entre chacun des nombres de trames échangées lorsque l'on considère 50 nœuds lorsque 110 utilisateurs (voix et web) sont considérés. Les moyennes du modèle gaussiens sont en ligne avec ceux qui était obtenu dans le cas précis.

9.7.5.2. Étude du volume de données

La Figure 9.22 montre le volume de trafic généré pour 50 nœuds au cours du temps, lorsqu'on simule un trafic hétérogène. Ceci prouve qu'avec le modèle agrégé ou les modèles précis web et voix, la génération d'un volume équivalent est obtenue. Les effets transitoires (liés à l'initialisation progressive des différentes applications) sont même plus courts lorsque le modèle agrégé est utilisé.

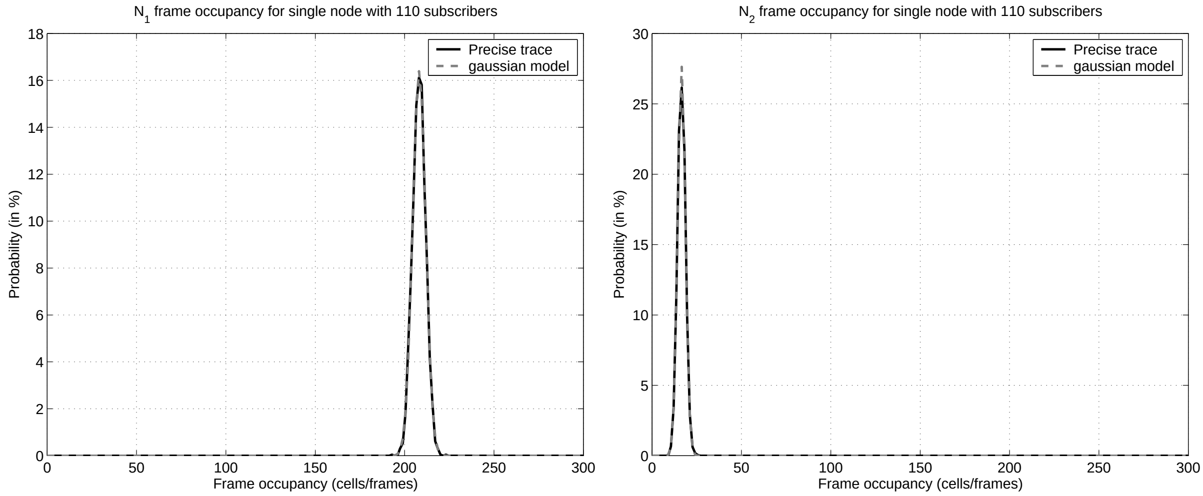


FIGURE 9.21.: PDF de N_1 et N_2 pour des nœuds multiples

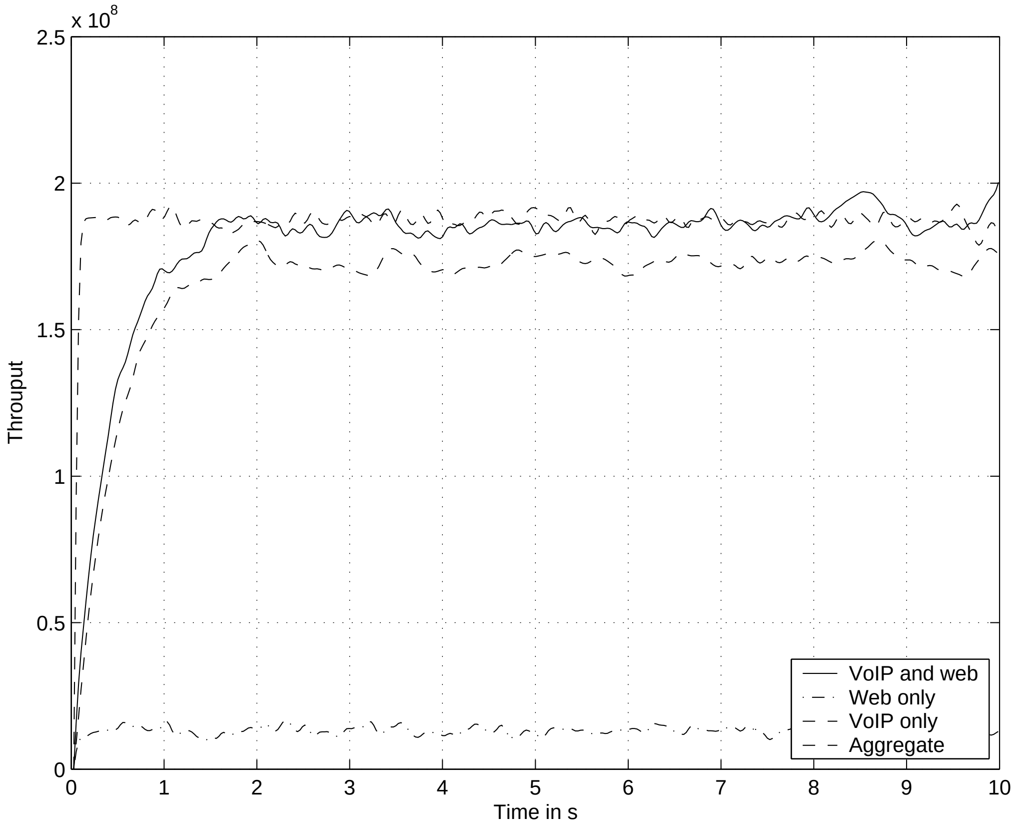


FIGURE 9.22.: Comparaison du débit total pour 50 nœud avec des compositions de trafic différentes

9.7.6. Étude de la durée des sessions web

9.7.6.1. Nœud unique

Dans ce cas, l'évolution de la durée des sessions web lorsque la charge du système augmente est étudiée. Le mécanisme de priorité impose au trafic web une priorité moindre qui se traduit par une augmentation de la durée des sessions. Plus la bande passante du système est grande, moins cette augmentation se produit puisque la limite n'est pas plus atteinte aussi rapidement.

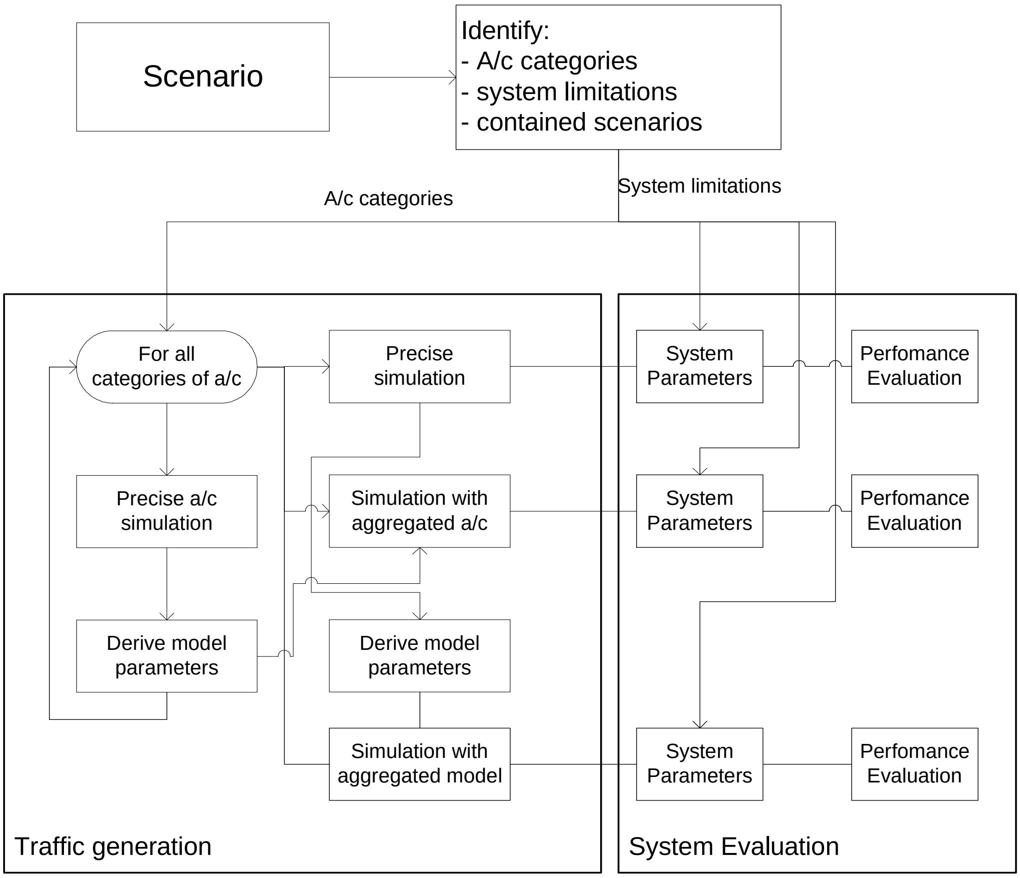


FIGURE 9.23.: Principe des simulations

9.7.6.2. Nœud multiple

Dans ce cas, le comportement est distinct du cas précèdent. En effet, la collectivisation des cellules du type 2 permet d'obtenir un palier durant lequel la durée des sessions n'augmente plus. Lorsque l'allocation sur la trame physique est de 250 trames par cadence, pour un nombre d'utilisateur variant de 100 à 120 utilisateurs, la durée des sessions web reste constante. Au delà, le trafic web est servi de façon insuffisante pour l'utilisateur. En réduisant la capacité totale du système, la plage de fonctionnement correct du système est également réduite (pour une charge de 225 trames, la limite se situe à 107 utilisateurs).

9.7.7. Méthodologie pour le dimensionnement des systèmes

Une méthode pour le dimensionnement d'un système satellite est proposée. La Figure 9.23 indique les différentes étapes. La méthode est basée sur la construction de modèles agrégés pour l'étude d'un système complet. Un modèle de chacun des nœuds du système est construit qui sera ensuite utilisé pour l'étude globale du système. En procédant de la sorte, des simulations précises qui sont coûteuses en ressources peuvent être éviter à l'échelle la plus grande. Une seule étude à grande échelle est requise pour valider l'approche sur les équivalents et peut être distincte de la phase de dimensionnement. Pour la conception d'un système aéronautique comme étudié précédemment, un modèle équivalent de nœud pour chaque type d'avion est construit (ou catégorie significative en terme de nombre de passagers) et ces modèles sont utilisés pour l'étude du niveau transport.

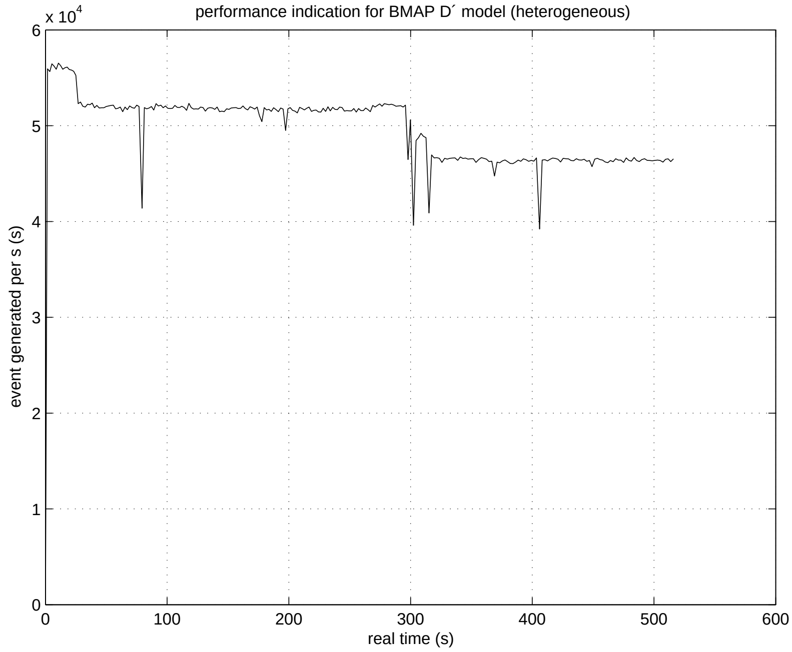


FIGURE 9.24.: Evolution des performances du simulateur (nœud unique)

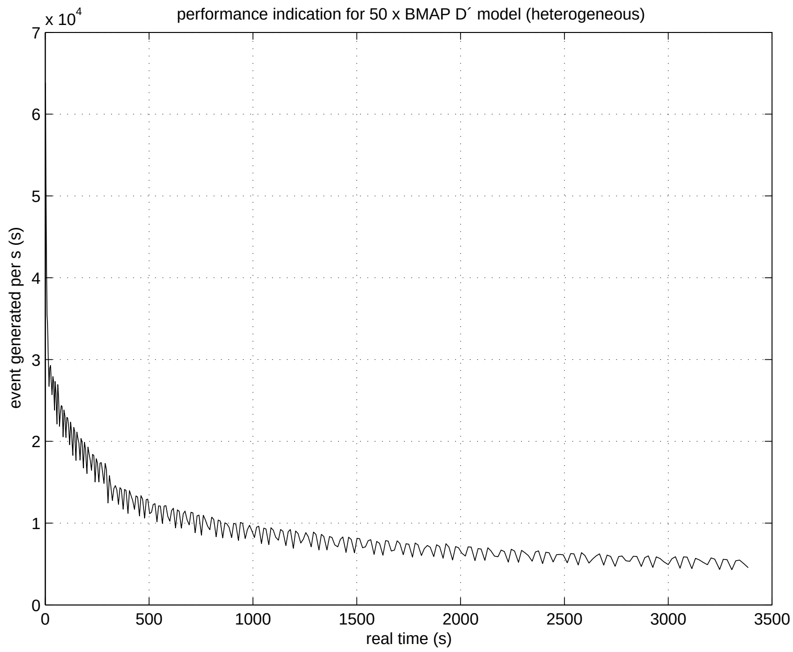


FIGURE 9.25.: Evolution des performances du simulateur (50 nœuds avec la même configuation que le cas précédent)

9.7.8. Analyse de performance

Les performances du simulateur (vitesse d'exécution, nombre d'événements générés...) sont analysées. En particulier les points suivant ont été étudiés

Optimisation du simulateur : La Figure 9.24 montre qu'un taux constant d'événements générés dans le simulateur est obtenu. Les Figures 9.25 et 9.26 montrent un exemple où les réglages d'un module d'observation ont été corrigés afin de ne pas compromettre les performance du simulateur. La chute de performance du premier cas est corrigée dans le second cas, par un ajustement adéquat du taux

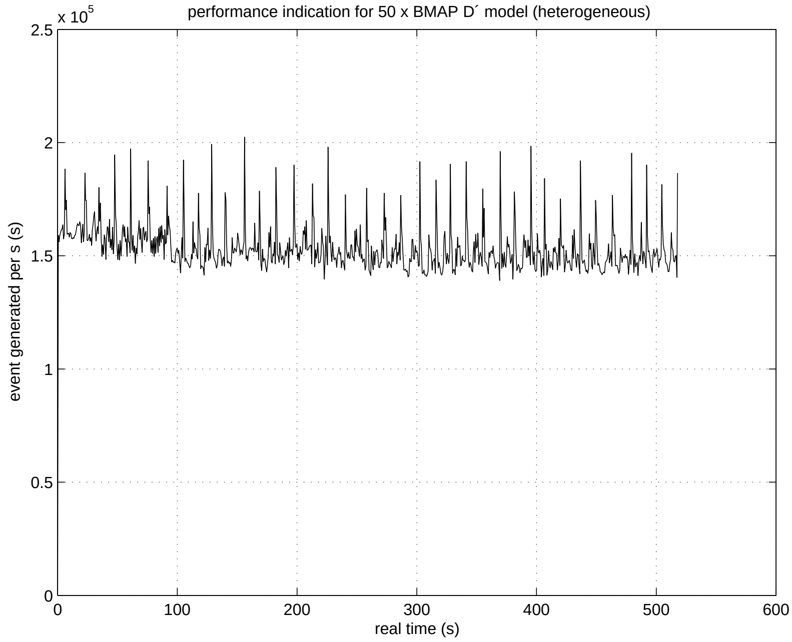


FIGURE 9.26.: Evolution des performances du simulateur (50 nœuds réglages optimisés)

d'observations activées dans le simulateur.

Comparaison des performances : Les performances du simulateur pour différents niveaux de détails sont comparées. Des performances équivalentes sont obtenues en utilisant le modèle agrégé et le modèle précis. Les événements comptabilisés par le simulateur sont de deux types : i) les créations de paquets ii) les états de chaîne de Markov pour le MMPP et BMAP. Les transitions d'états introduites ne font pas chuter les performances du simulateur. Toutefois le trafic du modèle agrégé est injecté à un niveau plus élevé, ce qui permet d'économiser la modélisation de niveau inférieur.

9.8. Conclusion

9.8.1. Résumé

Dans la partie du manuscrit en anglais, les travaux réalisés sont résumés avec davantage de détails. De façon très synthétique, les thèmes suivants ont été abordés :

1. La pyramide de trafic agrégé
2. Les modèles de trafic de la pyramide
3. Description d'un scénario avec la pyramide
4. Résultats par simulation i) équivalences des modèles voix ii) Agrégation au niveau nœud iii) Agrégation complète au niveau transport

9.8.2. Intérêt et pertinence du travail

9.8.2.1. Revue des réalisations et points découverts

Ce travail a permis de construire un cadre abstrait pour la description du trafic organisé selon des principes hiérarchiques. Des modèles de trafic pour la génération de flux de trafic y ont été définis. Des opérateurs ont été introduits pour décrire les modifications subies par ces flux au fil de la transmission.

Différentes mesures ont été décrites pour obtenir les caractéristiques de ces flux de trafic. Par la suite, certains modèles de trafic ont été revus et les différentes méthodes pour la détermination de leurs paramètres ont été mise en œuvre et comparées. Enfin, un simulateur construit sur la base de la pyramide des trafic agrégé et implémentant les modèles de trafic a été utilisé pour collecter les différentes mesures nécessaire pour prouver les équivalences entre divers modèles de trafic.

9.8.2.2. Apport à d'autres projets

Cette thèse s'étant déroulée au DLR, un certain nombre de projets réalisés pour le DLR ont bénéficié de ces travaux. Il s'agit principalement du projet de l'agence spatiale européenne ESA EuroskyWay et du projet de commission européenne WirelessCabin. Pour Euroskyway, dans le cadre de la validation de ce système de communication européen fournissant des services multimédias, le DLR devait concevoir un générateur de trafic reproduisant les données échangées dans charge du satellite. Cet équipement à été conçu de façon hiérarchique et été configuré par les mêmes niveaux que ceux proposés dans le modèle en pyramide. Le cadre de WirelessCabin (fourniture des services voix, accès IP à bord des avions) a donné le cas pratique de notre travail. L'architecture développée pour WirelessCabin comprenait un segment satellite. Les études de capacité de ce segment ont été effectuées en considérant des scénarios issus du modèle en pyramide. Enfin, d'autres projets ont pu bénéficier de nos résultats : il s'agit de ACM Modem (Advanced Coding and Modulation - Codage et modulations avancées) et du projet ULYSS (switching optique ultra rapide) où les modèles implémentés dans le simulateur ont été utilisés.

9.8.3. Perspective pour poursuivre la recherche

Une thèse menée en un temps fini ne peut aborder tous les thèmes et laisse parfois certains champs de coté qui demanderait d'être étudié plus avant. Deux directions pourraient être approfondis.

9.8.3.1. Adaptabilité des modèles

La modélisation du trafic ne s'effectue qu'un instant donné. Si de nouvelles applications naissent et que la capacité des liens augmente, il se peut que le comportement des utilisateurs soit modifié. Par exemple, les applications de partages de (gros) fichiers via lignes ADSL modifient peu à peu la composition du trafic web. Néanmoins, le modèle BMAP semble suffisamment flexible pour qu'on détermine un nouveau jeu de paramètres correspondant davantage aux futures pratiques. Les méthodes exposées dans ce travail reste bien entendu toujours d'application.

9.8.3.2. Simplification des modèles

Nous avons cherché à remplacer un modèle par un modèle équivalent. Une approche complémentaire consisterait également à se doter pour la simplification d'un modèle donné. Le modèle circulant MMPP possède de telles propriétés puisqu'il permet d'obtenir des modèles avec moins d'états ayant un spectre équivalent. Ce type d'étude mériterait un traitement à part entière. Il aurait l'avantage de rester dans une unique famille de modèle. Toutefois, les gains en terme de rapidité d'exécution ne serait que partiels, puisque cette a montré que les principaux gains sont obtenus en utilisant le modèle macroscopique gaussien au niveau transport et demande d'utilisé des modèles très simples.

Appendices

A

Complements on traffic models

The concepts that could not be fully presented in chapter 3 are exposed with more details.

A.1. Renewal models

A.1.1. Points process and renewal process

Definition A.1.1 (Renewal process)**.** *A random point process $\psi = \{T_n\}$ for which the inter-arrival times $\{\mathcal{I}_n\}$ form an i.i.d sequence is called a renewal process. T_n is called the nth renewal epoch and $F(t) = P(\mathcal{I} \leq t)$ denotes the common inter-arrival time distribution. The rate of the renewal process is defined as $\lambda = 1/E(\mathcal{I})$.*

The survival function of a renewal process $\mathfrak{F}$ is defined as the complementary of the distribution function F, $\mathfrak{F}(t) = 1 - F(t)$. As pointed in the historical introduction about queuing, the uniform Poisson process is a currently used point process which is built from an exponential inter-arrivals distribution

$$F(t) = 1 - \exp[-\lambda t] \tag{A.1}$$

The counting process $N(t)$ follows a Poisson law. ($P[N(t) = n] = \frac{(\lambda.t)^n}{n!} \exp[-\lambda t]$)

When a renewal process is considered at a time t, we have

$$T_{N(t)} < t < T_{N(t)+1}$$

Definition A.1.2 (Forward recurrence time)**.** *The forward recurrence time is the time till the next point strictly after time t:*

$$A(t) \triangleq T_{N(t)+1} - t,\ t > 0 \tag{A.2}$$

Similarly, we define

Definition A.1.3 (Backward recurrence time)**.** *The backward recurrence time is the time since the last point before or at time t:*

$$B(t) \triangleq t - T_{N(t)},\ t > 0 \tag{A.3}$$

Furthermore,

Definition A.1.4 (Density of $\mathcal{I}$)**.** *the density $f_{\mathcal{I}}$ is defined by*

$$f_{\mathcal{I}}(t) = \frac{d}{dt}P(t < \mathcal{I} < t + dt) = \frac{dF(t)}{dt} \tag{A.4}$$

Definition A.1.5 (Laplace-Stieltjes Transform (LST))**.** *The Laplace-Stieltjes Transform (LST) $f^*(s)$ of the density f is defined by*

$$f^*(s) = \int_0^\infty e^{-su} f(u)du \tag{A.5}$$

Definition A.1.6 (Renewal function)**.** *The function $H(t)$ defined as*

$$H(t) = \sum_{r=0}^{\infty} rP(N(t) = r) = \mathbb{E}(N(t)) \tag{A.6}$$

and equal to the expectation of $N(t)$ is the renewal function of the renewal process ψ.

In [Cox67] it was shown that:

$$H^*(s) = \frac{f^*(s)}{s(1 - f^*(s))} \tag{A.7}$$

for normal (started at $t = 0$) renewal processes.
For stationary process (started at $t = -\infty$)) for which at $t = 0$ does not necessary correspond to an arrival,

$$H^*(s) = \frac{1}{\mu s^2} \tag{A.8}$$

which shows that $E(N(t)) = t/\mu$

Asymptotic distribution of recurrence time If considered when $t \to \infty$, the limit density of the recurrence time is

$$\frac{\mathfrak{F}(x)}{\mu} \tag{A.9}$$

This can be used to characterize the asymptotic probability density of the process when it is established.

Superposition of Renewal process

Definition A.1.7 (Superposed renewal process)**.** *Given N renewal processes with density $f_i(t),\ i = 1..N$, the superposed process is the counting process where the arrivals correspond to arrival of one of the input renewal processes. In general the superposed process is not a renewal process.*

On figure A.1 the superposition of three renewal processes is depicted. The superposed renewal process is NOT a renewal process. Anyway, the distribution of the time between arrivals can be derived. The density of the interval duration probability is denoted $g(x)$ can be obtained by formula A.15 that is derived hereafter.

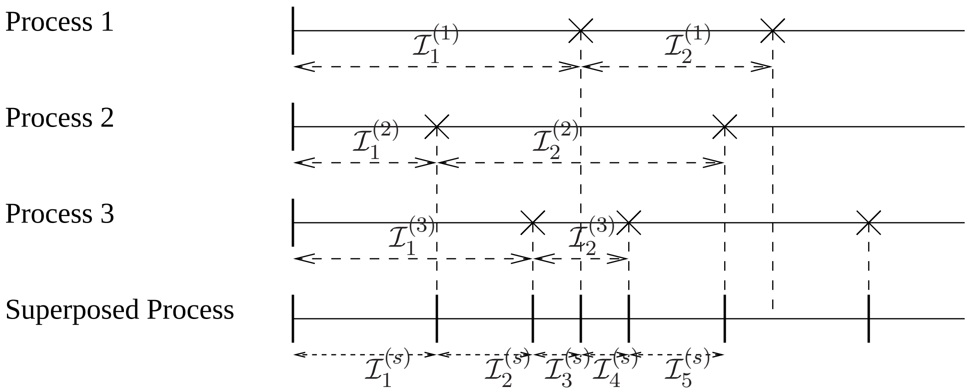


Figure A.1.: The superposed renewal process

Interval between two successive events The calculation is performed using the backward recurrence time. It is noted U for the superposed process.

$$U = min(U_1, U_2, ..U_p) \tag{A.10}$$

By the independence of the single process,

$$P(U > x) = \prod_{i=1}^{p} P(U_i > x) \tag{A.11}$$

$$= \{\int_x^\infty \frac{\mathfrak{F}(u)}{\mu} du\}^p \tag{A.12}$$

So the density of U is obtained by differentiation and is :

$$p\frac{\mathfrak{F}(x)}{\mu}\{\int_x^\infty \frac{\mathfrak{F}(u)}{\mu} du\}^{p-1} \tag{A.13}$$

Since the mean interval between arrival is for the superposed process μ/p. It then comes

$$\frac{\mathfrak{G}}{\mu/p} = p\frac{\mathfrak{F}(x)}{\mu}\{\int_x^\infty \frac{\mathfrak{F}(u)}{\mu} du\}^{p-1} \tag{A.14}$$

with $\mathfrak{G}$ is the survival function of g. It gives finally:

$$g(x) = -\frac{d}{dx}[\mathfrak{F}(x)\{\int_x^\infty \frac{\mathfrak{F}(u)}{\mu} du\}^{p-1}] \tag{A.15}$$

This is the formulation of the inter-arrival density function of the superposition process.

Approximation for the superposition process [Cin72] has showed the following result. The superposition of p independent and identically distributed renewal processes each with rate λ/p tends to be a Poisson process with rate λ as p tends for infinity. Indeed, in the poissonian case, each single poisson process has $\mathfrak{F}(u) = \exp[-\lambda t/p]$ and $\mu = \mathbb{E}[\mathcal{I}] = p/\lambda$. So

$$\begin{aligned}
g(x) &= -\frac{d}{dx}[\mathfrak{F}(x)\{\int_x^\infty \frac{\mathfrak{F}(u)}{\mu} du\}^{p-1}] \text{ using A.15} \\
&= -\frac{d}{dx}[\exp(-\lambda x/p)\{\int_x^\infty \frac{\exp[-\lambda u/p]}{\mu} du\}^{p-1}] \\
&= -\frac{d}{dx}[\exp(-\lambda x/p)\{\frac{1}{\mu}\frac{p}{\lambda}\exp(-\lambda x/p)\}^{p-1}] \\
&= -\frac{d}{dx}[\exp(-\lambda x)] \\
g(x) &= \lambda \exp(-\lambda x)
\end{aligned} \tag{A.16}$$

which shows that the superposition of p poisson processes with rate $\frac{\lambda}{p}$ is a poisson process with rate λ.

Observation of renewal process The following definitions are referred from [CL66].

Definition A.1.8 (Index of Dispersion of Intervals (IDI))**.** *Given a renewal process with inter-arrival sequence* $\{\mathcal{I}_k, k \geq 1\}$*. The sum of* k *consecutive inter-arrival times are noted* $\mathcal{I}^{(k)}$*. The Index of Dispersion of Intervals (IDI) is the sequence* $\{c_k^2, k \geq 1\}$ *defined by*

$$c_k^2 = \frac{k \operatorname{var}(\mathcal{I}^{(k)})}{[\mathbb{E}(\mathcal{I}^{(k)})]^2} = \frac{k \operatorname{cov}(\mathcal{I}_1, \mathcal{I}_1) + 2 \sum_{j=1}^{k-1} (k-j) \operatorname{cov}(\mathcal{I}_1, \mathcal{I}_{1+j})}{k[\mathbb{E}(\mathcal{I}_1)]^2} \tag{A.17}$$

The covariance between $\mathcal{I}_i$ and $\mathcal{I}_j$ is written $\operatorname{cov}(\mathcal{I}_i, \mathcal{I}_j) = \mathbb{E}(\mathcal{I}_i \mathcal{I}_j) - \mathbb{E}(\mathcal{I}_i)\mathbb{E}(\mathcal{I}_j)$.

Definition A.1.9 (Index of Dispersion of Counts (IDC))**.** *Given a renewal process with associated counting function* $N(t)$*. The Index of Dispersion of Counts (IDC) is the function*

$$I(t) = \frac{\operatorname{var}[N(t)]}{\mathbb{E}[N(t)]} \tag{A.18}$$

For a renewal process, the following results are available:
1) For a Poisson process, $I(t) = c_k^2 = 1$ for all t and k.
2) For a renewal process, $c_k^2 = c_1^2$ for all k.
3) If $c_k^2 = c_1^2$ for all k, then $\operatorname{cov}(\mathcal{I}_1, \mathcal{I}_2) = 0$ for all i, j $i \neq j$

When p renewal processes independent and identically distributed are superposed, $c_{k,n}^2$ and $I(t;p)$ are the two indexes of dispersion of the superposed process. We have the following result:

$$I(\infty; p) = \lim_{t \to \infty} I(t;p) = c_{\infty,p}^2 = \lim_{k \to \infty} c_{k,p}^2 = c_{1,1}^2 \tag{A.19}$$

These indexes can be used to characterize some particular points processes (i.e. the poissonian process).

Alternating process

Definition A.1.10 (Alternating renewal process)**.** *Given two renewal probability functions* $f_1(t)$ *and* $f_2(t)$ *the point process verifiying*

$$\frac{d}{dt} P(\mathcal{I}_{2n+1} < t) = f_1(t) \quad \textit{and} \quad \frac{d}{dt} P(\mathcal{I}_{2n} < t) = f_2(t) \tag{A.20}$$

is an alternating renewal process

Definition A.1.11 (Stationary alternating renewal process)**.** *When an alternating process was started at* $t = -\infty$ *and is observed from* $t = 0$*, we have the stationary version of the alternating process.*

For example, if we consider a model with exponentially distributed "ON" durations and exponential distributed "OFF" durations (with mean T_{on} and T_{off}. The mean duration of the ON state can be obtained:

$$\mathbb{E}(\mathbf{T}_{\text{on}}(t) \,|\, S(0) = ON) = \frac{T_{\text{on}} t}{T_{\text{on}} + T_{\text{off}}} + \frac{T_{\text{on}} t}{(T_{\text{on}} + T_{\text{off}})^2} [1 - \exp(-(\frac{1}{T_{\text{on}}} + \frac{1}{T_{\text{on}}})t)] \tag{A.21}$$

Definition A.1.12 (k-state renewal process)**.** *Given a* $k \times k$ *stochastic matrix P where the* $p_{i,j}$ *are the probability to be in state* j *after an arrival in state* i*. Semi-Markov process*

Point process and renewal process (and their derivatives) are good tools to model packet of constant size in an absolute timing reference. If the size of the packets need to be modelled more accurately, other models will be needed.

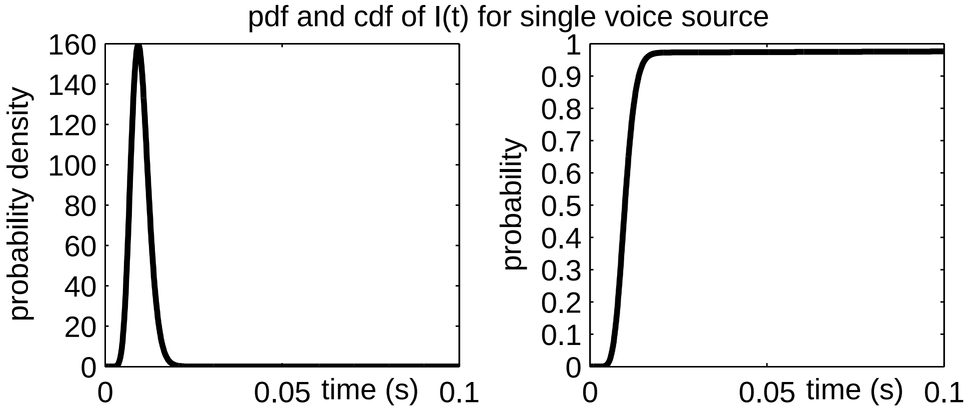


Figure A.2.: Density and CDF of PH-model for a single voice source

A.1.2. The PH-renewal process

Definition (PH-distribution)**.** *A probability distribution $F(.)$ on $[0, \infty]$ is a distribution of phase type (PH-distribution) if it is the distribution of the time until absorption in a finite Markov process of the type defined in section 3.1. The pair (α, T) is called a representation of $F()$.*

This distribution has following properties:

1) the distribution $F(.)$ has a jump of height α_{m+1} at $x = 0$ and a density $F'(x)$ on $[0, \infty]$ such that $F'(x) = \alpha \exp{(Tx)} T^0$
2) the Laplace-Stieltjes transform $f(s)$ of $F(.)$ is given by

$$f(s) = \alpha_{m+1} + \boldsymbol{\alpha}(sI - T)^{(-1)} \boldsymbol{T}^0, \text{ for } Res \geq 0 \tag{A.22}$$

3) the non central moments μ'_i of $F(.)$ all finite and given by:

$$\mu'_i = (-1)^i i! (\boldsymbol{\alpha} T^{-i} \mathbf{e}), \text{ for } i \geq 0 \tag{A.23}$$

Examples:

1) the generalized Erlang distribution of order m with parameters $\lambda_1, \ldots, \lambda_m$ has the representation $\boldsymbol{\alpha} = (1, 0, \ldots, 0)$ and

$$T = \begin{pmatrix} -\lambda_1 & \lambda_1 & 0 & \ldots & & 0 \\ 0 & -\lambda_2 & \lambda_2 & 0 & & 0 \\ \vdots & \ddots & & \ldots & & \\ 0 & & & 0 & -\lambda_{m-1} & \lambda_{m-1} \\ 0 & & \cdots & & 0 & -\lambda_m \end{pmatrix} \tag{A.24}$$

2) the hyperexponential distribution with distribution $F(x) = \sum_{i=1}^{m} \alpha_i (1 - \exp{(-\lambda_i x)})$ has the representation $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ and $T = -\text{diag}(\lambda_1, \ldots, \lambda_m)$.
3) the IAT for the voice model can be approximated with a distribution (γ, L) from the convolution of the two PH-distribution (α, T) and (β, S) where (α, T) is a 16-state Erlang with $\lambda = 16/T_{res}$ approximating a constant delay distribution and (β, S) is the exponential silence duration. The calculation of (γ, L) is performed following theorem 2.2.2 from [Neu81].

From the $m + 1$-state Markov process used to define the PH-distribution, another process can be obtained by restarting instantaneously the process with the probabilities $\alpha_1, \ldots, \alpha_m, \alpha_{m+1}$. The state $m + 1$ is an instantaneous state. The corresponding process on the states $1, \ldots, m$ has an infinitesimal generator Q^*.

Definition A.1.13 (PH-renewal Process of canonic type)**.** *The Markov process with infinitesimal generator*

$$Q^* = T + T^0 A^0, \tag{A.25}$$

where T^0 is the $m \times m$ matrix with identical columns $\boldsymbol{T}^0$ and $A^0 = \frac{1}{1-\alpha_{m+1}} diag(\alpha_1, \ldots, \alpha_m)$ is a Markov process where the successive visits to the instantaneous state $m+1$ form a renewal process with underlying distribution $F(.)$ as in (3.2). The point process is a PH-renewal process.

[Neu81] has shown in theorem (2.4.1) that the expected number of renewal $H(t) = \mathbb{E}(N(t))$ in $(0, t]$ is :

$$H(t) = \frac{t}{\mu_1'} - \frac{1}{1-\alpha_{m+1}} + \frac{\sigma^2 + {\mu_1'}^2}{2{\mu_1'}^2} + \frac{\boldsymbol{\alpha}[\Pi - \exp{(Qt)}]T^{-1}\mathbf{e}}{\mu_1'(1-\alpha_{m+1})} \tag{A.26}$$

The class of PH-process can be used to model different probability density functions. For example, it has been used by [FW98] to approximate a Weibull distribution with an hyperexponential H_{20} to model a service distribution. Moreover, the class has been extended by Schewel([Sch01]) to include a Pareto shape, to that the modelling of TCP connections can be performed with these distributions.

A.1.3. Markov renewal processes

semi-Markov matrices A non decreasing, right continuous point function F on the real line with $F(\infty) \leq 1$ is called a mass function. Such a function induces a measure on the Borel sets of R, an interval $(a, b]$ carries the measure F(b)-F(a).

Let J be a subset of the set of non-negative integers.

Definition A.1.14 (semi-Markov matrix)**.** *Let A be a matrix of mass functions $A_{jk}(j, k \in J)$, and put $B_j = \sum_k A_{jk}$. Then A is called a semi-Markov matrix over J if, for all $j \in J$,*

$$B_j(0^-) = 0, \quad B_j(+\infty) \leq 1$$

Definition A.1.15 (Markov matrix)**.** *A Markov matrix over J is a set of numbers $\tilde{A_{jk}}$ defined for each $j, k \in J$ such that $\tilde{A_{jk}} \geq 0$ and $\sum_k \tilde{A_{jk}} \leq 1$.*

If A is a semi-Markov matrix, then A(t) is a markov matrix for any fixed $t \in [0, \infty)$. In particular, $\tilde{A} = A(\infty)$ is a Markov matrix, and is called the associated Markov matrix of A. Let

$$A_{jk}^{(0)}(t) = \begin{cases} 0 & t < 0, \\ \delta_{jk} & t \geq 0 \end{cases} \tag{A.27}$$

$$A_{jk}^{(n)}(t) = \begin{cases} 0 & t < 0, \\ \sum\limits_{\nu} \int\limits_{0^-}^{t} A_{j\nu}(dy) A_{\nu k}^{(n-1)}(t-y) & t \geq 0 \end{cases} \tag{A.28}$$

We have

$$A^{(0)} = \mathbf{I} \tag{A.29}$$

$$A^{(n)} = \int_{0^-}^{t} A(dy) A^{(n-1)}(t-y), \quad n = 1, 2, ..., \tag{A.30}$$

Markov renewal process We let as before J be a subset of the non-negative integers and suppose A is a semi-Markov matrix defined on J. Let $(\Omega, \mathcal{F}, P)$ be a probability space and suppose we have, defined on it,

a) a function $L(\omega)\, \omega \in \Omega$ taking value in $\{1, 2, ..., \infty\}$,
b) functions $X_n(\omega)$ defined for $0 \leq n < L(\omega)$, $\omega \in \Omega$, and taking value in J,
c) functions $T_n(\omega)$ defined for $0 \leq n < L(\omega)$, $\omega \in \Omega$ taking values in $[0, \infty)$ such that for any $\omega \in \Omega$ we have $0 = T_0(\omega) \leq T_1(\omega) \leq ....$ Suppose for each $n \geq 0$, that $mathcalF$ contains the σ-algebra generated by the sets $\{X_m = k, T_m \leq t, L > n\}(m = 0, ..., n; k \in J; t \in [0, \infty))$.

Definition A.1.16 (Markov renewal process)**.** $\{X_n, T_n; L\}$ *is called a Markov renewal process induced by the semi-Markov matrix* $A(t)$ *if*

$$P\{X_{n+1} = j, T_{n+1} \leq t \,|\, X_0, ..., X_n; T_0, ..., T_n\} \tag{A.31}$$
$$= P\{X_{n+1} = j, T_{n+1} \leq t \,|\, X_n, T_n\} = A_{X_n j}(t - T_n) \tag{A.32}$$

almost everywhere on $\{L > n\}$ *for any* $j \in J$

The Markov renewal process can be related to some particular Markov chains.

Proposition A.1.17. . *The process* $\{X_n, T_n; L\}$ *is a special case of 2 dimensional Markov chain with state space* $J \times [0, \infty)$. *We have* $T_0 = 0$ *and*

$$P\{X_n = k, T_n \leq t \,|\, X_0 = j\} = A^{(n)}_{jk}(t)$$

with $A^{(n)}_{jk}(t)$ are defined as in (A.27) and (A.28).

Exact superposition The theoretical superposition is exposed in the article of [CD83] which requires the introduction of a Markov renewal process on a space $E = J \times R^+$ where J is on countable set (as N or N^d). We assume we have the following random variables S_n and T_n on E and R^+ which represent the system state and the corresponding transition time. A σ-algebra Ξ is associated to E.

Definition A.1.18. *The stochastic process* $\{S_n, T_n; n \geq 0\}$ *is a Markov renewal process induced by the kernel Q if*

$$\begin{aligned} &Pr[S_{n+1} \in A, T_{n+1} \leq t \,|\, S_0, ..., S_n, T_0, ..., T_n] \\ &\qquad = Pr[S_{n+1} \in A, T_{n+1} \leq t \,|\, S_n, T_n] = Q(A \times [0, t - T_n] \,|\, S_n) \end{aligned}$$

where $A \in \Xi, t \in [0, \infty)$ *and* $n \geq 0$.

The kernel Q is an application from $(\Xi \otimes R^+) \times E$ into $[0, 1]$.

Proposition A.1.19. *Sojourn times: Let* $X_0 = T_0 = 0$ *and* $X_n = T_n - T_{n-1}$, $n \geq 1$. *Then given* $n_1, ..., n_k$ *and* $t_1, ..., t_k$, *we have*

$$Pr[X_{n_1} \leq t_1, ..., X_{n_k} \leq t_k \,|\, S_n; n \geq 0] = \prod_{i=1}^{k} Pr[X_{n_i} \leq t_i \,|\, S_{n_i - 1}, S_{n_i}]$$

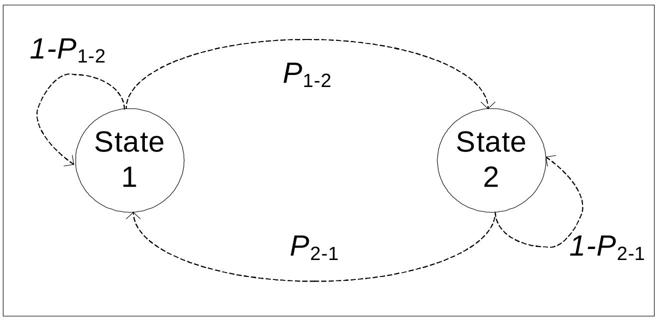
that is the sojourn times $\{X_{n_i}, i = 1...k\}$ *are conditionally independent given* $\{S_{n_i - 1}, S_{n_i}, i = 1...k\}$.

Proposition A.1.20. *Underlying Markov chain: The stochastic process* $\{S_n, n \geq 0\}$ *is a time homogeneous Markov chain defined on* (E, Ξ) *with transition probability given by*

$$Pr[S_{n+1} \in A \,|\, S_n = y] = \lim_{t \to \infty} Q(A \times [0, t] \,|\, y) = Q(A \times [0, \infty) \,|\, y)$$

.

This procedure has the drawback that the number of states N^d required to describe the superposition is exploding quite fast, so that such processes are difficult to be used in practice.



2 States Markov Chain

Figure A.3.: Example of a two state Markov Chain

A.2. State based models

A.2.1. Mathematical foundation

A.2.1.1. Markov Chains

Given an integer m and a $m \times m$ matrix stochastic P ($\forall i, \sum_{j=1}^{m} P_{ij} = 1$). The discrete process $X_n = i$ on the states $1 \le i \le m$ is said to be a Markov chain if the probability

$$\mathbb{P}(X_n = i \,|\, X_1, ..., X_{n-1}) = \mathbb{P}(X_n = i \,|\, X_{n-1}) \tag{A.33}$$

and moreover

$$\mathbb{P}(X_n = i \,|\, X_{n-1} = k) = P_{ki} \tag{A.34}$$

Given a discrete-time finite state stationary Markov chain with transition matrix P, a state j is accessible from state i if there is a sequence of transitions from i to j that has non zero probability. The probability of being in state j after the k-th transition, given that the initial state was i is given by $(P^k)_{ij}$. Two states i and j communicate if they are accessible from each other (one state always communicates with itself).

Two states are said to belong to the same ergodic class if they communicate with each other. If the state space by itself forms an ergodic class, the Markov chain is called irreducible. (differently said theFinally, the state space of the Markov chain is a communicating class; this means that, in an irreducible Markov chain, it is possible to get to any state from any state). Otherwise it is called reducible. A state is called transient if, given that we start in state i, there is a non-zero probability that we will never return back to i.

Among reducible Markov chains, two types are distinguished: periodic and aperiodic ones. The period of a Markov chain is concerned with the times at which the chain might return to s state from which it started. If this can only happen at times that are multiples of d, where d is the largest integer with

The stationary probability vector $\mathbf{p}$ is a vector such as $\mathbf{p}P = \mathbf{p}$ and $\mathbf{p}\mathbf{e} = 1$. An irreducible chain has a stationary distribution if and only if all of its states are positive-recurrent. In that case, $\mathbf{p}$ is unique and corresponds to the distribution that will be reached over the states at the limit.

$$\lim_{n\to\infty} P(X_n = i) = \mathbf{p}$$

If the original chain state is chosen according to $\mathbf{p}$, the chain remains in its stationary distribution.

For a Markov chain P, such that $P(X(t+1) = j|X(t) = i) = P_{ij}$, P is stochastic. If P is regular, there is an unique invariant distribution π verifying $P\pi = \pi$. More over $\mathbf{e}P = \mathbf{e}$

Figure A.3 shows an example of a 2 states Markov chain. The corresponding transition matrix is:

$$P = \begin{bmatrix} 1 - P_{1-2} & P_{1-2} \\ P_{2-1} & 1 - P_{2-1} \end{bmatrix}$$

A.2.1.2. Continuous time Markov Process

Markov process are the time continuous version of Markov chains. They are defined by a matrix Q, playing a similar role as the P matrix but such as $Q_{ij} \geq 0$,$Q_{ii} < 0$ $Qii = -\sum_{j \neq j} Qij$

In probability theory, a continuous-time Markov process is a stochastic process $X(t)$ $t > 0$ that satisfies the Markov property and takes values from a set called the state space. The Markov property states that at any times $s > t > 0$, the conditional probability distribution of the process at time s given the whole history of the process up to and including time t, depends only on the state of the process at time t. In effect, the state of the process at time s is conditionally independent of the history of the process before time t, given the state of the process at time t. The process follows the strong Markov property:

$$\mathbb{P}(X_s = i \,|\, X_t, t \in [0, s]) = \mathbb{P}(X_s = i \,|\, X_t) \tag{A.35}$$

Definition Let $X(t)$ be the random variable describing the state of the process at time t. Now prescribe that in some small increment of time from t to $t + h$, the probability that the process makes a transition to some state j, given that it started in some state $i \neq j$ at time t, is given by

$$\mathbb{P}(X(t+h) = j|X(t) = i) = q_{ij}h + o(h),$$

where $o(h)$ represents a quantity that goes to zero faster than h as h goes to zero. Hence, over a sufficiently small interval of time, the probability of a particular transition is roughly proportional to the duration of that interval.

Continuous-time Markov processes are most easily defined by specifying the transition rates qij, and these are typically given as the ij-th elements of the transition rate matrix, Q (sometimes called a Q-matrix by convention). Q is a finite matrix according to whether or not the state space of the process is finite (it may be countably infinite, for example in a Poisson process where the state space is the non-negative integers). The most intuitive continuous-time Markov processes have Q-matrices that are:

- conservative: the i-th diagonal element q_{ii} of Q is given by $q_{ii} = -q_i = -\sum_{j \neq i} q_{ij}$,
- stable: if for any given state i, all elements q_{ij} (and q_{ii}) are finite.

In this case, we have

$$\mathbb{P}(X_{t+dt} = j \,|\, X_t = i) = Qij;\ i \neq j$$

and

$$\mathbb{P}(X_{t+dt} = i \,|\, X_t = i) = 1 + Qii;\ i \neq j$$

(Note, however, that a Q-matrix may be non-conservative, unstable or both.) When the Q-matrix is both stable and conservative, the probability that no transition happens in some time r is

$$\mathbb{P}(X(s) = i \,\forall\, s \in (t, t+r] \,|\, X(t) = i) = e^{-q_i r}.$$

That is, the probability distribution of the waiting time until the first transition is an exponential distribution with rate parameter qi (= -qii), and continuous-time Markov processes are thus memoryless processes.

The stationary probability distribution, π, of a continuous-time Markov process, Q, may (subject to some important technical assumptions) be found from the property

$$\pi Q = 0.$$

Note that

$$\pi.\mathbf{e} = 1,$$

where $\mathbf{e}$ is a column matrix with all elements consisting of 1's.

Generation Given a Markov Process X_t of infinitesimal generator matrix Q, for $\lambda \geq \max(-Q_{ii})$ there exist a Markov chain Y_t with a matrix $P = Q + I/\lambda$ and a Poisson process N_t such that

$$X_t = Y_{N_t}$$

By generation of the time line using the events of the Poisson process $N_t$and of the state using the transition of the Markov chain P, the Markov process can be generated.

A.2.1.3. Useful results

We summarize hereafter some results related to eigenvalues of Matrix having the form described previously (for Markov chains and processes).

Perron-Frobenius theorem

Theorem A.2.1. *Given a non negative matrix A there is an eigenvalue λ_{PF} that is real and non negative with associated non negative left and right eigenvectors. For any other eigenvalue λ of A, $|\lambda| \leq \lambda_{PF}$.*

Moreover, if A is regular ($A^k > 0$ for some k) the eigenvalue λ_{PF} has multiplicity one and the left and right eigenvectors are positive and unique (up to positive scaling).

Let T be a real $n \times n$ matrix with non negative entries.

Definition A.2.2 (Irreducible matrix)**.** *A matrix M is called irreducible if for all i, j there is a k such that $(T^k)_{ij} \geq 0$)*

Theorem A.2.3 (Perron Frobenius)**.** *Let $M \geq 0$ be an irreducible matrix. There exist a unique positive real number η with the following properties*

1. *There is a real vector $\boldsymbol{v_0} > 0$ verifying $M\boldsymbol{v_0} = \eta\boldsymbol{v_0}$*
2. *$\boldsymbol{v_0}$ has geometric and algebraic multiplicity one.*
3. *For each eigenvalue λ of M, we have $|\lambda| \leq \eta$*
   $\boldsymbol{u_0}\boldsymbol{v_0} = \boldsymbol{u_0}\boldsymbol{e} = 1$, where $\boldsymbol{u_0}$ is a left eigenvector of M.

Asymptotic calculation of matrix exponential The following results are used to compute the limits for the exponential of a matrix.

$$exp(Mt) = [v_0 u_0]\exp(-\eta t) + o(\exp(-\eta t)) \tag{A.36}$$

where v_0 and u_0 are the Perron-Frobenius vector introduced previously.

Spectrum If Q is the generator matrix of a Markov-Process with N states. Q can be decomposed using following formula:

$$Q = \sum_{i=0}^{N-1} \Psi_i g_i h_i \tag{A.37}$$

where Ψ_i is the eigenvalue of Q and g_i a right-eigenvector of Q and h_i a left-eigenvector of Q for the value Ψ_i

In addition to these results, [Bel70] summaries a lot of useful results on matrices.

A.2.2. Batch Markovian Arrival Process

Definition (Batch Markovian Arrival Process)**.** *A 2-dimensional Markov process (Continuous Time Markov Process) $\{N(t), J(t)\}$ on the state space $\{(i,j)\ : i \geq 0, 1 \leq j \leq m\}$ with the infinitesimal generator Q having the following structure*

$$Q = \begin{bmatrix} D_0 & D_1 & D_2 & D_3 & \cdots \\ & D_0 & D_1 & D_2 & \cdots \\ & & D_0 & D_1 & \cdots \\ & & & . & \cdots \\ & & & & \cdots \end{bmatrix} \tag{A.38}$$

$N(t)$ is the counting process and $J(t)$ is the phase of the process. It is required that D_0 has negative diagonal elements and non-negative off-diagonal elements, D_k are non-negative, and that $D = \sum_{k=0}^{\infty} D_k$ is an irreducible infinitesimal generator.

Definition A.2.4 (BMAP matrix generating function)**.**

$$D(z) \triangleq \sum_{k=0}^{\infty} D_k z^k, \quad \text{for} \quad |z| \leq 1 \tag{A.39}$$

The stationary probability vector $\boldsymbol{\pi}$ of the Markov process with generator D verifies

$$\boldsymbol{\pi} D = \mathbf{0} \qquad\qquad \boldsymbol{\pi}\mathbf{e} = 1 \tag{A.40}$$

D_0 governs the transitions without arrivals, D_j transitions with batch of size j.

In this case the arrival rate is

$$\lambda_1^{-1} = \pi \sum_{k=1}^{\infty} k D_k \mathbf{e} \tag{A.41}$$

Interpretation useful for implementation of BMAP model. In each state, the diagonal term of D_0 (D_{0ii}) is the mean sojourn time in the state i, $1 \leq i \leq m$. The sojourn time in state i is exponentially distributed with mean λ_i. At the end of it, there occurs a transition to another (or possibly the same) state and this transition is potentially corresponding to an arrival. The non-diagonal terms of D_0 can be used to form the probability $p_i(0,k)$ with $1 \leq k \leq m$, $k \neq i$ that there is a transition to state k without arrivals. For $j \geq 1$, arrivals of batches of size j are considered with the probability $p_i(j,k)$ of a transition to state k with a batch arrival of size j. The relationship between the probability and the matrix is:

$$\lambda_i = -(D_0)_{ii} \qquad p_i(0,k) = \frac{(D_0)_{ik}}{\lambda_i},\ k \neq i \qquad p_i(j,k) = \frac{(D_j)_{ik}}{\lambda_i} \tag{A.42}$$

Since the matrix D is stochastic we have:

$$\sum_{k=1, k\neq i}^{m} p_i(0,k) + \sum_{j=1}^{\infty}\sum_{k=1}^{m} p_i(j,k) = 1 \tag{A.43}$$

The counting function Let N_t be the number of arrivals in $(0,t]$ and J_t the state of the Markov process at time t. Let

$$P_{ij}(n,t) = \Pr\{N_t = n, J_t = k \,|\, N_0 = 0, J_0 = i\} \tag{A.44}$$

be the (i,j) entry of a matrix $P(n,t)$. The matric generating function $P^*(z,t) = \sum_{n=0}^{\infty} P(n,t) z^n$ verifies

$$P^*(z,t) = e^{D(z)t} \tag{A.45}$$

A.2.3. The Markov Modulated Poisson Process

Definition The MMPP is a particular case of BMAP with an infinitesimal generator Q and arrival rate matrix $\Lambda = diag(\lambda_1, ..., \lambda_m)$. The parameters characterizing an MMPP are a m-state continuous Markov chain with infinitesimal generator Q and the m-Poisson arrival rates $\lambda_1, ..., \lambda_m$. All arrivals have the same size (no batch arrivals). In the notations from the BMAP we then have :

$$D_0 = Q - \Lambda \qquad D_1 = \Lambda \qquad D_k = 0,\ k \geq 2 \tag{A.46}$$

The stationary probability vector $\boldsymbol{\pi}$ of the Markov process with generator Q verifies

$$\boldsymbol{\pi} Q = \mathbf{0} \qquad \boldsymbol{\pi}\mathbf{e} = 1 \tag{A.47}$$

The state of the underlying chain is $J(t)$ at t. If the MMPP is not a renewal process, the sequence $\{(J_n, X_n), n \geq 0\}$ where J_0 is the state of $J(t)$ at $t = 0$, $X_0 = 0$ and for the kth arrivals J_k is the state of the underlying process and X_k is the time between the $k-1$th and kth arrival is Markov renewal sequence with the following transition probability:

$$\begin{aligned} F(x) &= \int_0^x \exp\left[(Q-\Lambda)u\right] du\, \Lambda \\ &= \{I - e^{(Q-\Lambda)x}\}(Q-\Lambda)^{-1}\Lambda \end{aligned}$$

The element of the matrix F_{ij} are the conditional probabilities $\Pr\{J_k = j, X_k \leq x \mid J_{k-1} = i\}$.

The matrix $F(\infty) = (\Lambda - Q)^{-1}\Lambda$ is stochastic and is the transition probability of the Markov chain embedded at arrivals epochs.

The counting function and rate related results Let N_t be the number of arrivals in $(0, t]$ and J_t the state of the Markov process at time t. Let

$$P_{ij}(n,t) = \Pr\{N_t = n, J_t = k \mid N_0 = 0, J_0 = i\} \tag{A.48}$$

be the (i,j) entry of a matrix $P(n,t)$. The matric generating function $P^*(z,t) = \sum_{n=0}^{\infty} P(n,t)z^n$ verifies

$$P^*(z,t) = e^{(Q-(1-z)\Lambda)t} \tag{A.49}$$

The arrival rate can be obtained from

$$\lambda_1^{-1} = \boldsymbol{\pi}\Lambda\mathbf{e} \tag{A.50}$$

The autocovariance function of the MMPP can be expressed as

$$C[k] = \mathbf{p}(\Lambda - Q)^{-2}\Lambda\{F(\infty)^k - \mathbf{e}.\mathbf{p}\}(\Lambda - Q)^{-2}\Lambda\mathbf{e}, \tag{A.51}$$

where

$$F(\infty) = (\Lambda - Q)^{-1}\Lambda \tag{A.52}$$

and $\mathbf{p}$ is the steady-state vector of $F(\infty)$. In the MMPP(2) case, we have:

$$\sigma = \frac{\lambda_1.\lambda_2}{\lambda_1.\lambda_2 + \lambda_1.r_2 + \lambda_2.r_1} \tag{A.53}$$

Even if MMPP models are just modelling one size of packets, these models form a compact class that don't have too many parameters and made them suitable for different method for model parameter derivation.

A.2.4. Circulant MMPP

A.2.4.1. Definition

A sub-category of MMPP traffic models can be introduced by considering the family of models where the $N \times N$ matrix Q has the first raw of the form

$$\mathbf{a} = [a_o\, a_1\, \ldots\, a_{N-1}]$$

with $a_0 = -\sum_{k=1}^{N-1} a_k$ and the other rows are obtained by circulating the raw of one element to the right. The matrix Q is can be written with the $N \times N$ permutation matrix P:

$$P = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & \\ \vdots & & & \ddots & \\ 0 & \cdots & & 0 & 1 \\ 1 & 0 & \ldots & 0 & 0 \end{pmatrix} \quad Q = \begin{pmatrix} a_0 & a_1 & a_2 & \cdots & a_{N-1} \\ a_{N-1} & a_0 & a_1 & \cdots & a_{N-2} \\ \vdots & a_{N-1} & \ddots & \ddots & \\ a_2 & \cdots & & a_0 & a_1 \\ a_1 & & \ldots & a_{N-1} & a_0 \end{pmatrix}$$

Using this P matrix, Q can be written

$$Q = \sum_{k=0}^{N-1} a_k P^k \tag{A.54}$$

where P is the permutation matrix. Q is then called a circulant chain.

The eigenvalues of P are the $\mathrm{N^{th}}$ roots of unity and are obtained from $w_j = e^{2\pi j i/N}$ $j = 1, .., N$, the eigenvalues of P^k are so $w_j^k = e^{2\pi jki/N}$ $j = 1, .., N$. Using A.54, the eigenvalues of Q are derived as $\psi_j = \sum_{k=0}^{N-1} a_k w^{jk}$.

The eigenvectors are common to P and Q and obtained from the $N \times N$ Fourier matrix

$$F = \frac{1}{\sqrt{N}}\left[w^{-jk}\right]$$

The matrix F does not depend of $\mathbf{a}$, Q can be called an eigenvalue only matrix. Its eigenvalues can be retrieved as

$$\overrightarrow{\psi} = \sqrt{N}\mathbf{a}F^\star$$

A.2.4.2. Autocorrelation and Power density

When stochastic process are involved (by stochastic process, is meant a real value random variable which realization is $X_t(w)$ where w is random) , a correlation function is defined by

$$R(\tau, t) = \mathbb{E}[X(t+\tau)X(t)]$$

and reduced to a continuous function for stationary stochastic processes. When a time series is considered, the correlation function is similarly defined for $R(n, m) = \mathbb{E}[X(n+m)X(n)] = R(m)$

This notion needs to be extended for point process. If the series of the arrival instants is known, the corresponding rate function $a(t)$ can be considered. The autocorrelation of the rate function if then

$$R(\tau) = \mathbb{E}[a(t)a(t+\tau)] \tag{A.55}$$

In the case of circulant MMPP, the rate function is described by Q the state transition matrix and $\overrightarrow{\gamma}$ the rates vector. As for other MMPP, the steady state vector is called π and verifies $\pi Q = 0$ and $\pi\mathbf{e} = 1$. The average input rate is defined by $\bar{\gamma} = \pi\overrightarrow{\gamma}$

if Q is diagonalizable, we get

$$R(\tau) = \bar{\gamma}\delta(0) + \Psi_0 + \sum_{l=1}^{N-1} |\Psi_l| e^{Re(\psi_l)|\tau|} \cos(Im(\psi_l)|\tau| + \arg\Psi_l) \tag{A.56}$$

with

$$\Psi_l = \sum_i \sum_j \pi_i \gamma_i \gamma_j g_{l_i} h_{l_j}$$

For the circulant MMPP,

$$R(\tau) = \sum_{l=0}^{M} \psi_l e^{\lambda_l |\tau|} \tag{A.57}$$

and

$$\Psi_l = \frac{1}{N^2} \{ \sum_i \gamma_i w^{li} \} \{ \sum_j \gamma_j w^{-lj} \}$$

That can also be written in matrix form $\overrightarrow{\Psi} = \frac{1}{N} | \overrightarrow{\gamma} F^* |^2$

Using the Wiener-Khintchin theorem (as exposed in [Lac00]), the power density $P_\lambda(\omega)$ is the Fourier transform of $R_\lambda(\tau)$

$$P_\Lambda(\omega) = \int e^{-2\pi i \omega \tau} R(\tau) d\tau = 2\pi \bar{\gamma}^2 \delta(\omega) + \gamma + \sum_{i=0}^{N-1} \frac{-2\Psi_i \psi_i}{\psi_i^2 + \omega^2}$$

where $\delta(\omega)$ is the Dirac distribution. The term $2\pi\lambda^2\delta(\omega)$ represents the constant mean arrival rate and the term λ represents the white noise created in each phase of the Poisson process.

$$E(X_1) = -\Phi C^{-1} \mathbf{e}$$

where $\Phi = \frac{1}{\lambda_l} \pi D_l$

$$E(X_1 X_k) = -\Phi C^{-1} (-C^{-1} D)^{-k} C^{-1} \mathbf{e}$$

$$\rho(k) = \frac{\Phi C^{-1} (I - \mathbf{e}\Phi)(-C^{-1} D)^k C^{-1} \mathbf{e}}{\Phi GgGgC^{-1}(2I - \mathbf{e}\Phi) C^{-1} e}$$

The circulant MMPP have tractable properties for the calculation the power spectrum. The properties of additivity of the power spectrum for aggregated model is an appealing property. Practical problems were encountered for the implementation of parameters derivation procedure (the constraint minimization procedure, to be used was giving incoherent results), so that these models were not used in the numerical investigation. The order of the model to have circulant form is in the range of 50, 100 (with many nul parameters), so that the handling of the parameters is a bit more difficult than for the MMPP(2).

B

Appendix - Measurements from a stream

In the Chapter 2, the concept of stream has been introduced. The definition of a stream is recalled here.

Definition (Abstract unit in the absolute time reference)**.** *A pair* $(\mathcal{S}, \mathcal{I})$ *is defined for every new incoming packet.* $\mathcal{S}$ *is the size of the packet and* $\mathcal{I}$ *is the inter-arrival time i.e. the difference between the arrival time of the packet and the arrival time of the previous one.*

Definition (Abstract unit in the slotted time reference)**.** *Considering a given time slot duration* T_s*, a pair* $(\mathcal{S}, \mathcal{I})$ *is required for every* n *describing the traffic seen in the time slot* $[nT_s, (n+1)T_s]$*.* $\mathcal{S}$ *is the overall size of the packets in the time slot and* $\mathcal{I}$ *is the position in the time slot of the first packet. If the time slot is empty,* $\mathcal{S} = 0$ *and the value of* $\mathcal{I}$ *has not importance.*

Definition (Abstract unit stream)**.** *A sequence of abstract units* $\Phi = \{\mathfrak{A}_i, i \in \mathbb{N}\}$ *is an abstract unit stream*

Definition (Absolute abstract unit stream)**.** *A sequence of absolute abstract units* $\Phi^\star = \{\mathfrak{A}_i^\star, i \in \mathbb{N}\}$ *is an abstract unit stream*

When these abstract concepts need to be measured at a point in a communication network, different statistics can be obtained, that are exposed hereafter.

B.1. Second order properties of times between events

In [Cox67] the evaluation of many statistics from series of events is presented. For the series of time between events $\mathcal{I}_1, \mathcal{I}_2, ...$, their common marginal distribution is $F_\mathcal{I}$

$$\mathbb{E}(\mathcal{I}) = \int_0^\infty t f_\mathcal{I}(t) dt$$

$$\mathrm{var}(\mathcal{I}) = \int_0^\infty t^2 f_\mathcal{I}(t) dt - \{\mathbb{E}(\mathcal{I})\}^2$$

and the ratio

$$C^2(\mathcal{I}) = \frac{\mathrm{var}(\mathcal{I})}{\{\mathbb{E}(\mathcal{I})\}^2}$$

The ratio $C(\mathcal{I})$ is called the coefficient of variation of $\mathcal{I}$. For exponential distribution, $C(\mathcal{I}) = 1$. The correlation in the sequence $\{\mathcal{I}\}$ can be evaluated, for example the autocorrelation

$$\rho_k = \frac{\mathrm{cov}(\mathcal{I}_i, \mathcal{I}_{i+k})}{\mathrm{var}(\mathcal{I})} \tag{B.1}$$

or the autocovariance $C_k = \mathrm{var}(\mathcal{I})\rho_k$

A variance function is introduced

$$\begin{aligned} V_k &= \mathrm{var}(\mathcal{I}_1 + ... + \mathcal{I}_k) \\ &= k\,\mathrm{var}(\mathcal{I}) + 2\sum_{l=1}^{k-1}\sum_{h=1}^{l}\mathrm{cov}(\mathcal{I}_i, \mathcal{I}_{i+k}) \\ &= kV_1 + 2\sum_{l=1}^{k-1}(k-l)C_l \end{aligned} \tag{B.2}$$

An index of dispersion for intervals, J_k is also introduced.

$$J_k = \frac{V_k}{k\{\mathbb{E}(\mathcal{I})\}^2} \tag{B.3}$$

These formulas show that it is possible to compute the coefficient of variation, the variance function and the coefficient of dispersion from the sequence of the inter-arrivals time.

B.2. Second order properties of counts

The counting process N_t can also be used to evaluate the characteristics of the arrival process. The process need to be evaluated on the continuous time interval $[0, t]$. The mean time curve and the variance time curve are defined. The mean time curve is the evolution of $M(t) = \mathbb{E}(N_t)$ versus t and the variance time curve is the evolution of $V(t) = \mathbb{E}(N_t^2) - \mathbb{E}(N_t)^2$

Similarly, the index of dispersion of counts (IDC) can be used to measure the properties on the incoming traffic stream. It is defined by $I(t) = \frac{V(t)}{M(t)}$. For a Poisson process $I(t)$ is constantly 1, for a renewal process it tends to $C(\mathcal{I})$.

In order to define a covariance time plot $C_1(t)$, let $C_i(\tau)$ be the covariance between the number of arrivals in two intervals of length τ separated by $i-1$ similar intervals.

$$\begin{aligned} C_1(k\tau) &= \sum_{i=1}^{k} iC_i(\tau) + \sum_{i=k+1}^{2k}(2k-i)C_i(\tau) \\ V(k\tau) &= kV(\tau) + 2\sum_{i=1}^{k-1}\sum_{j=1}^{i}C_j(\tau) \\ &= kV(\tau) + 2\sum_{i=1}^{k-1}(k-i)C_j(\tau) \end{aligned} \tag{B.4}$$

$$C_1(t) = \frac{V(2t)}{2} - V(t) \tag{B.5}$$

B.3. Estimation

In this section, it is explained how to evaluated in practice:

- First and second order moments of intervals.
- Correlation between counts.
- Variance-time plot.

B.3.1. Theoretical construction of the estimates

First and second order moments of intervals Given the observation $\{\mathcal{I}_1, \mathcal{I}_2, ..., \mathcal{I}_{n_0}\}$. An estimate of $\mathbb{E}(\mathcal{I})$ is obtained

$$\bar{\mathcal{I}} = \frac{1}{n_0}\sum_{i=1}^{n_0}\mathcal{I}_i \tag{B.6}$$

The recurvive version is obtained from the following equation

$$\bar{\mathcal{I}}_{n_0+1} = \frac{n_0 \times \bar{\mathcal{I}}_{n_0} + \mathcal{I}_{n_0+1}}{n_0+1} \tag{B.7}$$

To estimate the covariance, an estimate of

$$C_j = \mathbb{E}(\{\mathcal{I}_i - \mathbb{E}(\mathcal{I})\}\{\mathcal{I}_{i+j} - \mathbb{E}(\mathcal{I})\}) \tag{B.8}$$

is required. Since the mean value of $\mathcal{I}$ is unknown, the estimator is

$$\tilde{C}_j = \frac{1}{n_0-j}\sum_{i=1}^{n_0-j}(\mathcal{I}_i - \bar{\mathcal{I}}'_j)(\mathcal{I}_{i+j} - \bar{\mathcal{I}}''_j) \tag{B.9}$$

$$= \frac{1}{n_0-j}(\sum_{i=1}^{n_0-j}\mathcal{I}_i\mathcal{I}_{i+j}) - \bar{\mathcal{I}}'_j\bar{\mathcal{I}}''_j \tag{B.10}$$

where $\bar{\mathcal{I}}'_j = \frac{1}{n_0-j}\sum_{i=1}^{n_0-j}\mathcal{I}_i$ and $\bar{\mathcal{I}}''_j = \frac{1}{n_0-j}\sum_{i=1}^{n_0-j}\mathcal{I}_{i+j}$

Correlation between counts If n arrival events have been observed in $[0, t_0]$, the interval is divided in k intervals of length τ if n_j is the observed number of events in the jth interval of length τ. Then

$$\tilde{C}_i(\tau) = \frac{1}{k-i}\sum_{j=1}^{k-i} n_j n_{j+i} - \frac{1}{(k-i)^2}(\sum_{j=1}^{k-i} n_j)(\sum_{j=1}^{k-i} n_{j+i}) \tag{B.11}$$

Variance-time plot $V(t)$ can be estimated by segmenting the time interval in intervals of length τ and n_i is again the number of events in the ith interval.

B.3.2. Analysis of inter-arrival times

The autocorrelation of a randon sequence of interval time $\mathcal{I}_n$ is defined as $E[(\mathcal{I}_n - E(\mathcal{I})) \times (\mathcal{I}_{n-k} - E(\mathcal{I}))]$, $k \in [1, N]$ and the autocorrelation coefficient is defined as $C_{[k]} = \frac{E[(\mathcal{I}_n - E(\mathcal{I}))\times(\mathcal{I}_{n-k} - E(\mathcal{I}))]}{E[(\mathcal{I}_n - E(\mathcal{I}))^2]}$. The inter-arrival time sequence is obtained by the segmentation of the time axis of the instant of arrivals t_n defined in the previous module by $\mathcal{I}_n = t_n - t_{n-1} \tilde{=} X_n\tau$ where τ is a sufficiently small interval. The sequence $\{X_n\}$ is then the discrete approximation of the sequence $\{\mathcal{I}_n\}$. From this sequence the coefficient $C_{[k]} =$ can be obtained iteratively by the scheme indicated in figure B.1.

It is estimated with a recursive formula up to rank k. At every new sample, a new estimate of the mean is calculated ($E(n) = \{E(n-1) * (n-1) + X_n\}/(n)$). This estimate is combined with the k last samples to calculate the k new $C_{[k]}^{(n)}$ estimates $C_{[k]}^{(n)} = (X_n - E(n)) \times (X_{n-k} - E(n)$. These value are again put in recursive mean estimator, in order to get the $\widehat{C_k(n)} = \{\widehat{C_k(n)}(n-1) + C_k^{(n)}\}/n$.

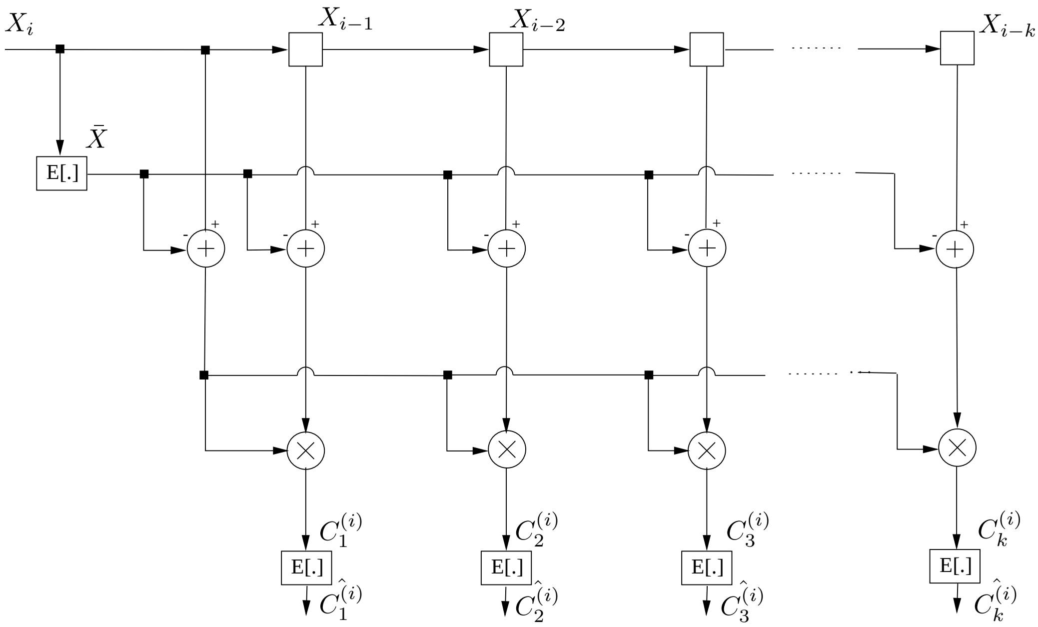


Figure B.1.: Mode of operation autocorrelation

C

Appendix - Results for the queues with BMAP/MMPPs inputs

This chapter gathers results about a queuing system with BMAP or MMPP arrivals that are served by a service unit. These queues are noted BMAP/G/1 or MMPP/G/1 indicating the arrival process (BMAP or MMPP), the service law G and the service of one customer at the time.

C.1. Arrival Properties of the MMPP process

We first summarize results characterising the arrival process.

C.1.1. Notation

The considered MMPP has a state matrix Q with an arrival matrix Λ. The matrix $F(\infty)$ of the transition probability of the Markov chain embedded at arrivals epochs is important in order to characterize the arrival process and can be expressed with a matrix form.

$$F(\infty) = (\Lambda - Q)^{-1}\Lambda \tag{C.1}$$

and **p** is the stationary vector of $F(\infty)$ (i.e. $\mathbf{p}.F(\infty) = \mathbf{p}$ and $\mathbf{pe} = 1$). The stationary vector of $F(\infty)$ can be obtained

$$\mathbf{p} = \frac{1}{\boldsymbol{\pi}.\boldsymbol{\lambda}}.\boldsymbol{\pi}.\Lambda \tag{C.2}$$

The mean arrival rate can be expressed as:

$$\lambda_A = \boldsymbol{\pi}\boldsymbol{\lambda} \tag{C.3}$$

C.1.2. Conditional moments of the time between arrivals in an MMPP

C.1.2.1. Laplace transform of the arrival process

The laplace transform is defined for renewal process. For the time between the arrivals of the MMPP , a Laplace transform is defined equivalently.

$$f^*(s) = \mathbb{E}(e^{-s\mathcal{I}}) = (sI - Q + \Lambda)^{-1}\Lambda$$

$$f^*(s_1, \ldots, s_n) = \mathbb{E}\{(\exp[-\sum_{k=1}^{n} s_k \mathcal{I}_k]\} = \prod_{k=1}^{n}\{(s_k I - Q + \Lambda)^{-1}\Lambda\}$$

C.1.2.2. Moments for the m-state inter-arrivals

Results on the moments of the inter-arrivals times can be given using the characteristic matrices of the arrival process.

$$\begin{aligned} \mu'_{i,k} &= \mathbb{E}\{\mathcal{I}_i^k\} \\ &= \frac{\partial^k f^*(s_1, \cdots, s_i)}{\partial^k s_i}\Big|_{s_1=0,..,s_i=0} \\ &= k!\big[(\Lambda - Q)^{-1}\Lambda\big]^{i-1}(\Lambda - Q)^{-(k+1)}\Lambda, \;\; i \geq 1, \; k \geq 1 \end{aligned} \tag{C.4}$$

$$\begin{aligned} \mu_{1,k+1} &= \mathbb{E}\{\mathcal{I}_1\mathcal{I}_{k+1}\} \\ &= \frac{\partial^2 f^*(s_1, \cdots, s_{k+1})}{\partial s_1 \partial s_{k+1}}\Big|_{s_1=0,..,s_{k+1}=0} \\ &= (\Lambda - Q)^{-2}\Lambda\big[(\Lambda - Q)^{-1}\Lambda\big]^{k-1}(\Lambda - Q)^{-2}\Lambda, \;\; k \geq 1 \end{aligned} \tag{C.5}$$

The k-step correlation matrix is given by

$$\begin{aligned} &\mathbb{E}\{(\mathcal{I}_1 - \mathbb{E}[\mathcal{I}_1])(\mathcal{I}_{k+1} - \mathbb{E}[\mathcal{I}_{k+1}])\} \\ &\qquad = (\Lambda - Q)^{-2}\Lambda\big[(\Lambda - Q)^{-1}\Lambda\big]^{k-1}\{I - (Q - \Lambda)^{-1}\Lambda\}(\Lambda - Q)^{-2}\Lambda \end{aligned} \tag{C.6}$$

C.1.2.3. Moments for the interval-stationary version

$T_{a,i}$ is the time between ith and $(i+1)$st arrivals in the interval-stationnary version (for a process that was startet at $t = -\infty$ and not $t = 0$.

$$\begin{aligned} \mathbb{E}\{T_{a,i}^k\} &= k!\mathbf{p}[(\Lambda - Q)^{-1}\Lambda]^{i-1}(\Lambda - Q)^{-(k+1)}\Lambda\mathbf{e} \\ &= k!\mathbf{p}(\Lambda - Q)^{-(k+1)}\Lambda\mathbf{e} \end{aligned} \tag{C.7}$$

$$\begin{aligned} &\mathbb{E}\{(T_{a,1} - \mathbb{E}[T_{a,1}])(T_{a,k+1} - \mathbb{E}[T_{a,k+1}])\} \\ &= \mathbf{p}(\Lambda - Q)^{-2}\Lambda\Big\{\big[(\Lambda - Q)^{-1}\Lambda\big]^{k-1} - \mathbf{ep}\Big\}(\Lambda - Q)^{-2}\Lambda\mathbf{e} \end{aligned} \tag{C.8}$$

where

$$\mathbf{ep} = \begin{pmatrix} p_1 & p_2 & \cdots & p_m \\ p_1 & p_2 & \cdots & p_m \\ \vdots & \vdots & \ddots & \vdots \\ p_1 & p_2 & \cdots & p_m \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \times \begin{pmatrix} p_1 & p_2 & \cdots & p_m \end{pmatrix}$$

C.1.2.4. Simplifications for MMPP(2)

Notation When a MMPP(2) with two states is considered with characteristic matrices

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \qquad Q = \begin{pmatrix} -r_1 & r_1 \\ r_2 & -r_2 \end{pmatrix} \tag{C.9}$$

The parameter σ is defined.

$$\sigma = \frac{\lambda_1.\lambda_2}{\lambda_1.\lambda_2 + \lambda_1.r_2 + \lambda_2.r_1} \tag{C.10}$$

It is the smallest eigenvalue of $F(\infty) = (\Lambda - Q)^{-1}\Lambda$.
The auto-covariance of the inter-arrivals can be written:

$$C[k] = C[1]\sigma^{k-1},\ k \geq 1 \tag{C.11}$$

Computation of c_k^2 for the MMPP(2) In [SW86] the MMPP(2) is used to model the superposition of single known renewal process. In the general case, if this is not known the following asymptotic has been derived.

$$\begin{aligned}
C_k^2 &= \frac{\operatorname{cov}(\mathcal{I}_1, \mathcal{I}_1) + 2\sum_{j=1}^{k-1} \frac{k-j}{k} \operatorname{cov}(\mathcal{I}_1, \mathcal{I}_{1+j})}{\mathbb{E}[\mathcal{I}_1]^2} \\
&= \frac{\operatorname{cov}(\mathcal{I}_1, \mathcal{I}_1) + 2\sum_{j=1}^{k-1} \frac{k-j}{k} C_k[1]\sigma^{j-1}}{\mathbb{E}[\mathcal{I}_1]^2} \\
\lim_{k\to\infty} C_k^2 &= \frac{\operatorname{cov}(\mathcal{I}_1, \mathcal{I}_1) + \frac{C_k[1]}{\sigma}\frac{2\sigma}{\sigma-1}}{\mathbb{E}[\mathcal{I}_1]^2} \\
&= \frac{\operatorname{cov}(\mathcal{I}_1, \mathcal{I}_1) + \frac{2C_k[1]}{\sigma-1}}{\mathbb{E}[\mathcal{I}_1]^2}
\end{aligned} \tag{C.12}$$

Using the fact that

$$\sum_{k=1}^{n} kq^k = \frac{\sum_{k=1}^{n} q^n - nq^{n+1}}{1-q} \qquad \text{and } \lim_{k\to\infty} 2\sum_{j=1}^{k}(1-\frac{j}{k})a^j = \frac{2a}{a-1} \tag{C.13}$$

C.2. MMPP/G/1 - Queues with MMPP input

In this section, queues with MMPP arrival are considered.

It is assumed that the service time has a generic distribution denoted by $H(t)$ with mean h. The input process is a MMPP with matrix Q and Λ with mean arrival rate $\lambda_A = \pi\lambda$ With these assumptions, some queueing results can be obtained.

C.2.1. The descrition of MMPP/G/1 queue

The queueing properties of such an arrival process can be written by a semi-Markov process with transition probability matrix $\tilde{Q}(t)$

$$\tilde{Q}(t) = \begin{pmatrix} \tilde{B}_0(t) & \tilde{B}_1(t) & \tilde{B}_2(t) & \dots \\ \tilde{A}_0(t) & \tilde{A}_1(t) & \tilde{A}_2(t) & \dots \\ 0 & \tilde{A}_0(t) & \tilde{A}_1(t) & \dots \\ 0 & 0 & \tilde{A}_0(t) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad t > 0$$

The probabilities $\tilde{A}_n(t)$ and $\tilde{B}_n(t)$ have their transforms:

$$\begin{aligned} A_n(s) &= \int_0^\infty e^{-st} d\tilde{A}_n(t) \qquad B_n(s) = \int_0^\infty e^{-st} d\tilde{B}_n(t) \\ A(z,s) &= \sum_{n=0}^{\infty} A_n(s) z^n \qquad B(z,s) = \sum_{n=0}^{\infty} B_n(s) z^n \end{aligned}$$

with the particular values of:

$$\begin{aligned} A_n &= A_n(0) \qquad B_n = B_n(0) \\ A &= A(1,0) \qquad B = B(1,0) \end{aligned}$$

Some results are obtained to express the matrix **A** (matrix of the probability A_{ij} that a service ends with the MMPP in state j given that the service began in state i) and the vector $\boldsymbol{\beta}$ (with the j^{th} component beeing the conditional number of arrivals during a service which starts with the arrival process in state j) can be used to describe the queue. We have

$$\begin{aligned} A &= \int_0^\infty e^{Qt} d\tilde{H}(t) dt \\ \boldsymbol{\beta} &= \rho \mathbf{e} + (Q + \mathbf{e}\boldsymbol{\pi})^{-1} (A - I) \boldsymbol{\lambda} \\ B_n &= (\Lambda - Q)^{-1} \Lambda A_n \end{aligned}$$

C.2.2. Waiting time distribution

The joint transform of the waiting time distribution and the state of MMPP can be determined by

$$\mathbf{W}(s) = \begin{cases} s(1-\rho)\mathbf{g}[sI + Q - \Lambda(1 - H(s))]^{-1} & s \geq 0 \\ \pi & s = 0 \end{cases} \tag{C.14}$$

The transform of the virtual waiting time is given by

$$W(s) = \mathbf{W}(s)\mathbf{e} \tag{C.15}$$

where π is the stationary vector $\pi Q = 0$, $\pi \mathbf{e} = 1$ and **g** is the steady state of the matrix **G**, the matrix G is explained hereafter.

C.2.2.1. Approximation of the virtual waiting time

The complementary density function of the virtual waiting time $P(W_v > x)$ can be approximated by

$$P(W_v > x) \approxeq a_1 \exp(-s_1 x) + a_2 \exp(-s_2 x) \tag{C.16}$$

For a MMPP(2)/D/1 queue, the roots s_1 and s_2 of $W_v(s)$ need first to be obtained. $\mathbf{g} = [g_1 1 - g_1]$ can be calculated. Afterwards, the approximate queing statistics can be computed for the approximated queue.

C.2.3. The queue length at departure and the matrix G

The queue length at departure can be obtained through the equations verified by $\tilde{Q}(\infty)$ of the Markov chain at departure. Its stationnary vector is written $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1, ...)$. We have the following equation

$$x_i = x_0 B_i + \sum_{\nu=1}^{i+1} x_\nu A_{i+1-\nu} \tag{C.17}$$

The matrix G is the root of

$$G = \int_0^\infty e^{(Q-\Lambda+\Lambda G)t} d\tilde{H}(t)dt$$

The matrix G can be computed iteratively (or depending of $\tilde{H}$ faster calculation can be obtained) **g** is the steady state of the matrix **G**. It can be shown that

$$x_0 = \frac{1-\rho}{\lambda_A}\mathbf{g}(\Lambda - Q)$$

The queue length could be obtained recursively but other approaches may lead to exact results. Two types of queue size distribution:

- at packets departure instants x_k
- steady state (at an arbitrary instant): y_k
- at packets arrivals instant x_k^a.

The determination of the sequence is derived from the equation verified by the z-transform of the sequences ($X(z), Y(z)$ and $X^a(z)$) (derived in [Ram80]). The calculation of the matrices A_n is necessary. The following method is recommended because otherwise, unstable x_k sequence are obtained.

$$X(z)[zI - A(z)] = (-x_0(Q-\Lambda)^{-1})D(z)A(z) \tag{C.18}$$

$$Y(z)D(z) = \lambda(z-1)X(z) \tag{C.19}$$

$$\lambda_A^{-1}X^a(z)\Lambda\mathbf{e} = -\lambda_A^{-1}X^a(z)(Q-\Lambda)\mathbf{e} \tag{C.20}$$

Calculation of G A scaling parameter Θ is used from the arrival matrices Λ and Q.

$$\Theta = \max_i(\Lambda_{ii} - Q_{ii}) \tag{C.21}$$

$$\gamma_n = \int_0^\infty \exp(-\Theta t)\frac{(\Theta.t)^n}{n!}d\tilde{H}(t) \tag{C.22}$$

From there, the following recursion can be used:

Initial step Given $G_0 = 0$, and $H_{0,k} = I$ for $k = 0, 1, 2, ...,$, compute γ_n for $n = 0, 1, .., n^\star$ ($\sum_{k=1}^{n^\star} \gamma_k > 1 - \epsilon_1$ and $\epsilon_1 << 1$).

Recursion For k=0,1,2,..., compute

$$\begin{aligned} H_{n+1,k} &= \Big[I + \frac{1}{\Theta}(Q - \Lambda + \Lambda G_k)\Big]H_{n,k} \ , \ n = 0, 1, .., n^\star \\ G_{k+1} &= \sum_{n=0}^{n^\star} \gamma_n H_{n,k} \end{aligned}$$

Stopping step

$$||G_{k-1} - G_k|| < \epsilon_2$$

Simplifications Using a deterministic arrival rate ($H(t) = \delta(t)$), the serie $\{\gamma_n\}$ can be directly computed.

$$\gamma_n = \exp(-\theta)\frac{\theta^n}{n!} \tag{C.23}$$

C.2.3.1. Moments of queue length

Knowing the queue size z-transform (for example $X(z)$), the moments of the queue length can be obtained for the i-th derivative of $\mathbf{X}^{(i)} = X^{(i)}(1)$. The mean is given from $\mathbf{X}^{(1)}\mathbf{e}$ and the variance from $\mathbf{X}^{(2)}\mathbf{e}$.

Using

$$X(z) = \sum_{i=0}^{\infty} x_i z^i$$

By differentiating,

$$X(z)[zI - A(z)] = -\mathbf{x}_0(Q-\Lambda)^{-1}D(z)A(z)$$
$$\text{where } D(z) = (Q-\Lambda) + z\Lambda$$

This enables to write :

$$\mathbf{X}(\mathbf{1}) = \boldsymbol{\pi} - x_0(Q-\Lambda)^{-1}QA(I-A-\mathbf{e}\boldsymbol{\pi})^{-1} \tag{C.24}$$

$$\mathbf{X}^{(1)}\mathbf{e} = \frac{1}{2(1-\rho)}\Big\{\mathbf{X}(\mathbf{1})A^{(2)}\mathbf{e} + U^{(2)}\mathbf{e} + 2\big\{U^{(1)} - \mathbf{X}(\mathbf{1})[I-A^{(1)}]\big\}(I-A-\mathbf{e}\boldsymbol{\pi})^{-1}\boldsymbol{\beta}\Big\} \tag{C.25}$$

$$\mathbf{X}^{(1)} = (\mathbf{X}^{(1)}\mathbf{e})\boldsymbol{\pi} + \big\{U^{(1)} - \mathbf{X}(\mathbf{1})[I-A^{(1)}]\big\}(I-A-\mathbf{e}\boldsymbol{\pi})^{-1} \tag{C.26}$$

$$\mathbf{X}^{(2)}\mathbf{e} = \frac{1}{3(1-\rho)}\Big\{3\mathbf{X}^{(1)}A^{(2)}\mathbf{e} + 3\mathbf{X}(\mathbf{1})A^{(3)}\mathbf{e} + U^{(3)}\mathbf{e} \tag{C.27}$$

$$+3\big\{U^{(2)} + \mathbf{X}(\mathbf{1})A^{(2)} - 2\mathbf{X}(\mathbf{1})[I-A^{(1)}]\big\}(I-A-\mathbf{e}\boldsymbol{\pi})^{-1}\boldsymbol{\beta}\Big\} \tag{C.28}$$

These equations require the computation of the derivative of U

$$U(z) = -\mathbf{x}_0(Q-\Lambda)^{-1}D(z)A(z) \tag{C.29}$$

$$U^{'}(z) = -\mathbf{x}_0(Q-\Lambda)^{-1}[D(z)A'(z) + \Lambda A(z)] \tag{C.30}$$

$$U^{''}(z) = -\mathbf{x}_0(Q-\Lambda)^{-1}[D(z)A^{''}(z) + 2\Lambda A'(z)] \tag{C.31}$$

$$U^{(3)}(z) = -\mathbf{x}_0(Q-\Lambda)^{-1}[D(z)A^{(3)}(z) + 3\Lambda A''(z)] \tag{C.32}$$

and of the derivatives of A.

C.2.3.2. Moments for arbitrary queue

$$\mathbf{Y}^{(1)}\mathbf{e} = \mathbf{X}^{(1)}\mathbf{e} - \frac{1}{2\lambda}\boldsymbol{\pi}D^{(2)}\mathbf{e} + [\frac{1}{\lambda}\boldsymbol{\pi}D^{(1)} - \mathbf{X}(\mathbf{1})](\mathbf{e}\boldsymbol{\pi} + D)^{-1}D^{(1)}\mathbf{e} \tag{C.33}$$

$$\mathbf{Y}^{(1)} = (\mathbf{Y}^{(1)}\mathbf{e})\boldsymbol{\pi} + [\lambda\mathbf{X}(\mathbf{1}) - \boldsymbol{\pi}D^{(1)}](\mathbf{e}\boldsymbol{\pi} + D)^{-1} \tag{C.34}$$

$$\mathbf{Y}^{(2)}\mathbf{e} = \mathbf{X}^{(2)}\mathbf{e} - \frac{1}{\lambda}\mathbf{Y}^{(1)}D^{(2)}\mathbf{e} - \frac{1}{3\lambda}\boldsymbol{\pi}D^{(3)}\mathbf{e} - 2[\mathbf{X}^{(1)} - \frac{1}{\lambda}\mathbf{Y}^{(1)}D^{(1)} - \frac{1}{\lambda}\boldsymbol{\pi}D^{(2)}](\mathbf{e}\boldsymbol{\pi} + D)^{-1}D^{(1)}\mathbf{e} \tag{C.35}$$

C.2.4. Queue sizes distribution computation

C.2.4.1. Principle

The principle is to compute

$$x_i = \Big[x_0\bar{B}_i + \sum_{\nu=1}^{i-1} x_\nu \bar{A}_{i+1-\nu}\Big](I - \bar{A}_1)^{-1},\ i \geq 1. \tag{C.36}$$

where the matrices $\bar{A}_k$ and $\bar{B}_k$ are defined as

$$\bar{A}_k = \sum_{i=k}^{\infty} A_i G^{i-k} \ , \qquad \bar{B}_k = \sum_{i=k}^{\infty} B_i G^{i-k} \ k \geq 0. \tag{C.37}$$

They can be calculated by the backward recursion

$$\bar{A}_k = A_k + \bar{A}_{k+1} G \ , \qquad \bar{B}_k = B_k + \bar{B}_{k+1} G. \tag{C.38}$$

C.2.4.2. Computation of A using the backward approximation

A can be calculated using the recursion. $\{K_n^{(j)}\}$ is defined recursively by $K_0^{(0)} = I, K_n^{(0)} = 0, n \geq 1$, and

$$K_0^{(j+1)} = K_n^{(j)} \left[\Theta^{-1}(Q - \Lambda) + I\right] \tag{C.39}$$
$$K_n^{(j+1)} = K_n^{(j)} \left[\Theta^{-1}(Q - \Lambda) + I\right] + K_{n-1}^{(j)} \Theta^{-1} \Lambda \tag{C.40}$$

$$A_i = \sum_{k=i}^{\infty} \gamma_i K_i^{(k)} \tag{C.41}$$

So A is approximated by the following partial sum to approach:

$$A = \sum_{i=0}^{\infty} A_i \tag{C.42}$$

C.2.4.3. Calculation the derivatives of A

A and its derivatives are computed with the help of the matrix S. It is defined by :

$$S = \begin{bmatrix} Q & \Lambda & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Q & 2\Lambda & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & Q & 3\Lambda \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & Q \end{bmatrix} \text{ with } \mathbf{0} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{C.43}$$

Writing

$$L = \begin{bmatrix} A & A' & A'' & A^{(3)} \end{bmatrix} \tag{C.44}$$

L will be obtained by the following recursion:

$$L_0 = I_2^8 = \begin{bmatrix} I_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$
$$L_{k+1} = L_k \times (I_8 + S\Theta^{-1})$$

that enables the approximation of L through:

$$L = \sum_{k=0}^{\infty} \gamma_k L_k \tag{C.45}$$

C.3. Queues with BMAP input arrival process

C.3.1. Notation

The BMAP arrival process is defined by a series of D_k of non negative matrices for $k \geq 1$ and with D_0 having negative diagonal elements and nonnegative off diagonal elements.

$$D = \sum_{k=0}^{\infty} D_k$$

is an irreductible infinitesimal generator.

The matrix generating function is defined by

$$D(z) = \sum_{k=0}^{\infty} D_k z^k \text{ for } |z| \leq 1.$$

$\boldsymbol{\pi}$ is stationary probability vector of the process with generator D. $\boldsymbol{\pi}$ satisfies

$$\boldsymbol{\pi} D = 0 \qquad\qquad \boldsymbol{\pi}\mathbf{e} = 1$$

With these assumptions, some queueing results can be obtained.

Link with the MMPP The BMAP generalizes the MMPP with $D_0 = Q - \Lambda$, $D_1 = \Lambda$ and $D_k = 0,\ k \geq 2$

$$D_{MMPP} = \sum_{k=0}^{\infty} D_k = Q \tag{C.46}$$

The function $D(z)$ is calculated

$$D_{MMPP}(z) = \sum_{k=0}^{\infty} D_k . z^k = Q - \Lambda + \Lambda z \tag{C.47}$$

C.3.2. Properties of the arrival process

The mean arrival rate is written:

$$\lambda_A^{-1} = \boldsymbol{\pi} \sum_{k=1}^{\infty} k D_k \mathbf{e} = \boldsymbol{\pi}\mathbf{d}$$

where $\mathbf{d} = \sum_{k=1}^{\infty} k D_k \mathbf{e}$.

C.3.3. BMAP/G/1 - Queues with MMPP input

It is assumed that the service time has a generic distribution denoted by $\tilde{H}(t)$. This distribution has a Laplace-Stieltjes transform written $H(s)$ and a mean value of μ_1. The traffic intensity ρ is defined by $\rho = \frac{\mu_1}{\lambda_A}$.

C.3.3.1. Embedded Markov Renewal Process at departures

The BMAP/Q/1 queue can be analyzed through the embedded Markov renewal process at departures. The transition probability matrix is given by

$$\tilde{P}(t) = \begin{pmatrix} \tilde{B}_0(t) & \tilde{B}_1(t) & \tilde{B}_2(t) & \dots \\ \tilde{A}_0(t) & \tilde{A}_1(t) & \tilde{A}_2(t) & \dots \\ 0 & \tilde{A}_0(t) & \tilde{A}_1(t) & \dots \\ 0 & 0 & \tilde{A}_0(t) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad t > 0$$

the following mass functions are used:

- $[\tilde{A}_n(t)]_{ij}$: the probability at time 0, which left at least one customer in the system and the arrival process in phase i, the next departure occurs no later than time t with the arrival process in phase j, and during that service there were n arrivals.
- $[\tilde{B}_n(t)]_{ij}$: the probability at time 0, which left the system empty and the arrival process in phase i, the next departure occurs no later than time t with the arrival process in phase j, and leaving n customers in the system.

The same transform matrices are defined:

$$\begin{aligned} A_n(s) &= \int_0^\infty e^{-st} d\tilde{A}_n(t) \qquad B_n(s) = \int_0^\infty e^{-st} d\tilde{B}_n(t) \\ A(z,s) &= \sum_{n=0}^{\infty} A_n(s) z^n \qquad B(z,s) = \sum_{n=0}^{\infty} B_n(s) z^n \end{aligned}$$

with the particular values of:

$$\begin{aligned} A_n &= A_n(0) \qquad B_n = B_n(0) \\ A &= A(1,0) \qquad B = B(1,0) \end{aligned}$$

The same relation between A and $\tilde{H}$ can be obtained as for the MMPP/Q/1 queue.

$$A = \int_0^\infty e^{Dt} d\tilde{H}(t) dt$$

C.3.3.2. Virtual waiting time

The Laplace-Stieljes distribution of the virtual waiting time is

$$W_v(s) = sy_0[sI + D(H(s))]^{-1}\mathbf{e} \tag{C.48}$$

$$W_v(0) = \pi.\mathbf{e} \tag{C.49}$$

where H(s) is the Laplace Stieljes transform of $\tilde{H}$, the service time distribution and D is the infinitesimal generator

C.3.4. Tail probabilities

In [ACW92a], asymptotics for the BMAP/G/1 queuing metrics are derived. Since the BMAP (also called N process or versatile Markovian Point Process) is a generalization of MAP (Markovian arrival process) that includes PH-renewal process and MMPP), so the provided asymptotics are quite useful.

The considered queue is a BMAP/G/1, a single server with first-come first-served discipline , unlimited waiting room and i.i.d (independent and identically distributed) service time to serve customers that have BMAP arrivals.

The following variable in the queuing system are considered: Q is the steady state queue length (number in system) and L the workload (virtual waiting time) at an arbitrary time and W is the steady-state waiting time (experienced by an arriving customer before beginning service), Q^a is the steady state queue length (number in system) just before an arrival, Q^d is the steady state queue length (number in system) just after a departure.

$$\lim_{t\to\infty} e^{\eta t} \Pr[L > t] = \alpha_L \qquad \lim_{t\to\infty} e^{\eta t} \Pr[W > t] = \alpha_W \tag{C.50}$$

$$\lim_{k\to\infty}\sigma^{-k}\Pr[Q>k]=\beta \qquad \lim_{k\to\infty}\sigma^{-k}\Pr[Q^a>k]=\beta^a \qquad \lim_{k\to\infty}\sigma^{-k}\Pr[Q^d>k]=\beta^d \tag{C.51}$$

with the following positive constants σ and η are the asymptotic decay rates and $\alpha_L, \alpha_W, \beta, \beta^a, \beta^d$ are associated asymptotic constants.

Some relations between these parameters can be derived:

$$\beta^a = \alpha_W \qquad \beta^d = \alpha_L \qquad \frac{\alpha_L}{\alpha_W} = \frac{\rho(1-\sigma)}{\eta\sigma} \tag{C.52}$$

and also

$$\mathbb{E}(e^{\eta V}) = \frac{1}{\sigma} \tag{C.53}$$

where V is the service time distribution, with characteristic function $\phi(\eta) = \mathbb{E}(e^{\eta V})$.

In order to determine all asymptotics just one of the asymptotic decay rate (e.g. σ) and one constant is required to derive all other asymptotics.

For the BMAP/G/1, with the characteristic matrices $D_k, 0 \geq k \geq M$ (where M is the maximal batch size), the normalized arrival polynome $D(z)$ can be formed by

$$D_n(z) = \sum_{k=0}^{M} D_k \frac{z^k}{\rho} \qquad \text{where } \lambda = \sum_{k=1}^{M} k\boldsymbol{\pi} D_k \mathrm{e} \tag{C.54}$$

where $\boldsymbol{\pi}$ is the stationary probability vector of $D = \sum_{k=0}^{M} D_k$. They have shown that the couple $(\sigma, \eta$ shall be the unique solution of

$$\mathit{pf}(D_n(\frac{1}{\sigma})) = \frac{\eta}{\lambda} \qquad e^{\eta V} = \frac{1}{\sigma} \tag{C.55}$$

with $\sigma\in]0,1[$ and $\eta > 0$. And $p_D(z) = \mathit{pf}(D_n(z))$ is the Perron-Frobenius eigenvalue of $D_n(z)$. The left side equation is based on characteristics of the arrival process and the right side equation is based on the service characteristics.

From there it remains to calculate $\beta = \frac{\rho l^d \mathrm{e}}{\eta}$ where l^d is the asymptotic for the queue length distribution at departure derived from $X^d(z)[zI - A(z)] = (-x_0^d D_0^{-1})D(z)A(z)$.

$$l^d = \frac{\eta(-x_0^d D_0^{-1})\mathrm{v}(\frac{1}{\sigma})\mathrm{u}(\frac{1}{\sigma})}{\rho(p'(\frac{1}{\sigma})-1)} \tag{C.56}$$

where $p'(\frac{1}{\sigma}) = \rho\Phi'(\eta)p'_D(\frac{1}{\sigma})$

The authors have also studied exponential approximation to derive this asymptotic like in [ACW92b],[ACW92c]

D

Probability density

Many of the following distribution are described in [STC95]

D.1. Discrete densities

D.1.1. The uniform distribution

It is a discrete random value that takes n integer values $(1, 2, .., n)$ with the same probability

$$\Pr(X = k) = \frac{1}{n}$$

The mean is $\mathbb{E}[X] = \frac{n+1}{2}$ and the variance $\sigma^2 = \frac{n^2-1}{12}$

D.1.2. The binomial distribution

The binomial distribution takes two values (0 and 1) with the probability

$$\Pr(X = 0) = q \quad \Pr(X = 1) = p \;\; p + q = 1 \; p, q > 0$$

The mean is $\mathbb{E}[X] = p$ and the variance $\sigma^2 = pq = p(1 - p)$

D.1.3. The geometrical distribution

The geometrical distribution takes integers value in range $1, 2, ..$ with the probability

$$\Pr(X = i) = p(1 - p)^{i-1} \quad p \in [0, 1] \; p + q = 1 \;\; i \geq 1$$

The mean is $\mathbb{E}[X] = \frac{1}{p}$ and the variance $\sigma^2 = \frac{q}{p^2} = \frac{(1-p)}{p^2}$. It can be used as a discrete equivalent of the exponential distribution

D.1.4. The binomial distribution

It is a discrete random value that takes $n + 1$ integer values $(0, 1, 2, .., N)$ with the same probability

$$\Pr(X = k) = C_N^k p^k (1 - p)^{N-k}, \quad p \in [0, 1] \quad 0 \leq k \leq N$$

The mean is $\mathbb{E}[X] = Np$ and the variance $\sigma^2 = Np(1 - p)$. It can be used as a discrete equivalent of the poisson arrivals.

D.2. Continuous density

D.2.1. The uniform distribution

This law is used to model a random value U in an interval $[a, b]$. The distribution is constant in the whole interval. So the probability density function is the following

$$\begin{aligned}\frac{dP}{du} &= \frac{1}{b-a} \text{ if } u \in [a, b]\\ \frac{dP}{du} &= 0 \text{ otherwise}\end{aligned}$$

The statistical parameters of this distribution are straigthforward.

$$\text{Mean : } E[U] = \frac{a+b}{2}$$
$$\text{Variance : } \sigma_U^2 = \frac{(b-a)^2}{12}$$

D.2.2. Exponential distribution

The exponential distribution is a positive random value X with an exponential probability density function. For this law, the probability density function and cumulative density function are :

$$\begin{aligned}\frac{dP_X}{dx} &= \lambda e^{-\lambda x}\\ P(X \leq t) &= \int_0^t \frac{dP_x}{dx} dx\\ &= 1 - e^{-\lambda t}\end{aligned}$$

The parameter of the law (λ) can be characterized by the following statistics:

$$\begin{aligned}\text{Mean : } E[X] &= \frac{1}{\lambda}\\ \text{Variance : } \sigma_X^2 &= \frac{1}{\lambda^2}\end{aligned}$$

D.2.3. Delayed Exponential Distribution

If random value only takes value in the interval $]\mu, \infty[$, the law is generalized using the following density :

$$\frac{dP_X}{dx} = \lambda e^{-\lambda(x-\mu)}$$

The mean is then shifted, but the variance remains the same.

$$\begin{aligned}\text{Mean : } E[X] &= \frac{1}{\lambda} + \mu\\ \text{Variance : } \sigma_X^2 &= \frac{1}{\lambda^2}\end{aligned}$$

D.2.4. Weibull Distribution

The Weibull Distribution W is a random value taking value in the interval $]\tau, \infty[$, which generalizes the exponential delayed distribution, with the following probability density and cumulative density functions

:

$$\begin{aligned}\frac{dP_W}{dw} &= \frac{\alpha}{\beta}(\frac{w-\tau}{\beta})^{\alpha-1}e^{-(\frac{w-\tau}{\beta})^{\alpha}} \\ P(W \leq x) &= \int_0^x \frac{dP_W}{dw}dw \\ &= 1 - e^{-(\frac{x-\tau}{\beta})^{\alpha}}\end{aligned}$$

The parameters are :

1. α shape factor
2. β scale factor
3. τ localization factor

When the shape factor (α) is equal to 1, we retrieve the exponential distribution. In this case, $\beta = \frac{1}{\mu}$. τ is also describing the starting point of the distribution.
These parameters can be characterized by the following statistics:

$$\begin{aligned}\text{Mean : } E[W] &= \beta.\Gamma(\frac{1}{\alpha}+1)+\mu \\ \text{Variance : } \sigma_W^2 &= \beta^2.[\Gamma(\frac{2}{\alpha}+1)-\Gamma(\frac{1}{\alpha}+1)^2] \\ \text{where } \Gamma(x) &= \int_0^{\infty} e^{-t}.t^{x-1}dt\end{aligned}$$

D.2.5. Pareto Distribution

The Pareto is a random probability function Π taking its value in $[a,\infty[$ with the following probability density and cumulative density functions are :

$$\begin{aligned}\frac{dP_\Pi}{dp} &= \frac{\alpha}{a}(\frac{a}{p})^{\alpha+1} \text{ if } p \geq a \\ &= 0 \text{ otherwise} \\ P(\Pi \leq x) &= \int_0^x \frac{dP_\Pi}{dp}dp \\ &= 1-(\frac{a}{x})^{\alpha+1}\end{aligned}$$

This function can be characterized by the following statistics:

$$\begin{aligned}\text{Mean : } \mathbb{E}[\Pi] &= \frac{\alpha}{\alpha-1}a \quad \alpha > 1 \\ \text{Variance : } \sigma_\Pi^2 &= \frac{\alpha}{\alpha-2}a^2 \quad \alpha > 2\end{aligned}$$

D.2.6. Gamma Distribution

The probability density function of the gamma distribution is

$$F_\Gamma(x;k,\theta) = x^{k-1}\frac{e^{-x/\theta}}{\theta^k\,\Gamma(k)} \text{ for } x > 0$$

where $k > 0$ is the shape parameter and $\theta > 0$ is the scale parameter of the gamma distribution. This function can be characterized by the following statistics:

$$\begin{aligned}\text{Mean : } \mathbb{E}[\Gamma] &= k\theta \\ \text{Variance : } \sigma_\Gamma^2 &= k\theta^2\end{aligned}$$

D.2.7. Log Normal distribution

The log-normal distribution has probability density function

$$f(x;\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-(\ln x-\mu)^2/2\sigma^2}$$

for $x > 0$, where μ and σ are the mean and standard deviation of the variable's logarithm. The expected value is $\mathrm{E}(X) = e^{\mu+\sigma^2/2}$ and the variance is $\mathrm{var}(X) = (e^{\sigma^2} - 1)e^{2\mu+\sigma^2}$..

D.2.8. Erlang E(r,λ)

The Erlang distribution is derived from the exponential density and is the sum of n exponential distribution with mean λ/r.

$$f_{E(r,\lambda)}(t) = \frac{\lambda(\lambda t)^{r-1}e^{-\lambda t}}{(r-1)!} \tag{D.1}$$

The laplace transform of this distribution is

$$f^*_{E(r,\lambda)}(s) = \frac{\lambda^r}{(s+\lambda)^r} \tag{D.2}$$

D.2.9. Gaussian models

D.2.9.1. Classical bell model

The continuous probability density function of the normal distribution is the Gaussian function

$$\varphi_{\mu,\sigma^2}(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) = \frac{1}{\sigma}\varphi\left(\frac{x-\mu}{\sigma}\right), \quad x \in \mathbb{R},$$

where $\sigma > 0$ is the standard deviation, the real parameter μ is the expected value, and

$$\varphi(x) = \varphi_{0,1}(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2}, \quad x \in \mathbb{R},$$

is the density function of the "standard" normal distribution, i.e., the normal distribution with $\mu = 0$ and $\sigma = 1$.

As a Gaussian function with the denominator of the exponent equal to 2, the standard normal density function φ is an eigenfunction of the Fourier transform.

Some properties of the probability density function are:

- The density function is symmetric about its mean value μ.
- The mean μ is also its mode and median.
- The inflection points of the curve occur at one standard deviation away from the mean, i.e. at $\mu-\sigma$ and $\mu+\sigma$.

D.2.9.2. Bounded model

When the values from the classical bell model are bounded, the distrubution can be bounded between b_l and b_h. In that case, the probability function has to be rescaled. The scaling factor can be written

$$S = \int_{b_l}^{b_h} \exp\frac{-u^2}{\sigma} du$$

D.2.9.3. Bivariate and multivariate gaussian model

More generally, than the classical bell curve, a random variable with the density:

$$p_X(x_1,...,x_n) = \frac{\sqrt{|A|}}{(2\pi)^{\frac{n}{2}}}.\exp\Big(-\frac{1}{2}\sum_{j,k}(x_j-m_j)a_{jk}(x_k-m_k)\Big)$$

where $m=(m_1,...,m_n)\in\mathbb{R}$ is a gaussian multivariate.

Acronyms

ARIMA AutoRegressive Integrated Moving Average
AVBDC Absolute Volume Based Dynamic Capacity
ATM Asynchronous Transfer Mode
BW Bandwidth
BMAP Batch Markovian Arrival Process
CAC Call Admission Control / Connection Admission Control
CCITT Comite Consultatif Internationale de Telegraphie et Telephonie
CDF Complementary Distribution Function
CRA Continuous Rate Assignment
DES Discrete Event Simulation
DVB-RCS Digital Video Broadcasting - Return Channel via Satellite
DVB-S Digital Video Broadcasting - Satellite
DVB-S2 Digital Video Broadcasting - Satellite version 2
ETSI European Telecommunications Standards Institute
EM Expectation Maximization
FCA Free Capacity Assignment
GEO Geostationary Earth Orbit
IDC Index of Dispersion of Counts
IDI Index of Dispersion of Intervals
ITU International Telecommunication Union
ITU-T ITU Telecommunications sector
LAN Local Area Network
IAT Inter Arrival Time
IP Internet Protocol
LST Laplace-Stieltjes Transform
LEO Low Earth Orbit
MAP Markov Arrival Process
MMPP Markov Modulated Poisson Process
NCC Network Control Center
PASTA Poisson Arrivals See Time Average
PH Phase Type
PDF Probability Density Function
QoS Quality of service
RBDC Rate Based Dynamic Capacity
RM Resource Management
TDMA Time Division Multiple Access
TRM Traffic Resource Manager

TCP Transmission Control Protocol

TL Transport Level

VoIP Voice over IP

VBDC Volume Based Dynamic Capacity

WWW World Wide Web

Principal notations

$(.)^{\star}_{\tau}$	Converter from abstract unit to absolute abstract unit	20
$\overline{(.)_{\tau}}$	Converter from absolute abstract unit to abstract unit	20
$\psi = \{T_n, n \geq 0\}$	Simple point process	38
$\Phi^{(i)}$	Traffic stream measured at level i	20
$\oplus$	Addition operator	22
$\top$	Aggregation operator	21
$\sqsupset$	Level conformation operator	23
$\triangle$	Level conversion operator	22
$\mathfrak{A} = (\mathcal{S}, \mathcal{I})$	Abtsract Unit	19
a_i	Application i	12
$f^{\star}(s)$	Laplace-Stieljes trnsfrom of the density $f(s)$	168
ℓ	Level l	18
n_k	Node k	12
s_j	Subscriber j	12
$\mathcal{I}$	Packet inter-arrival time	20
$\mathcal{M}_V(t)$	Data volume metric	27
$N(t)$	Counting process for ψ	38
N_I	Number of type I nodes in a full size test scenario	106
N_{II}	Number of type II nodes in a full size test scenario	106
N_{III}	Number of type III nodes in a full size test scenario	106
N_{IV}	Number of type IV nodes in a full size test scenario	106
N_1	Number of cells of type 1 cells measured for the traffic stream $\Phi^{(4)}$	106
N_2	Number of cells of type 2 cells measured for the traffic stream $\Phi^{(4)}$	106
$\mathcal{P}_\ell$	Packet at level ℓ	14
$Q(z)$	z-transform of queue size distribution	78
$\mathcal{T}$	Packet absolute arrival time	19
$W^{*}(s)$	Laplace transform of waiting time (in queue)	78
$\mathcal{S}$	Packet size	20

Bibliography

[ACW92a] J. Abate, G. Choudhury, and W. Whitt. Asymptotics for steady-state tail probalities in structured markov queueing models, 1992.

[ACW92b] J. Abate, G. Choudhury, and W. Whitt. Exponential approximations for tail probabilities, i: Waiting times, 1992.

[ACW92c] J. Abate, G. Choudhury, and W. Whitt. Exponential approximations for tail probabilities, ii: Sojourn times, 1992.

[AMS82] D. Anick, D. Mitra, and M. M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *the Bell System Technical Journal*, 61(8), 1982.

[AR99] Sofia Anderson and Tobias Ryden. Maximum likelihood estimation of a structured mmpp with applications to traffic modelling. In *Proc. 13th ITC Specialist Seminar on Measurement and modelling of IP Traffic*, pages 45–63, 1999.

[AZN98] R. G. Addie, M. Zukerman, and T. Neame. Broadband traffic modeling: Simple solutions to hard problems. *M-COM*, 3:88–95, 1998.

[BCC$^+$04a] M. Berioli, P. M. L. Chan, C. M. Chao, H. Che, N. Courville, P. Febvre, O. Franzrahe, O. Grémillet, F. Hermanns, Y. F. Hu, A. Jahn, R. Janos, A. Lorelli, C. Párraga, P. Pillai, M. Rohrer, and Z. Sun. Protocol analysis and design report, 2004. D14.

[BCC$^+$04b] M. Berioli, C. M. Chao, P. Chan, H. Che, N. Courville, M. De Sanctis, F. Del Sorbo, P. Febvre, O. Franzrahe, O. Grémillet, F. Hu, A. Jahn, R. Janos, A. Lorelli, J. M uller, C. Párraga, P. Pillai, and Z. Sun N. Riera Dìaz. Wirelesscabin target system specification document, 2004. D9.

[Bel70] R. Bellman. *Introduction to Matrix Analysis*. McGraw-Hill, New York, 1970.

[Ber94] Jan Beran. *Statistics for Long-Memory Processes*. Chapman & Hall, CRC, Boca Raton, 1994.

[BG05a] V. Boussemart and O. Grémillet. Investigation of self-similarity by aggregation of individual web-browsers. In *AIAA ICSSC*, 2005.

[BG05b] V. Boussemart and O. Grémillet. Modelling of web traffic generated by aggregated individual browsers. *Internal DLR memo*, 2005.

[BH00] François Baccelli and Dohy Hong. ”tcp is max-plus linear (and what it tells us on its throughput)”. Technical report 3986, INIRIA, 8 2000.

[BHJ99] E. Brockmeyer, H. L. Haslstrøm, and A. Jensen. *The life and works of A. K. Erlang*. Transactions of the Danish Academy for Technology and Science, 1999.

[Bil97] J. Bilmes. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical report, University of Berkeley, ICSI-TR-97-021, 1997.

[Bou05] Vincent Boussemart. *Investigations on Internet Traffic*. PhD thesis, DRT, University of Valenciennes, 2005.

[Bra68] P. T. Brady. A statistical analysis of on-off patterns in 16 conversations. *the Bell System Technical Journal*, 47(1), 1968.

[BWD+05] H. Bischl, M. Werner, A. Dreher, L. Richard, E. Lutz, J. Bostic, H. Brandt, P. Todorova, F. Krepel, and M. Emmelmann. Atm-based multimedia communication via ngso-satellites. *International Journal of Satellite Communications and Networking,*, Vol. 23(Issue 1):1 – 32, 2005.

[CD83] W. P. Cherry and R. L. Disney. The superposition of two independent markov renewal processes. *Zastosowania mathematyki*, (XVII-4):567, 1983.

[Cin69] E. Cinlar. Markov renewal theory. *Advances in applied probability*, 1:123–187, 1969.

[Cin72] E. Cinlar. *Superposition of point process*, pages 549–606. Number 1. Wiley, New York, 1972.

[CL66] D.R Cox and P.A.W. Lewis. *The Statistical Analysis of Series of Events*. Methuen, London, 1966.

[CL99a] F. Carducci and G. Losquadro. The EuroSkyWay worldwide system providing broadband service to fixed and mobile end-users. *International Journal on of Satellite Communications*, 17:143–154, 1999.

[CL99b] Hyoung-Kee Choi and John O. Limb. A behavioral model of web traffic. In *Seventh International Conference on Network Protocols (ICNP'99)*, page 327, 1999.

[Cla45] A. C. Clarke. Extra-terrestrial relays. *Wireless World*, 51(303 (Oktober)):305–307, 1945.

[Cox67] D. R. Cox. *Renewal theory*. Mathuen, London, 1967.

[CRD99] Dennis P. Connors, Bo Ryu, and Son Dao. Modeling ans simulation of broadband satellite networks - part i: Medium access control for qos provisionning. pages 72–79, March 1999.

[Day95] J.D. Day. The (un)revised osi reference model. *Computer Communications Review*, 25:39–55, 1995.

[DFG+03] M. DeSanctis, O. Franzrahe, O. Grémillet, M. Holzbock, Y .F. Hu, A. Jahn, J. Mueller, F. Kasslatter, A. Rogoyski, and M. Werner. Aeronautical communications for personal and multimedia services via satellite. 2003.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 34:1–38, 1977.

[DZ83] J.D. Day and H. Zimmermann. The osi reference model. *Proceedings of the IEEE*, 71:1334–1340, 1983.

[ETS98] ETSI. Selection procedures for the choice of radio transmission technologies of the UMTS, May 1998. TR 101 112 v3.2.0.

[ETS05a] ETSI. Digital video broadcasting (DVB) second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications, March 2005. EN 302 307.

[ETS05b] ETSI. DVB interaction channel for satellite distribution systems, September 2005. EN 301 790.

[FM98] V. Frost and B. Melamed. Traffic modeling for telecommunications networks. *IEEE Journal on on Selected Areas in Communications*, 16(5):733–748, June 1998.

[FR01] Frank H. P. Fitzek and Martin Reisslein. Mpeg-4 and h.263 video traces for network performance evaluation. *IEEE Network*, 15:40–53, december 2001.

[FW98] Anja Feldmann and Whard Whitt. Fitting mixture of exponentials to long-tails distributions to analyse network performances. *Performance Evaluation*, 1998.

[GJG+99] R. Goyal, R. Jain, M. Goyal, S. Fahmy, B. Vandalore, and S. Kota. Traffic management for tcp/ip over satellite atm networks. pages 56–61, March 1999.

[Gog01] Helmuth Gogl. *Measurement and characterization of traffic streams in high-speed wide area networks*. PhD thesis, University of Munich, 2001.

[GRB05] O. Grémillet, José Radzik, and Michel Bousquet. Investigation of properties of aggregated traffic models: from the applications to the satellite link. In *ICWWC Sienna*, 2005.

[GW94] M. W. Garett and W. Willinger. Analysis, modeling and generation of self-similar vbr video traffic. In *Proc. SIGCOMM '94 Conf.*, pages 269–280, Sept. 1994.

[Hap98] Christian Hapke. Development of cell-level traffic models for selected audio and video sources. Master's thesis, Technical University Berlin, 1998.

[HJLW00] M. Holzbock, A. Jahn, E. Lutz, and M. Werner. Future broadband satellite communications for airliners. In *Jahrbuch der Deutsche Gesellschaft fü Luft- und Raumfahrt (2000) (German Space and Aerospace Society Annual Convent)*, pages on CD–ROM, 2000.

[HKS99] Helmuth Hlavas, Gabriele Kotsis, and Christine Steinkellner. Traffic source modeling. Technical report no. tr-99101, Institute of Applied Computer Science and Information Systems, University of Vienna, 1999.

[HL86] H. Heffes and D. M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on on Selected Areas in Communications*, 4:856–868, 1986.

[Hos84] J. R. .M. Hosking. Modeling persistence in hydrological time series using fractional differencing. *Water Resources Research*, 20:1898–1908, 1984.

[Hur51] Hurst. Long term storage capacity of reservoirs. *Transactions on Am. Soc. Civ. Eng.*, 116:770–799, 1951.

[INT05] The internet traffic archive. `http://ita.ee.lbl.gov`, 2005.

[IT93] ITU-T. Recommendation e.800, March 1993. Terms and definitions related to quality of service and network performance including dependability.

[IT95a] ITU-T. Recommendation i.350, March 1995. General aspects of Quality of Service and network performance in digital networks, including ISDNs.

[IT95b] ITU-T. Recommendation i.356, 1995. B-ISDN ATM Layer Cell Transfer Performance.

[Jac90] V. Jacobson. Compressing TCP/IP headers for low-speed serial links. *Internet Request for Comments*, (1145), 1990.

[Jah04] Axel Jahn. Wirelesscabin: Development and demonstator of wireless access for multimedia services in aircraft cabin, short project summary, 2004.

[JHRW02] A. Jahn, M. Holzbock, N. Riera, and M. Werner. Passenger multimedia service concept via future satellite systems. In *Jahrbuch der Deutsche Gesellschaft fü Luft- und Raumfahrt (2002) (German Space and Aerospace Society Annual Convent)*, pages on CD–ROM, 2002.

[JMBH+05] A. Jahn, V. W. C. Chook M. Berioli, Y. F. Hu, O. Franzrahe, and M. De Sanctis. Results from the wirelesscabin demonstration flight. In *Proc. 14th IST (European Information Society Technologies) Mobile and Wireless Communications Summit*, 2005.

[Kel96] Frank Kelly. *Notes on effective bandwidth*. Royal Statistical Society lecture note series, chapter 8, 1996.

[Ken51] D. G. Kendall. Some problems in the theory of queues. *Journal of the Royal Statistical Society, Series B*, 13(2):151–185, 1951.

[Ken53] D. G. Kendall. Stochastic processes occuring in the theory of queues and their analysis by the method of the imbedded markov chain. *Ann. Math. Statist.*, 24:338–354, 1953.

[KL97] Sunil U. Khaunte and John O. Limb. Statistical characterization of a world wide web browsing session. Technical Report GIT-CC-97-17, 1997.

[Kle76a] L. Kleinrock. *Queueing Systems, Vol.1 Theory*. John Wiley, New York, 1976.

[Kle76b] L. Kleinrock. *Queueing Systems, Vol.2 Computer Applications*. John Wiley, New York, 1976.

[KLL02] Alexander Klemm, Christoph Lindemann, and Marco Lohmann. Traffic modeling of ip networks using the batch markovian arrival process. In *Computer Performance Evaluation / TOOLS*, pages 92–110, 2002.

[KLL03] Alexander Klemm, Christoph Lindemann, and Marco Lohmann. Modeling ip traffic using the batch markovian arrival process. *Performance Evaluation*, 54:149–173, 2003.

[KM00] G. Kotsis and M. Marot. Application and user behaviour characterisation. D1.1, Deliverable of the BISANTE project, 2000.

[KOS97] Sang. H. Kang, Changhwan Oh, and Dan K. Sung. A traffic measurement-based modeling of superposed atm cell streams. *IEICE*, E80-B:434–441, March 1997.

[Lac00] B. Lacaze. *Processus Aéatoires pour les Communications Numériques*. Hermes science, 2000.

[LBT01] Jean-Yves Le Boudec and Patrick Thiran. *Network Calculus*. Springer Verlag LNCS 2050, 2001.

[LH05] O. L ucke and M. Holzbock. Improved multi service traffic modelling and satellite system dimensioning for collectively mobile user groups. In *Proc. 23th AIAA International Communications Satellite Systems Conference*, 2005.

[Lin51] D. V. Lindley. The theory of queues with a single server. *Proceedings Cambridge Phil. Society*, 48:277–289, 1951.

[Lit61] J. D. C. Little. A proof of the queueing formula l=λw. *Operations Research*, 9:383–387, 1961.

[LPA94] San-Qi Li, Sangkyu Park, and Dogu Arifler. Smaq: a measurement-based tool for traffic modeling and queuing analysis. i. design methodologies and software architecture. *IEEE/ACM Transactions on Networking*, 36(8):56–65, August 1994.

[LTWW93] William E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel V. Wilson. On the self-similar nature of Ethernet traffic. In Deepinder P. Sidhu, editor, *ACM SIGCOMM*, pages 183–193, San Francisco, California, 1993.

[Luc91] David M. Lucantoni. New results on the single server queue with a batch markovian arrival process. *Stochastic Models*, 7:1–46, 1991.

[LWJ00] E. Lutz, M. Werner, and A. Jahn. *Satellite Systems for Personal and Broadband Communications*. Springer-Verlag, London, 2000.

[MB03] G. Maral and M. Bousquet. *Satellite Communications Systems*. John Wiley, Chichester, 2003.

[MH87] K. S. Meier-Hellstern. A fitting algorithm for markov-modulated poisson process having two arrival rates. *European Journal of Operational Research*, 29:370–377, 1987.

[MSTBGE99] I. Mertzanis, G. Sfikas, R. Tafazolli, and B.G. B. G. Evans. Protocol architectures for satellite atm broadband networks. pages 46–54, March 1999.

[Neu75] M. F. Neuts. *Probability distribution of phase type*, pages 173–206. University of Louvain, 1975.

[Neu76] Marcel F. Neuts. A versatile markovian point process. *Journal of Applied Probability*, 16:764–779, 1976.

[Neu78] M. F. Neuts. Renewal processes of phase type. *Naval Res. Logist. Quart.*, 25:445–454, 1978.

[Neu81] M. F. Neuts. *Matrix-Geometric solutions in stochastic models : An algorithmic approach*. Baltimore, MD : J. Hopkins University press, 1981.

[NGJV02] C. Parraga Niebla, O. Grémillet, A. Jahn, and S. Venditti. Design and implementation of a real time traffic emulation system for multimedia satellite communications. 2002.

[NRSV91] I. Norros, J. W. Roberts, A. Simonian, and J. Virtamo. The superposition of variable bit rate sources in atm multiplexers. *IEEE Journal on Selected Areas in Communications*, 9, April 1991.

[ns205] The network simulator - ns-2. `http://www.isi.edu/nsnam/ns/`, 2005.

[O205] Opnet. `http://www.opnet.com/`, 2005.

[Pax94] Vern Paxson. Empirically derived analytic models of wide-area TCP connections. *IEEE/ACM Transactions on Networking*, 2(4):316–336, 1994.

[PF95] Vern Paxson and Sally Floyd. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, 1995.

[Pol57] Felix Pollaczek. Problèmes stochastiques posés par le phénomène de formation d'une queue d'attente à un guichet et par des phénomènes apparentés. *Mémorial des Sciences Mathématiques*, 1957.

[Pos81a] J. Postel. RFC 791: Internet Protocol, September 1981.

[Pos81b] J. Postel. RFC 793: Transmission control protocol, September 1981.

[qLH93] San qi Li and Chia-Lin Hwang. Queue response to input correlation functions: continuous spectral analysis. *IEEE/ACM Transactions on Networking*, 1(6):678–692, December 1993.

[Ram80] V. Ramaswami. The n/g/1 queue and its detailled analysis. *Advances in Applied Probability*, 12:222–261, 1980.

[Ris02] Alma Riska. *Aggregate Matrix-Analytic Techniques and their Application*. PhD thesis, College of William and Mary, Virginia, 2002.

[RMV96] J. Roberts, U. Mocci, and J. Virtamo. *Broadband Network Teletraffic, Final Report of Action COST 242*. Springer, 1996.

[Ryd94] Tobias Ryden. Parameter estimation for makov modulated poisson process. *Stochastic Models*, pages 795–829, 1994.

[Ryd96] Tobias Ryden. An em algorithm for estimation in markov-modulated poisson processes. *Computational Statistics and Data Analysis*, pages 431–447, 1996.

[Ryu99] Bo Ryu. Modeling ans simulation of broadband satellite networks - part ii: Traffic modelling. pages 48–56, July 1999.

[Sch01] Hans-Peter Schwefel. Behavior of tcp-like elastic traffic at a buffered bottleneck router. In *IEEE Infocom*, 2001.

[SLTG00] D. Staehle, K. Leibnitz, and P. Tran-Gia. Source traffic modeling of wireless applications. Technical Report 261, University of Würzburg, 6 2000.

[Spa02] Kathleen Spaye. *Superposition of Markovian traffic sources and frame aware buffer acceptance*. PhD thesis, University of Antwerpen, 2002.

[SqL93] Chia-Lin Hwang San-qi Li. Queue response to input correlation functions: discrete spectral analysis. *IEEE/ACM Transactions on Networking*, 1(5):522–533, October 1993.

[STC95] M. Samuelides, L. Touziller, and Y. Caumel. *Modeles Aleatoires*. Polycopie Supaero, 1995.

[SW86] K. Sriram and W. Whitt. Characterizing superposition arrival processes in packet multiplexers for voices and data. *IEEE Journal on on Selected Areas in Communications*, 4:833–846, 1986.

[Tak55] L. Takàcs. Investigation of waiting time problems by means of a poisson process. *Publs. Math. Inst. Acad. Sci. Hung.*, 6:101–289, 1955.

[Tan96] Andrew S. Tanenbaum. *Computer Networks; 3rd edition*. Prentice Hall, 1996.

[VA99] Darryl Veitch and Patrice Abry. A wavelet based joint estimator of the parameters of long-range dependence. *IEEE Transactions on Information Theory*, 45(3):878–897, 1999.

[Var01] Andras Varga. The omnet++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference (ESM'2001)*, June 2001.

[Wat93] K. Watkins. *Discrete event simulation in C*. McGraw-Hill, 1993.

[Wer02] Markus Werner. *Routing and Dimensioning in Satellite Networks with Dynamic Topology*. PhD thesis, TU München, 2002.

[Wir97] P. E. Wirth. The role of teletraffic modeling in the new communications paradigms. pages 86–92, August 1997.

[WL98] Markus Werner and Erich Lutz. Multiservice traffic model and bandwidth demand for broadband satellite systems. In *Evolution of Satellite Personal Communications from Second to Future Generations Systems, Temporary Documents of European COST 252 Action*, 252TD(98)30, Venice, Italy, November 1998. COST 252 Management Committee. Identical with the same-titled paper at EMPS '98.

[Wol82] R. Wolff. Poisson arrivals see time averages. *Operations Research*, 30:223–231, 1982.

List of Tables

List of Figures

Appendices

Analyse des modèles de trafic agrégé
pour les télécommunications multiservices par satellite

Le thème de la thèse est la modélisation du trafic de données transitant sur des systèmes de communications par satellite. Les services concernés sont multiples (voix, navigation sur la toile, courrier électronique, audio/vidéo) et s'effectuent au travers d'un lien satellitaire.
Après un chapitre introductif précisant le cadre de l'étude, le second chapitre introduit un modèle hiérarchique pour la description du trafic: le modèle en pyramide. Ce modèle est basé sur différents niveaux d'abstraction dénommés application, souscripteur, nœud et transport. On y introduit également des opérateurs qui manipulent les paquets au sein de chaque niveau.
Le chapitre 3 est dédié à l'étude des modèles de trafic. On y aborde certains modèles théoriques ainsi que les modèles choisis pour la description des applications. Diverses méthodes pour la détermination des modèles sont envisagées.
Dans le chapitre 4, un scénario de référence pour les communications aéronautiques est présenté. Le modèle hiérarchique en pyramide est utilisé et les opérateurs peuvent être décrits de façon précise. Ensuite les caractéristiques des principaux modules d'un simulateur le reproduisant sont décrites synthétiquement.
Les chapitres 5, 6 et 7 présentent les résultats obtenus aux niveaux souscripteur, nœud et transport. Au niveau souscripteur, les propriétés des applications sont vérifiées et des modèles agrégés sont dérivés. L'utilisation d'un modèle agrégé au niveau nœud étant d'un intérêt supérieur, les résultats sont ensuite étendus à ce niveau. Les équivalences entre modèles agrégés et précis sont démontrées sous différents points de vue (volume de donnée, distribution des files d'attente, …). Finalement, le dernier chapitre résume les résultats obtenus.

Mots clés : Systèmes de communication satellitaire – Modèles de trafic – Ingénierie du trafic – Métrologie – Réseaux large bande – Réseaux de communication multiservices – Processus de Markov Poissonien en salve – Algorithme EM (Expectation Maximisation)

Analysis of aggregate traffic models for multiservice satellite communications

This thesis deals with the modelling of data traffic transmitted over satellite communications systems. Various type of traffic are considered from a broad range of applications (voice, web surfing, email, audio/video broadcast) and data are exchanged over a satellite link.
After an introduction chapter setting the framework, the second chapter introduces a hierarchical model for the modelling of traffic: the pyramid traffic model. This model is based on different abstraction levels called application, subscriber, node and transport. Some operators are also introduced to describe the modifications of packet inside the different levels.
Chapter 3 is devoted to traffic models and some theoretical models are described together with the selected models describing the applications under study. Different methodologies for the models parameter determination are reviewed.
In chapter 4, a study case scenario for aeronautical communications is introduced. The hierarchical pyramid traffic model can be applied and the operators are described accurately. The principal modules of a simulator describing this case are described.
Chapter 5, 6, 7 summarize the results gathered at the subscriber, node and transport level. At subscriber level, the match between the models and the real application traffic is checked. Aggregate models bring more benefit at the node level and are hence described at this level. The equivalency of aggregate and precise models is demonstrated from different point of view (identical volume, queuing statistics). At last, chapter 8 summarize the results obtained.