BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
DEPARTMENT OF NETWORKED SYSTEMS AND SERVICES

DISQUISITION ON PRICING OF TELECOMMUNICATION SERVICES AND BILLING SYSTEM FUNCTIONALITIES

Ph.D. Theses
by

Bálint Dávid ARY

Research supervisor:
Prof. Sándor IMRE
Department of Networked Systems and Services

Budapest, 2013

PH.D. SCHOOL ON IT SCIENCE
AT
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS

BUDAPEST, HUNGARY
MAY, 2013

ary.balint@isolation.hu

Alulírott Ary Bálint Dávid kijelentem, hogy jelen disszertációt meg nem engedett segítség nélkül, saját magam készítettem, és a disszertációban csak a megadott forrásokat használtam fel. Minden olyan részt, melyet szó szerint, vagy azonos értelemben, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

A dolgozat bírálatai és a védésről készült jegyzőkönyv a Budapesti Műszaki és Gazdaságtudományi Egyetem Villamosmérnöki és Informatikai Karának dékáni hivatalában elérhetőek.

I hereby confirm, that the current Ph.D. thesis was written by me, without any non-authorized help. Besides my own work experience, all the sources that were used are listed at the end of this document.

The reviews of the dissertation and the report of the thesis discussion are available at the Dean's Offce of the Faculty of Electrical Engineering and Informatics of the Budapest University of Technology and Economics.

Bálint Dávid ARY
Budapest
May 3, 2013

Summary

The mobile telecommunication market has gone through significant changes since the first commercial cellular network was launched in 1979. As mobile phone penetration approached or passed 100% in most developed countries, the judgment of the service has been slightly lowering from *premium category* to *generally available*, thus mobile network operators are continuously introducing novel value added services, complex, attractive and customizable tariff packages to sustain the level of service and the temptation of the market for consumers and investors. These new services brought new players to the market and altered the legacy business models. However, the majority of the income is still generated by the subscribers and managed by the point of sale (POS) and billing system of the provider. My preliminary research objective was to examine the effects of novel services and paradigm change on the existing billing architecture and billing systems.

New services introduced to mass market might have a substantial effect on the performance of the billing system. I have created mathematical models and simulations to calculate the required processing power, the number and distribution of the partial CDRs as well as the required size of the database to store them. I have calculated and simulated the amount of unit reservation messages used to charge and rate pre-paid services.

Taking the existing standards and implementations into consideration I have examined the possibilities to reduce the amount of administrative overhead in online charging, thus lowering the required CPU power and network usage. I have created a mode-switching model, where the system dynamically switches between online and offline mode and introduced two new extensions for the regular online charging. Calculations and simulations were introduced to confirm my results.

A major part of my research was to find a new and highly flexible rating model. The main functionalities of the offline charging were identified and a new, state-graph based model was introduced. The new model aids the Advice of Charge functionality, where the price of the service is estimated before the actual consumption. A simulation was created to confirm the estimated results.

Magyar nyelvű összefoglaló

A mobil telekommunikációs piac lényeges változásokon ment keresztül mióta az első publikus és mindenki számára elérhető cellás mobiltelefon rendszert beüzemelték 1979-ben. Ahogy a fejlett államokban a mobil telefon penetráció megközelítette (sőt legtöbb esetben meg is haladta) a 100%-ot, úgy változott meg az emberek szemlélete a szolgáltatással kapcsolatosan. A kezdeti időkben prémium kategóriás szolgáltatásnak számító rádiótelefont ma sokkal inkább az általános, mindennapi életünkhöz tartozó eszköznek tekintjük. A mobil távközlési szolgáltatók éppen ezért mindent megtesznek, hogy új és vonzó szolgáltatásokkal, személyre szabható tarifacsomagokkal fenntartsák az előfizetők és befektetők érdeklődését és a szolgáltatás prémium színvonalát. Ezen új változások nem csak a technológia, hanem a piachoz tartozó üzleti modellek megújulását is jelentik. Mindazonáltal a bevétel legnagyobb részét még mindig a végfelhasználók generálják, melyet a szolgáltatók az árusító (point of sale - POS) és számlázórendszereik segítségével realizálnak. A disszertáció fő célkitűzése az új telekommunikációs szolgáltatások és a paradigmaváltások hatásának vizsgálata a jelenlegi számlázási architektúrán és számlázórendszer implementációkon.

A bevezetett, vagy bevezetni kívánt új szolgáltatások szignifikáns hatással lehetnek a számlázórendszer teljesítményére. Olyan modelleket és szimulációkat dolgoztam ki, amelyekkel kiszámítható, vagy megbecsülhető a szükséges számítási kapacitás, a parciális CDR-ek darabszáma és eloszlása, a tárolásukhoz szükséges adatbázis mérete, valamint kiszámoltam és szimuláltam a pre-paid szolgáltatások árazásához szükséges egységlefoglalások számát.

Megvizsgáltam annak a lehetőségét, hogy hogyan lehet a pontos árazást a jelenleginél kevesebb hálózati forgalommal és processzor teljesítménnyel biztosítani. Kidolgoztam egy mód-váltó modellt, ahol a rendszer dinamikusan vált online és offline számlázás között, valamint bemutattam két új kiegészítést az online számlázáshoz. A számítási eredményeket szimulációval igazoltam.

A kutatásaim egyik fő gócpontja egy olyan új árazási alrendszer kidolgozása volt, amely biztosítja a flexibilis árazást és lehetőséget ad újabb szolgáltatások bevezetésére. Meghatároztam az offline árazás főbb lépéseit, valamint kidolgoztam egy új, állapotgráf alapú árazási modellt. A modell lehetőséget ad a hívásdíj előrejelzésre, melyet számításokkal és szimulációval igazoltam.

Biography

Bálint ARY was born in 1981 in Pécs, Hungary. He received his MSc degree in IT from the Budapest University of Technology and Economics in 2005. He started the PhD school on IT science at the same year at the same university. First, he had worked as a project manager at the Mobile Innovation Center for four years, where the scope of the project was to design and develop a flexible rating and charging system. He was also a subject matter expert at Amdocs Hungary and worked as a billing system developer for Pannon GSM (now Telenor). His main areas were the event processing and rating modules and later on the different infra and operational related subsystems. From 2009 he works at Vodafone Hungary as a Business Analyst, and from 2010 as a Solution Analyst. Currently he is responsible for clarifying and designing the IT related requirements for the different business ideas.

Acknowledgement

I would like to thank all the help and guidance I have received from my research supervisor, as well as all the knowledge and experience for all my colleagues and whom I have worked together with. For my family and friends I would like to thank all the faith and endurance.

Keywords

charging billing rating mobile telecommunication post-paid pre-paid

Contents

List of Figures

List of Tables

List of Abbreviations

3G	Third Generation
3GPP	3rd Generation Partnership Project
AoC	Advice of Charge
API	Application Programming Interface
ASN.1	Abstract Syntax Notation One
ATM	Automated Teller Machine
BI	Business Intelligence
BR	Branded Reseller
CAC	Call Admission Control
CAMEL	Customized Application for Mobile Network Enhanced Logic
CAPEX	Capital Expenditure
CDR	Charging Data Record or Call Detail Record
CM	Customer Management
CRM	Customer Relationship Management
CSR	Customer Sales Representative
DWH	Datawarehouse
EDR	Event Detail Record
EOD	End-of-Day
FIFO	First In, First Out
FQPC	Fully Qualified Partial CDR
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HLR	Home Location Register
IC	Interconnect
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IN	Intelligent Node
IP	Internet Protocol
IRSF	International Revenue Share Fraud
ISDN	Integrated Services Digital Network
LAN	Local Area Network

LDC	Long Duration Call
LTE	Long Term Evolution
MMS	Multimedia Messaging Service
MSC	Mobile Switching Center
MSISDN	Mobile Subscriber ISDN Number
MVNO	Mobile Virtual Network Operator
NGN	Next Generation Network
NMT	Nordic Mobile Telephone
NRTRDE	Near Real-Time Roaming Data Exchange
OC	One Time Charge(s)
OCS	Online Charging System
OPEX	Operational Expenditure
OSS	Operation Support System
PDG	Packet Data Gateway
POS	Point of Sale
RC	Recurring Charge(s)
RPC	Reduced Partial CDR
SA	Service Aggregator
SGSN	Serving GPRS Support Node
SIM	Subscriber Identity Module
SIP	Session Initiation Protocol
SMS	Short Message Service
SP	Service Provider
TAP	Transfer Account Procedure
TISPAN	The Telecoms & Internet converged Services & Protocols for Advanced Networks
TTM	Time to Market
UC	Usage Charge(s)
UMTS	Universal Mobile Telecommunications System
WAP	Wireless Application Protocol
WLAN	Wireless LAN
WSP	Wholesale Partner
xDR	CDR or EDR
XML	Extensible Markup Language

Chapter 1

Background

This section summarizes the background of the research. Section 1.1 gives a short overview about the history of charging and rating in mobile telecommunication networks in Hungary. Section 1.2 summarizes the income sources and expenses of a mobile telecommunication company and lists the possible charge types and business models. Section 1.3 explains the different terminology and definitions, while Section 1.4 shows the generic processes and architecture of the billing systems. In Section 1.5 I will detail the different standardizations and regulations, in Section 1.6 I will try to list the key parameters of these systems, while in Section 1.7 I will list the motivations and main drivers of my research.

1.1 A short history

The story of mobile telecommunication in Hungary started in 1990, when the first analogue, NMT based mobile network (branded as Westel 450) was commercially launched. The corresponding technology allowed the company (Westel Rádiótelefon Kft.) to offer voice based services for its subscribers. The monthly fee of the tariff package was approximately 75% of the average monthly income and included 250 minutes of free talk. After the 250th minute, the unit price of the call was differentiated based on the time of the day (peak, off-peak period) and cost approximately as much as a loaf of bread. The subscriber had to pay for originated and terminated calls as well [54].

In March 1994 two other players entered the market. Both Westel 900 (Westel 900 GSM Mobil Távközlési Rt.) and Pannon GSM (Pannon GSM Távközlési Zrt.) used GSM technology and rapidly grew in terms of subscriber numbers and available services. Although Westel 450 launched its voicemail and news service in 1995, it could not catch up with the novel and luring services of the GSM technology, which lead to the fact that the company slowly lost its subscribers and finally, went out of business in 30th of June 2003 [56].

Meanwhile Westel 900 and Pannon GSM amused their subscribers with new products and possibilities. The prices of the calls went down significantly and mobile terminated calls were no longer charged. The tariff packages got more and more complex, different allowances and discounts were developed and sold. The operators introduced the SMS service in 1995 and the conference call option in 1996 head-to-head. In 1997 both GSM provider launched their pre-paid solution (Domino and Praktikum for Westel 900 and Pannon GSM respectively) for cost sensitive users [31, 30].

In 2000, the first international brand, Vodafone entered the Hungarian market [32]. In terms of services, the three GSM providers introduced WAP, GPRS and MMS in the next three years, thus the pricing logic had to be extended for data calls. In 2003 Pannon GSM launched its branded pre-paid solution called DJuice and Westel 900 launched its first M-Commerce service, which allowed its subscribers to buy Cinema tickets with SMS. In 2005 the UMTS service was commercially launched by the providers. From 2007, a lot of new services were introduced (such as Push-to-talk or Mobile Office) and M-Commerce became a commodity service [31, 30, 32]. As a result, the mobile penetration in Hungary exceeded 100% in April 2007 [48].

Regarding brands: Westel renamed itself to T-Mobile in 3rd of May 2004; the Telenor group become 100% owner of Pannon GSM in 2002, removed the GSM postfix in 2006 and renamed itself to Telenor in May 2010. Vodafone launched the first Hungarian branded reseller (Postafon) in 20th of November 2009 [57], and Pannon completely separated the DJuice brand (creating the second branded reseller) in April 2010 [55] and launched Red Bull mobile in the same year [33]. In 2012 two supermarket brands launched their mobile telecommunication service: Lidl launched Blue Mobile in partnership with T-Mobile [66], while Tesco launched Tesco Mobile in a joint venture with Vodafone Hungary [68]. Postafon, DJuice, Red Bull Mobile and Blue Mobile are more likely branded resellers, while Tesco Mobile is a real MVNO utilizing the network infrastructure of the network operator while maintaining its own IT system.

Currently (as beginning of 2013) T-Mobile Global is the 17th, Telenor Global is the 13th and Vodafone Global is the 2nd biggest provider worldwide in terms of subscriptions [71]. However, T-Mobile Hungary owns 46%, Telenor Hungary owns 31% and Vodafone Hungary owns 23% of the Hungarian market approximately [46]. The mobile subscription penetration is 119%. Telenor Hungary and Vodafone Hungary have chosen Amdocs as a billing system vendor, while T-Mobile is currently rolling out its new billing system, also based on an Amdocs solution [61, 28].

1.2 Business models

In the beginning mobile telecommunication providers offered their own services for their subscribers. As remuneration, the subscribers paid after the consumed service (amount of minutes talked on the network), paid different one-time charges for some activities (activation fee) and paid a regular monthly fee. Even though the scope and number of offered services is widened in the last 10 years, the different charge types paid by the subscribers can be still classified into these three types: recurring charges (RC), one-time charges (OC) and usage charges (UC). These incomes were, and are generated by the billing and point-of-sale (POS) or customer relationship management (CRM) systems of the providers.

However, observing on a bigger scale, the incomes and expenditures of the mobile telecommunication companies have changed significantly. The very essential expenditures are the operational expenses, which are required to operate the required infrastructure, pay the footprint hired for the base stations, pay the salary of the employees, printing the bills and so on. The essential incomes are the RC, UC, OC and late payment fees paid by the subscribers. It can be seen, that the billing and POS systems are one of the most business critical systems in a telecommunication company.

Another significant heading in the monthly statement is the interconnect fee. Most of the network operators are operating their own network, and since the

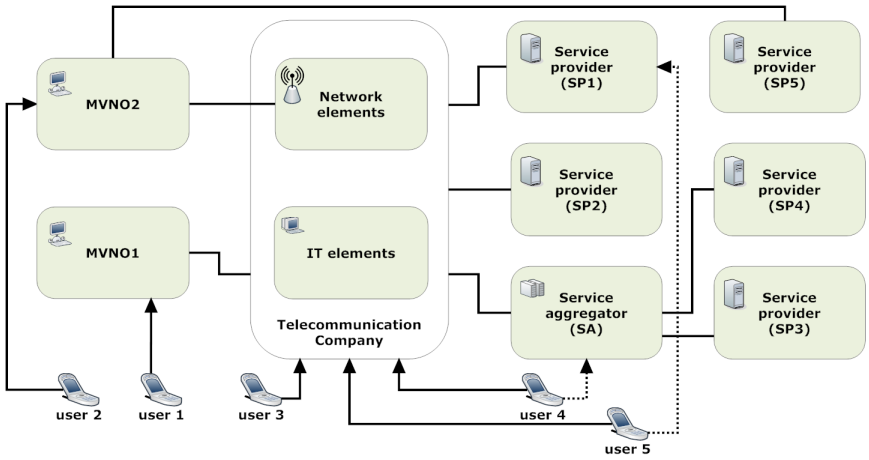

Figure 1.1: Generic service chain and business models

very beginning, the subscribers are able to call another subscriber in another network. Despite of the shared resource consumption (the call reserves resources in the originated and in the terminated network) the price of the call is paid by the subscriber who is initiating the call on her own network. To overcome this problem, the network operators have to pay terminating or interconnect fees to each other based on the amount of terminated calls on the target network. This fee applies also, if a network operator utilizes an underlying network to transmit the call (such as an optic cable of another network operator). The interconnect fees in most cases are calculated by a separate IC billing module.

As new players penetrated the market the business models have changed. The network providers are also offering 3rd party services to their subscribers. In some cases, these value added services are branded with the network providers brand and seamlessly offered to their customers. The subscribers are paying to these services as it would be their network providers' service and the settlement between the 3rd party and the network operator is done later, on a monthly wholesale basis. Also, the network operator itself can offer its services (voice, data calls) for other providers. In such case, the mobile telecommunication service will be branded with the 3rd party's brand, and offered to its customers. A general service chain is represented in Figure 1.1.

In Figure 1.1 the telecommunication company offers its own services and the services offered by SP1-SP4 to its subscribers. The company has a direct agreement with SP1 and SP2, and agreed with a service aggregator (SA), which offers (sales) the service of SP3 and SP4. An example would be the Mobile Office or WebMail for SP1 and SP2 and mobile payment for the service aggregator, where the subscribers can pay with their mobile phones for cinema tickets (SP3) or for parking and highway usage (SP4). Also, the telecommunication company has an agreement with a wholesale partner (MVNO1) to host its subscribers in its own IT system and allow MVNO1 to brand the service. MVNO2 only utilizes the network infrastructure of the telecommunication company and manage and operate its own customer care, POS and billing system. The services offered by SP1-SP4 can be available to the customers of MVNO1, and depending on the type of service, it can be available to the customers of MVNO2 as well. MVNO2 can have its own agreement with service providers (SP5) or service

aggregators. Based on the agreement and settlement between the providers and their subscribers, the following business models can be used in this setup:

Network operator centric business model The network operator centric business model is the legacy business model used from the very beginning. The network operator provides different services (its own, or bought from service providers or service aggregators) and bills the service itself to the subscribers. The subscribers does not necessary know who provided the service and pays the required fees to the network provider. The settlement between the network provider and service provider (aggregator) is done on a monthly, wholesale basis [3].

Service provider or aggregator centric business model In this model the subscribers are still belong to the network operator, but they are aware of the 3rd parties (providers or aggregators) and have a separate financial relationship with them. The fees are paid to the service provider (or aggregator) directly, but the network provider may charge the access fee (SMS or GPRS) to its subscribers [3].

Branded Reseller model In the branded reseller model, the subscribers belong (at least financially) to the 3rd party. The services they are requesting are branded to the brand of the 3rd party, but they are provided by the original network operator or by the different service providers or aggregators. The branded reseller usually utilizes strongly the IT and fully the network infrastructure of the network operator and pays the required fees on a wholesale basis to the operator [72].

Mobile Virtual Network Operator In the MVNO model, the 3rd party utilizes fully the network and slightly (or none at all) the IT infrastructure of the network operator. The MVNO must have a stronger IT layer to serve its subscribers (such as billing, POS or customer care) [72].

The business models represented here are not strictly designed and recognized laws and architectures worldwide. They are more like patterns that can be applied when a network provider and a 3rd party would like to offer a joint service. The partner integration to the network/IT architecture and the rebranding of services requires a lot of effort and shall be compared to the expected additional income when calculating the return of investments. Currently integrating a service provider/aggregator is easier than integrating a wholesale partner, since service providers are on the market for approximately 10 years, while the *outbound* interfaces of the network providers are not usually open and/or mature enough.

1.3 Disambiguation

The industry, the corresponding publications and standards are using a specific terminology. In this section we will give a definition which will be (and have been used so far) in this document.

The words charging, rating, billing and accounting have different meanings in the 3GPP (3rd Generation Partnership Project) and IETF (Internet Engineering Task Force) standards, since they use different approach when defining the architecture: IETF focuses on the protocols, while 3GPP specifies the role of the element [41]. In this document we will define these words based on the 3GPP standards, and we will use the following definitions:

Rating or Pricing The process, when the price of the call is determined based on the required and available information (such as call detail record or call information, rating logic, purchased offers, etc.).

Charging The process whereby a determined value (price) is assigned to a specific account in order to alter it.

Billing The process, where the priced events are grouped together and printed on a payable bill.

Accounting The process, that alters (deduct or increase) the account based on the different charges or payments.

The terminology defines post-paid and pre-paid subscribers. Let us define first post- and pre-paid service consumptions as follows:

Post-paid service consumption The price of the consumed service is paid after the service is consumed. Typically the payment is done on a monthly basis, when the subscriber or customer receives the invoice with the relevant charges. During the service, there is no cover and no assurance that the service will be paid by the subscriber.

Pre-paid service consumption The price of the service is paid before the service is actually used. Since the price is generally unknown at the time of payment, an account is managed in the system for each subscriber. The customers are mainly buying access for these services, and the system deducts the balance (in real-time) during the service consumption, thus the price of the service is covered and assured on the account.

With these definitions we can define the post-paid and pre-paid services as services that can only be consumed in a post-paid or pre-paid way and we can define post-paid, pre-paid and hybrid customers as follows:

Post-paid customer A customer or subscriber that can consume services only in post-paid mode. She receives a monthly invoice and pays her charges at the end of her billing period.

Pre-paid customer A customer or subscriber that can consume services only in pre-paid mode. She tops up her balance/account when she would like to use the service and no regular invoice is issued.

Hybrid customer A customer or subscriber that can consume services with pre- and post-paid mode. Some of the services are paid before the consumption, and some of them are billed to the subscriber at the end of her billing period. The differentiation between the modes can be based on the type of service (the voice service is post-paid, m-commerce is pre-paid) or based on the total amount of service consumed (the first 10 EUR is post-paid, the rest is pre-paid).

To assure, that the cover of pre-paid service consumption is valid, the price of these services has to be deducted from the subscribers' balance. This requires a huge amount of processing capacity and other resources. On the contrary, post-paid service consumptions can be rated and charged with even a significant delay. This difference implies that the system and protocol beneath essentially differs. The industry distinguishes two different rating and charging method as follows [37, 17]:

Offline charging During offline charging, the service is rated and charged after the service is consumed. The delay between the service and the charging can be hours or even days. The steps of offline rating are detailed in Section 1.4.1.

Online charging During online or real-time charging, the service is rated and charged in real-time, before or during the service consumption. The steps of online rating are detailed in Section 1.4.2.

From these definitions, it can be seen, that the pre- and post-paid mode reflects the billing and payment, while the offline and online charging reflects the charging and rating methods. Generally, the payment determines the charging method and pre-paid subscribers are served by the online charging system, while post-paid users are rated and charged offline. However, this is not necessary true, and in some special cases the payment and charging can be mixed.

Since the offline charging systems are more flexible in most cases, some pre-paid services can be rated offline, taking the risk that the price of the service will not be covered by the account. On the contrary, some post-paid service has to be rated online in order to assure real-time control. An example for the first case can be the rating of data or SMS service, if the sophisticated pricing logic cannot be fulfilled by the online system. The bill shock prevention regulation effective from July 2010 requires the real-time rating and charging for roaming data services even for post-paid subscribers [67, 24].

In this document the term *call* will be used as a term for voice calls, SMSs, data sessions or any other communication service which can be initiated by a subscriber using the infrastructure of the mobile network operator.

1.4 IT architecture of a mobile telecommunication company

This section details the general rating and billing architecture of a mobile telecommunication company. Section 1.4.1 and 1.4.2 demonstrates the processes and architecture of offline and online charging respectively. Section 1.4.3 summarizes the additional requirements of the Next Generation services, while Section 1.4.4 tries to paint the big picture and lists the additionally connected systems.

1.4.1 Offline charging

Offline charging is mostly used to rate post-paid subscribers and post-paid services. The subscribers are created, and the required services are sold to the subscribers in the network operators POS, CM (Customer Management) or CRM system. Once the subscriber is created, the relevant services are provisioned to the network elements in order to enable the service.

If the subscriber requests an enabled service, the network elements are administrating and continuously logging the different parameters of the service (such as length of the session, the time when the service was requested, the geographic location and so on) which is finalized when the session has ended [18, 36]. These finalized records are called *call detail record*s, *charging detail record*s or *event detail record*s and often abbreviated as CDRs, EDRs, or more generally xDRs. The serving network element groups these records into files, and sends them to the billing system [8, 6].

It should be noted, that a single service request may generate more than one xDR. Standards define, that each chargeable event may generate a charging detail record (such as mobile originated and terminated calls) and the system may generate an additional CDR if one (or more) of the important chargeable attribute has changed. Moreover, during handover (MSC change) the network element itself changes, thus more than one xDR is generated from more than one source.

With the advent of GPRS, 3G services, IMS (IP Multimedia Subsystem) and LTE (Long Term Evolution), the services are no longer necessary priced according to the length of the call. It is possible, that a rather long session costs a small amount of money, or on the contrary, a short session represents an expensive service. From business point of view, long and expensive service consumptions would have high risks, since the network operator would only be notified when the call has ended, and huge debits could be accumulated without the possibility of any intervention. To overcome this problem, standards define partial CDRs for long calls [5, 7].

On the network element level, when IMS elements (for example) are serving a long session, they are periodically sending interim messages to the corresponding elements [21, 19, 79]. After a few interim messages the network element creates partial CDR if the specified threshold is reached - this can include data volume limit, time (duration) limit, maximum number of charging condition changes or management intervention [10, 11]. Partial CDRs are carrying information about the service consumption since the last partial CDR was issued [21, 19, 79]. Standards define two type of partial CDR. The FQPC (Fully Qualified Partial CDR) holds all the required information, while RPCs (Reduced Partial CDRs) are containing all the mandatory fields [7] and the changes that occurred in any other field relative to the previous partial CDR. The RPCs are converted into FQPCs later on in the network elements, or in the billing system [5, 21]. In this document the term *partial CDR* will be used as an umbrella term for both CDR types.

Once the session is closed, a final CDR is generated and sent to the billing system as regular CDRs. When the partial CDRs are reaching the billing system, the network operator shall decide whether to rate the partial CDRs as unique records, or wait for the rest of information and the final CDR, then aggregate the measured unit(s) and rate the whole session. This latter approach is referred as Long Duration Call (LDC) assembly. The aggregation is based on the record identifier (MSC address and Call Reference Number, PDG address and WLAN Charging ID, IMS Charging Identifier and so on) and on the partial record sequence [5, 7]. The call is rated and charged, when the last chunk arrives to the system.

In case of roaming, the visited network provider (roaming partner) is sending TAP (Transfer Account Procedure) files to the home network. The TAP files are usually sent through a clearing house, which is responsible for sanitizing (validating and converting) the records and dispatching them to the appropriate network provider [65]. These TAP files are arriving to the home network with a significant latency. Usually the records are 2 or 3 days old, but according to the standards [15], the record age can be up to 30 days. This significant delay means high risks to the network operators, thus Near Real-Time Roaming Data Exchange (NRTRDE) was introduced in 2008. The aim of NRTRDE, is that a small, but sufficient amount of information is transferred from the visited network to the home network (in most cases through a clearing house) within four hours to allow the operators to pre-calculate the price of the roaming service

to prevent international revenue share fraud (IRSF) and to initiate the blocking of further roaming calls if required.

Once the files and records are available, they are either sent to the billing system (active mode) by the network elements or gather by the *mediation* module (passive mode) through an offline, file based protocol [8, 6]. In either case, mediation will be responsible for correlating the partial CDRs and converting, filtering and dispatching the (final) CDR and TAP records. The conversation is required, since each and every network element might have its own CDR format (ASN.1, XML, binary, plaintext) with different attributes and values. Since not all of the CDRs are rated, mediation will filter out some of the records (for instance CDRs are generated for mobile originated and terminated SMSs, but a network provider usually rates and charges the mobile originated SMS for regular users). Once the records are formatted and sanitized, they are dispatched to different modules for further processing. Usually the TAP and NRTRDE records are created at this point for visited roaming users and sent to the clearing house. Besides the rating and charging module, the different data warehouse systems and business intelligence modules are also fed from the mediation.

The rating and charging module is responsible for calculating the price of the service from the information stored in the records, the rating logic of the purchased tariff packages and discounts of the customers and the accumulated usage information of the subscribers in the given billing period. The process is started with the *guiding* process, by analyzing the call scenario and the participating telephone numbers in the record (mobile-to-mobile call, call forwarding, call waiting, conference call, value added calls, etc.) and finding the relevant customer or subscriber in the database. Once the customer is found, the process queries the relevant purchased tariff package, discounts, allowances and accumulated usage information stored in the database. The *rating* engine is then responsible for calculating the price of the service from this information, and charging the value of the call on the subscriber's account. The rating or pricing logic shall take the accumulated usage information (total number of sent SMSs in the given month so far for example) and the fields of the CDR into consideration, and apply the tariff package and the different allowances and discounts on them. Once the price is calculated, the system writes the rated call detail record in the database and updates the accumulation accordingly [59, 42].

The guiding and rating modules of a billing system are daemon processes that are up and running continuously during the day. However, in most implementations the offline billing system is restarted around midnight (when the traffic is relatively low) to allow the different reference updates, new rating logics and new developments to become effective and to perform different maintenance tasks. The restarting process is a relative long (1-3 hours) and well controlled process, and often referred as end-of-day, or EOD. The time, when the system is up and running is called *event processing window* in the corresponding terminology.

One very powerful feature of the offline billing system can be the ability of re-rating. Although calls are rated and charged on the customers' account, there is a possibility to re-calculate the price of the calls if the invoice is not yet sent to the customers (if the billing process did not run so far). The re-rating process is a very useful and seamless tool, to fix different implementation errors (pricing logic and/or parameterization) or to apply retroactive discounts for the customers (the price of every call is less than originally, if the total number of SMSs sent reaches a predefined limit). The re-rating process is usually performed during the EOD session.

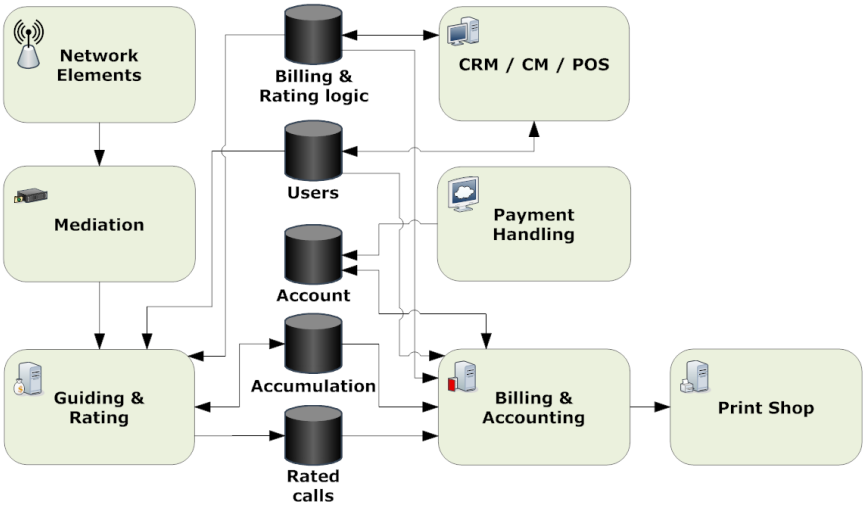

Figure 1.2: Generic offline billing system architecture

The batch process, which summarizes and aggregates the charged calls and creates the final (monthly) bill for the customer is called *billing*. The process sometimes applies different billing discounts on the final invoice based on the purchased offers. Once billing is executed, the charges are committed on the invoice (re-rating cannot be run for these calls from now on) and aggregated and printed on the bills. The billing process is a relatively long and resource consumption process, thus in most cases the network operators are creating more than one billing cycle and grouping the different customers to these cycles for load balancing purposes. The different bill cycles may have different start and end dates and thus, the different billing processes can be executed with a few days difference, allowing the network operator to operate with a smaller hardware setup and higher utilization.

The output of the billing process is the amount due and the printable invoice file per customer, the latter one is printed by a 3rd party print shop in most cases. The account balance and bill image is usually available from the CRM system as well. The structure, content, quality and understandability of the received invoice is one of the key metrics used to measure customer satisfaction. The invoice has to reflect the customer structure and the different charges, and has to be compliant with the local financial regulation, and in parallel, it has to remain understandable for the customers to reduce the number of complaints.

The customers are settling their bill through the payment handling module (*accounts receivable* or *post billing* module). This module handles the different payment channels (direct debit, ATM, check) and books the paid amount of money to the appropriate account. Usually the late payment fees are also calculated here if the customer does not pay the due amount in time, additionally the different collection activities are triggered from this module (notification, suspension, collection, etc.).

The general architecture of the offline charging is represented in Figure 1.2. The main purpose of the modules can be summarized as follows:

Network Elements The network elements are responsible for serving the requests and administrating the duration and other parameters of the calls.

At the end of the session the corresponding xDR is written and grouped together with other xDRs.

Mediation The mediation element is responsible for collecting the different files from the network elements. Once the records are collected, it is responsible for dropping the unnecessary records (such as the mobile terminated SMS record), formatting the rest if required and dispatching them to the appropriate modules.

Guiding and Rating It looks up the customer for the xDR in the database and determines the price of the call based on the rating logic, the purchased offers and the accumulated usage information. The rated calls are written to a database as well as the accumulation is updated in the corresponding table.

Billing and Accounting Based on the rated calls, the accumulated usage information and billing logic, it calculates the final invoiced amount and books it to the customers' account on a monthly basis. The bill image is sent to the print shop and available from the CRM module.

Print Shop Prints and mails the bill to the appropriate customers.

CRM, CM or POS It is responsible for creating new customers and selling different offers and discounts to them that modify the rating or billing logic used by the rating or billing module. It also helps the CSRs (Customer Sales Representatives) to handle different complaints by allowing them to check the account history and bill images of the customers.

Payment Handling The payment handling module is responsible for handling the different payments, channels and calculating the late payment fees. It is often responsible for monitoring the accounts and triggering the different collection processes.

The offline charging system is a real IT product and beyond the mediation module there is no real standardization, thus the names and actual roles of the modules may vary from vendor to vendor.

1.4.2 Online charging

Online charging is mostly used to rate pre-paid subscribers or services, that require real-time charging and/or price calculation. The subscribers are created, and the required services are sold to the subscribers from the same POS or CRM system that is used in offline charging. Online charging determines the price of the services in real-time, and in case of pre-paid subscribers it plays a major role in call admission control (CAC).

When the early mobile telephony and GSM was introduced, online charging was managed by the serving network elements. As novel services were introduced and the rating (pricing) logic of these services got more and more complex, the need for a centralized pre-paid billing platform emerged. Currently in most operators' system an intelligent node (often referred as IN) is responsible for managing and charging the pre-paid subscribers [1, 2, 9]. Most of these systems have out-of-the-box interfaces to the serving network elements (MSCs, SGSNs, etc.) to allow the system to enable, deny or tier-down the user initiated services.

Instead of files, the online charging is using online interfaces to transmit the details of the calls and the price of the service is deducted from the subscribers'

account when the service is requested (or being requested) [9, 12, 29]. The subscriber details, rating logic and accumulation is used as it is used for offline charging, but due to the real-time requirement of the system, the rating logic is usually simpler than in the offline case to allow easy as fast calculation. The method slightly differs for event (SMS, MMS, etc.) and for session based (Voice, Data) services [5, 10].

Event based services (such as SMS, MMS, Mobile payment or e-Gambling) allows easier rating mechanism. Once the user would like to consume the service, the pre-paid platform rates the service in advance and if the subscriber's balance is above this value then the call is authorized and the value of the service is deducted from the balance. If the subscriber does not have enough money on her account, then the call is rejected during the call admission control process [9, 12, 29].

The main problem with the session based services (such as voice call, GPRS or other data session, video telephony, and so on) is that the price is unknown until the service has ended, since the price is highly dependent on the length of the call (the length of the call is not necessary restricted or refers to the length of session in minutes, but to the length of session in the measured unit – e.g.: the amount of kilobytes transferred). The legacy approach was to deduct the balance only after the particular call has ended, but this method clearly carries the risk, that the account of the given subscriber does not cover fully the price of the service [20, 43]. Nowadays the reservation and rating is done in smaller chunks, allowing the operator to gain control over the long services and eliminating or lowering the before mentioned risk [63, 5]. If the subscriber's balance is relative low, the rating system shall be capable to determine the possible length of the call (or chunk) from the available balance. This mechanism is called *inverse rating*.

For a fairly simple rating logic (for example, the price of the service is 0.2 *credit/unit*) the implementation of inverse rating is fairly simple. However, as the rating logic gets more and more complex, the calculation of call length from the available balance gets harder and harder. Most of the online charging systems are offering a framework or model, where the rating logic can be implemented. In some cases, this framework assures the existence of inverse rating by sacrificing some of the flexibility of the pricing logic. Generally speaking, to assure a highly flexible pricing logic (such as different allowances, tiered discounts and subscriber specific discounted periods), we have to disclaim the existence of inverse rating.

Since online charging requires real-time communication with the network elements, network providers often denies roaming services from their pre-paid subscribers. However, roaming can be solved, if CAMEL (Customized Application for Mobile Network Enhanced Logic) services are installed on both networks. In such cases, the signaling and traffic is routed to the home network to run authorization, call admission control and other low-level processes.

Once the call has ended and the price of the call is charged on the pre-paid subscribers account, the online charging system generates an xDR for billing and complaint handling purposes. The xDR is loaded to a rated event database table through the mediation (and in some cases through the offline billing system). In case of post-paid users, these xDRs are used to charge the (rated) price on the post-paid account and later used for billing.

The pre-paid account is managed by the payment handling module through the online charging system. Users are allowed to top-up their balance any time and thus, billing cycles does not really exist. The different monthly based

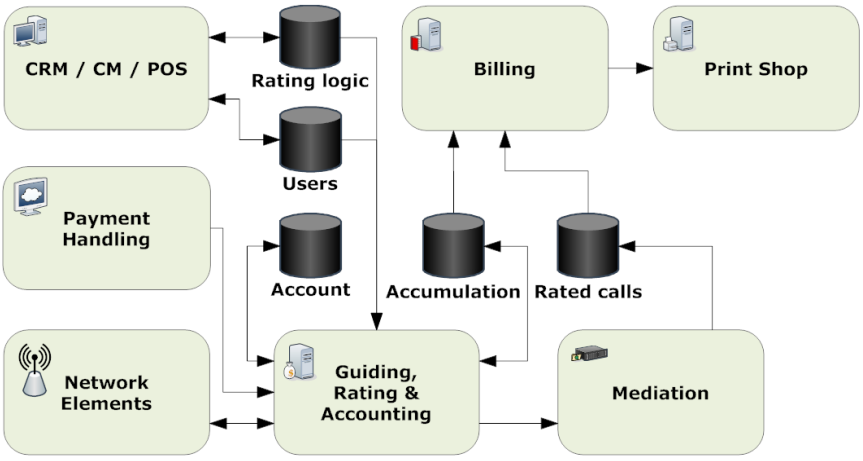

Figure 1.3: Generic online billing system architecture

discounts (30 free SMS in a month) is usually aligned to post-paid billing cycles, but in case of pre-paid services, the validity of such allowances are aligned to calendar months instead.

The general architecture of the online charging is represented in Figure 1.3. The main purpose of the modules can be summarized as follows:

CRM, CM or POS It is responsible for creating new customers and selling different offers and discounts to them that modify the rating logic. It also helps the CSRs (Customer Sales Representatives) to handle different complaints by allowing them to check the account history and the rated calls of the customers.

Network Elements The network elements are responsible for serving the requests, negotiating call admission control with the online charging system and administrating the duration and other parameters of the calls for xDR creating purposes.

Guiding, Rating and Accounting It looks up the customer from the database, calculate the price of the call and allow/deny the service request (call admission control). During the service (in case of session based service) it is responsible for monitoring and modifying the account and tear-down the service if the subscriber's account does not cover the price of the service. An xDR is created after the service has ended and sent to the mediation module for further processing.

Mediation The mediation element is responsible for collecting the different files from the network elements and from the online charging system. Once the records are collected, it is responsible for dropping the unnecessary records (such as the record from the network element), formatting the rest if required and dispatching them to the appropriate modules.

Billing Billing is responsible for creating an invoice based on the rated calls (and accumulation) and sending the created files to the print shops.

Print Shop Prints and mails the bill to the appropriate customers.

Payment Handling The payment handling module is responsible for handling the top-ups for the user and adjusting the account balance through the online charging system.

The online charging system has the properties of both the IT and network elements. The interfaces used by these systems are highly standardized, but the implementation and usage of the rating logic may vary from vendor to vendor.

1.4.3 Rating IMS services

The IP Multimedia Subsystem (IMS) is an optional part of UMTS. Besides the enhanced data rate, the UMTS concept offers novel value added services for its subscribers. The IMS allows the network operator to introduce different IP based multimedia services by offering a wide range framework. The protocols, interfaces and APIs allow the implementation of such services with minimal telecommunication knowledge [73].

The IMS frameworks deploys rating and charging related APIs for the applications and implements the network element related protocols for the offline and online charging systems [4]. The call admission control and real-time control is done by a SIP (Session Initiation Protocol) server, and a special element (the IMS Gateway Function - IMS GWF) is responsible for protocol conversion [5].

1.4.4 Connected systems

Since 1997 all the Hungarian mobile telecommunication companies and mainly all the mobile telecommunication companies around the world are offering prepaid and post-paid subscriptions; the different subscriptions requiring different serving elements underneath. Some elements/modules of the solutions can be shared; some of them are purely dedicated to the charging mode. In this section we will list the main modules that were already presented in the previous sections and the additional elements that are strongly tied to the billing systems. Of course, the real implementations of the functions listed in this section may vary from company to company and may change over time as the systems are evolving or the legacy systems are being replaced. The architecture and systems are also presented in Figure 1.4.

CRM/CM/POS The CRM module is responsible for creating new customers and selling different offers and discounts to them that modify the rating or billing logic used by the rating or billing module. It also helps the CSRs (Customer Sales Representatives) to handle different complaints by allowing them to check the account history and bill images of the customers.

Resources Management The resource management keeps track of the different stocks, resources used in the network and their lifecycle, such as the mobile devices, SIM (Subscriber Identity Module) or Mobile Subscriber ISDN Numbers (MSISDN). The resource management is also responsible for handling the MSISDN ageing and recycling process.

Contract Handling The printed and signed contracts of the subscriptions are uploaded and archived in this module.

Self-care In some cases, the subscribers are allowed to order different services via SMS or via the Internet. The self-care module handles such requests

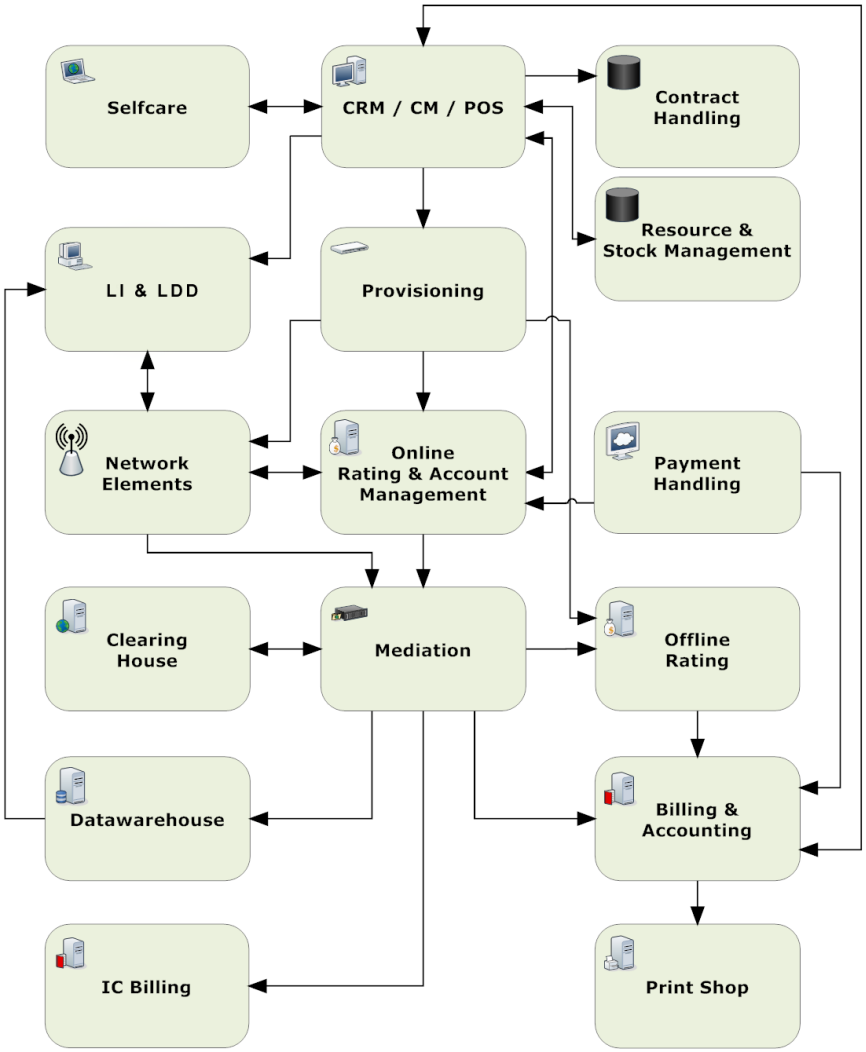

Figure 1.4: Billing system architecture and connected systems

and interacts with the CRM or CM module of the system to initiate the different changes.

Provisioning The provisioning module is responsible for propagating the effect of the services to the network elements. If the customer bought Packet Switched service (e.g.: GPRS), then the serving network elements (GGSN, HLR and IN) has to be notified to allow such services.

Clearing House The roaming records are sent and received to/from the clearing house, which is responsible for converting, sanitizing and dispatching these records among the different network operators.

Network Elements The network elements are responsible for serving the requests, negotiating call admission control with the online charging system and administrate the duration and other parameters of the calls for xDR creating purposes. At the end of the session the corresponding xDR is written and grouped together with other xDRs.

Online Charging It looks up the customer from the database, calculate the price of the call and allow/deny the service request (call admission control). During the service (in case of session based service) it is responsible for monitoring and modifying the account and tearing-down the service if the subscriber's account does not cover the price of the service. An xDR is created after the service has ended and sent to the mediation module for further processing.

Offline Charging It looks up the customer for the xDR in the database and determines the price of the call based on the rating logic, the purchased offers and the accumulated usage information. The rated calls are written to a database as well as the accumulation is updated in the corresponding table.

Mediation The mediation element is responsible for collecting the different files from the network elements. Once the records are collected, it is responsible for dropping the unnecessary records (such as the Mobile Terminated SMS record), formatting the rest if required and dispatching them to the appropriate modules.

Payment Handling The payment handling module is responsible for handling the different payments, channels and calculating the late payment fees. It is often responsible for monitoring the accounts and triggers the different collection processes. In case of online charging it is responsible for handling the top-ups for the user and for adjusting the account balance through the online charging system.

Datawarehouse The datawarehouse (DWH) collects all the service and customer related information to support the business intelligence (BI) and fraud offices.

Lawful Interception and Lawful Data Disclosure With the Lawful Interception module the law enforcement agencies are able to intercept the calls (including SMS, data or any other traffic) of the target users. The lawful data disclosure module supports the company's and the national security departments to collect different information for a specific subscriber (call information, location, name and address, etc.).

Billing Based on the rated calls, the accumulated usage information and billing logic, it calculates the final invoiced amount and books it to the customers' account on a monthly basis. The bill image is sent to the print shop and available from the CRM module.

IC Billing The Interconnect Billing module is responsible for rating and filtering the call detail records used to calculate the interconnect charges between the network operators.

Print Shop Prints and mails the bill to the appropriate customers.

OSS The Operation Support System is responsible for supporting the different operational activities. It continuously tracks the resource consumptions, performance and configurations of the IT systems and creates alerts if the observed indicator is outside a defined range.

1.5 Standards and regulation

The standardization of the UMTS elements and functions are done by 3GPP and 3GPP2. ETSI defines the main functionalities and interfaces of the different network elements and main IT systems. However, the inner functionality and implementation of the IT systems vary from vendor to vendor, which makes the replacement of these modules a struggling and expensive task. From network elements point of view, standards define the main protocols and functionalities, but (for example) the xDRs generated by the elements are proprietary or at least not standardized. The Telecoms & Internet converged Services & Protocols for Advanced Networks (TISPAN) is a standardization body of the ETSI, specializing in fixed networks and Internet convergence [74], and its main task is to apply the different NGN (Next Generation Network) and IMS standards to fixed line networks.

Regulation is done by the local regulation office (NMHH in Hungary for instance) and the European Union (in Europe). The offices/organizations defines the maximal interconnect fees and roaming charges, and sometimes they are requesting new functions to protect the rights of the subscribers and companies. Such regulation was in 2008 the NRTRDE function or in 2010 the bill shock prevention functionality, where the subscribers can limit their roaming data spending. The regulation also extends on the content and structure of the bill.

1.6 Billing system properties

There are several parameters of the charging, rating and billing systems that shall be taken into consideration. In the beginning, most network operator designed and developed its own billing system. As the services and rating logic evolved, these systems become cumbersome and become more a solution, than a product. Nowadays, off-the-shelf systems (products) are favored as they come with support, roadmap and in most case cheaper, than a newly developed home-made solution. The following properties shall be considered:

Maturity How mature the product is? A newly developed (fresh) product may lack a lot of features that will be developed in later versions as the customers of the product suggest them to the vendor. Also, a lot of bugs may be in the system that was not discovered during the testing process.

The more mature (higher version) the product is, the less problem may be revealed on the production environment.

Services What kinds of services/features are offered by the product? How well the business processes of the network provider are compliant with the product.

Customization How easily (rapidly) can the system be altered to the network provider's needs and integrated into the existing IT network? The time-to-market (TTM) is a key factor in the Telco sector.

Support How well the product is supported? Problems, bugs and new features shall be fixed and developed by the vendor (or by the integrator) as fast as possible. Companies with weak support shall block the normal operation and competitiveness of the mobile network operator.

Roadmap How future proof the product is? A well designed and mature product shall have a roadmap that allows the network operators to plan with the features of the new versions and integrate the upgrade into their own roadmap.

Price The price of the product is a very delicate, yet complex parameter. The following prices shall be taken into consideration: price of the product, price of the installation, integration and customization, price of the support, the licensing fees and the operational expenses. The OPEX (Operational Expenditure) and CAPEX (Capital Expenditures) shall be estimated and calculated prior the decision.

1.7 Motivations and research objectives

Although mobile telecommunication is slightly shifting from premium category to a commodity service, a lot of new services and features are being introduced lately. Companies are merging and offering all kinds of services with different business models. The tariff packages and business processes got more and more complex to serve the subscribers and maintain the level and temptation of the services. This implies that a billing and charging system shall be future proof to allow the rapid introduction of such new services.

My preliminary research objective was to examine the effects of novel services and paradigm change on the existing billing architecture and billing systems as follows:

- New services introduced to mass market might have a substantial effect on the performance of the billing systems. My aim was to find appropriate models and calculations to be able to compute the impact of these novel services on performance and on resource requirements.
- A significant amount of network traffic and CPU power are consumed to assure proper charging. Taking the existing standards and implementations into consideration, I have examined the possibilities to reduce the amount of administrative overhead, thus free some of the aforementioned resources.
- Novel services and business models require a highly flexible and complex billing system. A major part of my research was to find a new rating model that fulfils these requirements and assures new functionalities and opportunities to service providers.

The results of these researches can be grouped into three thesis groups as follows:

Thesis 1: I have created a mathematical model to calculate the required processing power of the post-paid rater module and calculated the number of CDRs that has to be processed.

Thesis 2: I have proposed protocol and architecture changes to reduce the amount of charging overhead in pre-paid billing systems

Thesis 3: I have created a new flexible and powerful post-paid rating model.

The next chapters will explain and introduce these results in details.

Chapter 2

Dimensioning processes for rating and charging systems

I have created a mathematical model to calculate the required processing power of the post-paid rater module and calculated the number of CDRs that has to be processed. In Section 2.1 I will present mathematical tools to compute the required processing power of the offline rating system if the CDR arrivals and the size of the event processing window are known. In Section 2.2 I will calculate the number and distribution of partial and final CDRs, while in Section 2.3 I will show how to calculate the required database size for partial CDR storage. Section 2.4 details the required equations and tools used to dimension online charging systems.

2.1 Dimensioning offline rating and charging systems

Offline rating and charging systems are using offline, file based protocols to gather the relevant data from the network elements. This allows some latency during the processing and we can undersize the systems according to peak periods. Even though queuing theory can be applied here with changing incoming probability over time, in most cases the business is not interested in a few minutes difference between processing times. This simplification allows us to observe and calculate the required processing power with a greater scale and using functions that represents the incoming CDR number and the processing power over time. However, the business is interested in the maximum age of the unprocessed CDRs to calculate the fraud possibility and the operational team is interested in the maximum queue size to calculate the required disk space. In the next subsections I will represent the required mathematical formula to compute the minimum processing power if the maximum queue size and latency constraints are given. Finding the proper, not oversized processing power is beneficial, and should lower the cost of IT infrastructure as well as software licensing fees.

The available literature mainly deals with queuing theory while calculating the appropriate sizing for telecommunication and other queue based processing systems [23, 58]. In many case [62] the estimated waiting time is calculated during peak hour, or a constraint is given for the maximum waiting time [26, 51] but the job or record is vanishing from the queue after a certain amount of time,

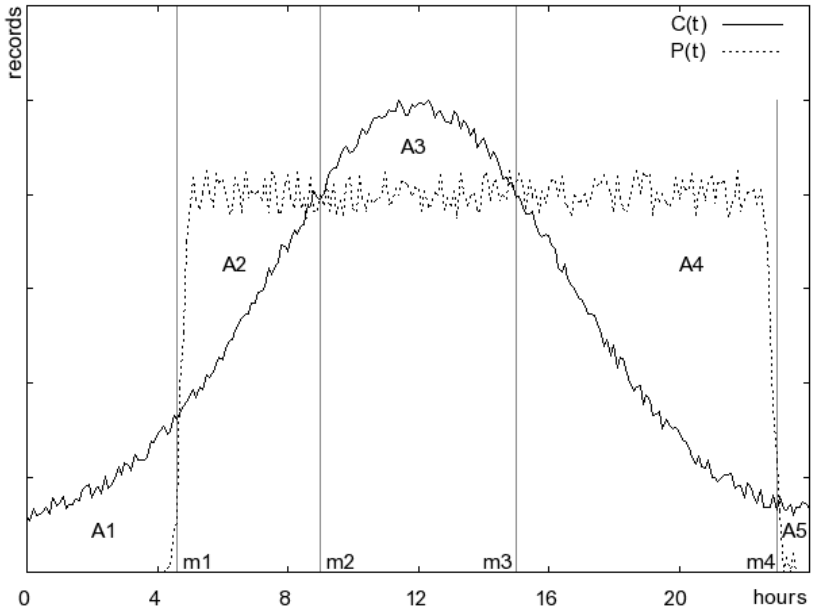

Figure 2.1: General incoming CDR ($C(t)$) and processing power ($P(t)$) functions

thus these models cannot be applied for telecommunication networks and call detail records. Some literatures are dealing with the optimal configuration of the network elements [40] and some are dealing with call center sizing [14] and scheduling [53], which is more sensitive to processing time jitters, and as so, these models shall be used instead in these cases.

In Section 2.1.1 I will clarify the used model and simplifications as well as the possible business requirements. In Section 2.1.2 and 2.1.3 I will detail the queue size and record age constraints respectively. Section 2.1.4 details the effect of national holidays and downtimes, while in Section 2.1.5 I will represent a simple case with simplified functions as an example for the calculations.

2.1.1 Assumptions and requirements

The queue size and the maximum item age in a processing queue cannot be given or calculated in a closed mathematical form in general. Since we are calculating the aforementioned values in a specific system, we can make some assumptions in order to simplify the complexity of the required formulas.

I will use two different functions to represent the main characteristics of the system. I will denote the number of incoming CDRs over time with $C(t)$, and I will represent the processing capacity of the system with $P(t)$. The later one is measured with the number of processable CDRs. Thus, if $P(t) \geq C(t)$ for every t, then the system will process every CDR immediately, which (taking the real-life examples into consideration) is a rude waste of resources and a beautiful example of system oversizing.

The incoming number of CDRs can be represented with a general bell-curve: the number of phone calls, sent SMSs and GPRS activities are low during the night and peaks during the mid-day. The price of the call (or other services) is often different in this two (peak and off-peak) periods. Generally, the event pro-

cessing window (see Section 1.4.1) starts, when the number of incoming CDRs is low but rising, and ends when the number of records is decreasing. The maximum processing power generally does not exceeds the number of incoming CDRs during peak hour, thus the two functions intersects four times as represented in Figure 2.1. The number of unprocessed CDRs in the processing queue increasing as long as the processing power is less than the number of incoming CDRs and decreasing in every other case. I will assume the followings:

AS1 The function representing the incoming CDRs ($C(t)$), and the function representing the processing power of the rating system ($P(t)$) resemble the functions represented in Figure 2.1. At least, the intersections and positions of the functions can be related to the displayed functions.

AS2 Both functions are day-independent. I do not distinguish between weekdays, holidays and working days, and I do not calculate or care the differences between consecutive days.

AS3 The scheduling of the CDRs in the processing queue is FIFO (first in, first out), which complies with the implementation of the available commercial telecommunication billing systems.

Generally, these rating systems shall comply with different business requirements as mentioned in Section 2.1. Some of them are mandatory from engineering point of view; some of them are purely business, financial or security requirements. I will represent a sizing model, where the following three requirements are taken into consideration.

R1 The system shall be capable to process the daily CDRs in one day. Moreover, the system shall have some spare capacity to process additional CDRs (taking Christmas or New Year's Eve into consideration for example).

R2 The maximum number of unprocessed CDRs should not exceed Q (a given IT parameter).

R3 The oldest unprocessed CDR shall not be older than K seconds (a given business requirement) during the normal period. The system shall catch-up (lower the oldest record age below this level) shortly after it is started.

To ease the further computations, please let me distinguish five different areas (A_1, A_2, A_3, A_4 and A_5) and five different moments (m_1, m_2, m_3, m_4 and m_5) as displayed in Figure 2.1 as follows:

A_1 Early morning area. The processing is not yet started, or the processing capacity is less than the number of incoming CDRs. The size of this area is equal to the number CDRs increasing the processing queue during this period.

A_2 Morning area. The rater is up and running and the processing capacity is more than the number of incoming CDRs. The size of this area is equal to the number CDRs vanishing from the queue during this period.

A_3 Peak area. The processing has started, but the number of incoming CDRs exceeds the processing capacity again. The processing queue is increasing, and the increment is equal to the size of this area.

A_4 Afternoon area. The number of incoming CDRs is below the processing capacity. The processing queue is decreasing.

A_5 Night area. The system shut down, but CDRs are still coming in. The processing queue is increased with the size of this area.

m_1 Start time. The moment, when the processing power exceeds the number of CDRs in the morning. This is the end of area A_1 and the start of area A_2.

m_2 Peak start time. The moment, when the number of incoming CDRs exceeds the processing power. This moment is around the start of the peak hour before noon. This is the end of area A_2 and the start of area A_3.

m_3 Off-peak start time. The moment, when the processing power exceeds the number of CDRs in the afternoon. This is the end of area A_3 and the start of area A_4.

m_4 Shutdown time. The number of incoming CDRs exceeds the processing power again. EOD will start shortly. This is the end of area A_4 and the start of area A_5.

m_5 Midnight. This is the end of the day, and the end of area A_5.

We can calculate the above defined areas with the help of $C(t)$, $P(t)$ and the above defined moments as follows:

$$A_1 = \int_0^{m_1} C(t)dt - \int_0^{m_1} P(t)dt \tag{2.1.1}$$

$$A_2 = -\int_{m_1}^{m_2} C(t)dt + \int_{m_1}^{m_2} P(t)dt \tag{2.1.2}$$

$$A_3 = \int_{m_2}^{m_3} C(t)dt - \int_{m_2}^{m_3} P(t)dt \tag{2.1.3}$$

$$A_4 = -\int_{m_3}^{m_4} C(t)dt + \int_{m_3}^{m_4} P(t)dt \tag{2.1.4}$$

$$A_5 = \int_{m_4}^{m_5} C(t)dt - \int_{m_4}^{m_5} P(t)dt. \tag{2.1.5}$$

2.1.2 Queue size

In this subsection I will give mathematical formulas for the first two requirements mentioned in Section 2.1.1. In order to process the proper amount of CDR in one day, we have to determine the processing capability to satisfy the following inequality:

$$\int_0^{m_5} P(t)dt > \int_0^{m_5} C(t)dt. \tag{2.1.6}$$

Using the areas defined in the requirements section, the following statement must comply:

$$D = -A_1 + A_2 - A_3 + A_4 - A_5 > 0. \tag{2.1.7}$$

where D denotes the additional CDR processing power in one day if it is greater than 0. Otherwise the first requirement $(R1)$ is not met.

I have proved, that if $\int_0^{24} P(t)dt > \int_0^{24} C(t)dt$ and the CDR processing queue was empty in the beginning, then the backlog will be zero at m_2 or m_4 every

day. The maximal CDR queue size (backlog) in this case can be calculated as follows: $Q_{max} = max(A_5 + A_1; A_3; A_5 + A_1 - A_2 + A_3; A_3 - A_4 + A_5 + A_1; 0)$.

I will prove, that if $D > 0$, then there is no unprocessed CDR at m_2 or m_4. In order to do this, let me denote the number of unprocessed CDRs at the end of the day with R. Since the queue size is increasing before m_1, during m_2 to m_3 and after m_4, the queue size cannot be negative and due to assumption $AS2$ the value of R on the previous day shall be equal to the current value, we can calculate R as:

$$R = max(0; max(0; R + A_1 - A_2) + A_3 - A_4) + A_5. \tag{2.1.8}$$

If $R + A_1 > A_2$, then the queue is not empty at m_2, since the unprocessed CDRs from the previous day, plus the morning CDRs are not processed till this moment, thus

$$R = max(0; R + A_1 - A_2 + A_3 - A_4) + A_5. \tag{2.1.9}$$

If $R + A_1 - A_2 + A_3 - A_4 > 0$, then $R = R + A_1 - A_2 + A_3 - A_4 + A_5$, but the condition of $A_1 - A_2 + A_3 - A_4 + A_5 < 0$ (see equation 2.1.7) out rules this possibility, leaving us only with the $R + A_1 - A_2 + A_3 - A_4 \leq 0$ option. In such case $R = A_5$, thus the processing queue is empty at m_4.

If $R + A_1 < A_2$, then the queue is empty at m_2, and we have the following equation for the queue size at the end of the day:

$$R = max(0, A_3 - A_4) + A_5. \tag{2.1.10}$$

Thus, the queue size is either $R = A_5$ if $A_3 \leq A_4$ (which makes the processing queue empty at m_4 as well), or $R = A_3 - A_4 + A_5$ otherwise.

It can be easily understood, that the maximum queue size can be calculated as follows:

$$Q_{max} = max(A_5 + A_1; A_3; A_5 + A_1 - A_2 + A_3; A_3 - A_4 + A_5 + A_1). \tag{2.1.11}$$

□

2.1.3 Constraint on record ages

According to the requirements, the system shall catch-up with the CDRs early in the morning. More precisely, the system shall process all the CDRs which are older than K between m_1 and m_2.

I have showed, that if $\int_0^{24} P(t)dt > \int_0^{24} C(t)dt$ and the CDR processing queue is not empty at m_2 (but empty at m_4), then $P(t)$ shall fulfil the following inequation in order not to have older than K records in the queue for every $m_2 \leq x \leq m_4$:

$$A_5 + \int_0^{min(0,x-K)} C(t)dt \leq \int_0^{x} P(t)dt, \tag{2.1.12}$$

if the queue is empty at m_2, then the following inequation shall be fulfilled:

$$\int_{m_2}^{(x-K)} C(t)dt \leq \int_{m_2}^{x} P(t)dt. \tag{2.1.13}$$

for every $m_2 \leq x \leq m_4$.

If the system queue is empty in m_2, then this requirement is straightforward. Otherwise (if the processing queue is empty only at m_4), this requirement can be modelled with the following integral function:

$$A_5 + \int_0^{min(0,x-K)} C(t)dt \leq \int_0^x P(t)dt. \tag{2.1.14}$$

The $P(t)$ processing function shall be capable to fulfil this inequality with the condition of $m_1 \leq x \leq m_2$. Let me denote the result of this equation (the minimal x, which fulfils this inequality) with G (as grace period).

Taking the mid-day ageing requirement into consideration, we have to differentiate two cases. If the queue does not clear out till m_2, then the above equation can be used with the condition of $G \leq x \leq m_4$, otherwise the requirement is fulfilled trivially until $x \leq m_2 + K$, for the rest of the time, the following equation can be used where $m_2 + K \leq x \leq m_4$:

$$\int_{m_2}^{(x-K)} C(t)dt \leq \int_{m_2}^x P(t)dt. \tag{2.1.15}$$

If the queue is empty at m_4, then the requirement is trivially fulfilled after m_4 until $m_4 + K$, moreover it is fulfilled until E (extension period), where E is the solution of the following integral function if $x \geq m_4 + K$

$$\int_{m_4}^{x-K} C(t)dt = \int_{m_4}^x P(t)dt. \tag{2.1.16}$$

□

2.1.4 Holidays and Downtimes

In Section 2.1.1 I made assumptions on the CDR incoming distribution and on the processing power of the billing system and on how these functions are intersecting each other. In some rare cases these assumptions are violated, thus the presented equations are no longer valid. Such cases can be for example:

- Sudden peaks in the call start distributions due to holidays (Christmas, New Year's Eve, etc.)
- Unplanned or non-regular planned downtime
- Regular planned downtime
- Oversized system
- Undersized system

The remedy for such situations depends on the root cause. If the disturbance is caused by a non-regular event (call distribution peaks, non-planned or non-regular downtime) one has to know the estimated effect of such events on the system. In light of the estimated number of additional (unprocessed) records the system should be oversized a bit to process these additional CDRs in a couple of days. The more contingency the system has the faster will it return to the standard operating point (the computation is straightforward). Sadly, in such cases the constraint on the record ages will be most likely violated for a short period of time.

If the system is over- or undersized, it is possible, that the two functions do not intersect. Such an undersized system cannot operate correctly on a long-term as the main requirement (minimal processing power) is violated. If

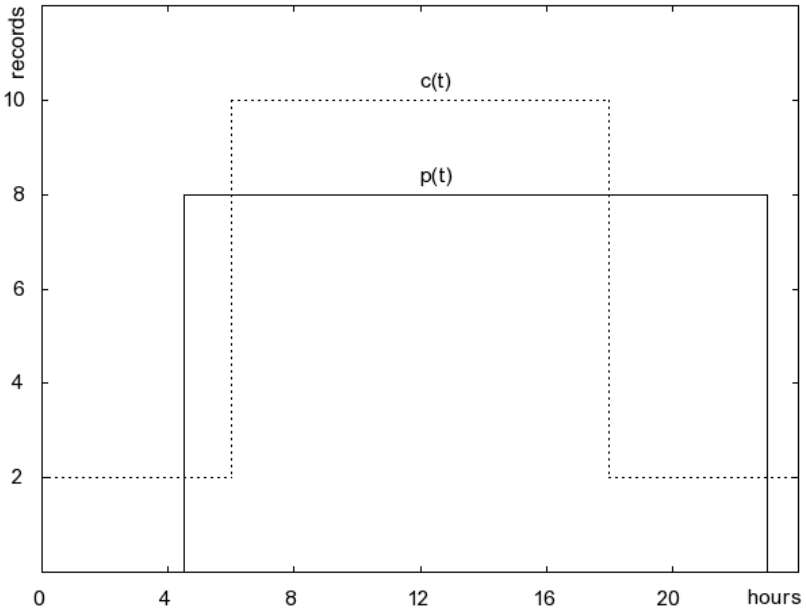

Figure 2.2: Incoming CDRs and processing power functions for simple calculation

the processing power is always greater than the CDR arrival number, then the system is highly oversized and the queue length is always zero.

If the functions are intersecting at two points (e.g. the system is operating 24/7), then the equations presented in the previous sections become simpler: The queue is zero at m_1, the maximum queue length can be calculated with the help of the new A_1, A_2 and A_3 areas. If the functions are intersecting more than four times, then the equations become more complex: A generic equation cannot be given, but the queue size can be calculated with an algorithm or with recursive functions. Without striving for completeness, if the A_i areas are known, and M_0 and A_0 are zero by definition, then the maximal queue size is the maximal value among $M_i = max(M_{i-1} - A_{2i-2}, 0) + A_{2i-1}$.

2.1.5 Demonstrative example

Let me give an example for these calculations. I will simplify both the incoming CDR and the processing functions as represented in Figure 2.2. The processing power will be denoted with P as follows:

$$C(t) = \begin{cases} 2 & \text{if } t < 6 \text{ or } t > 18 \\ 10 & \text{if } 6 \leq t \leq 18 \end{cases} \tag{2.1.17}$$

$$P(t) = \begin{cases} 0 & \text{if } t < 4.5 \text{ or } t > 23 \\ P & \text{if } 4.5 \leq t \leq 23. \end{cases} \tag{2.1.18}$$

Our task is to calculate the value of P so it fulfils the different requirements. The P_{min} minimum processing power can be calculated from the main queuing theory requirement, thus solving the inequality of $\int_0^{24} P(t)dt > \int_0^{24} C(t)dt$ gives us, that

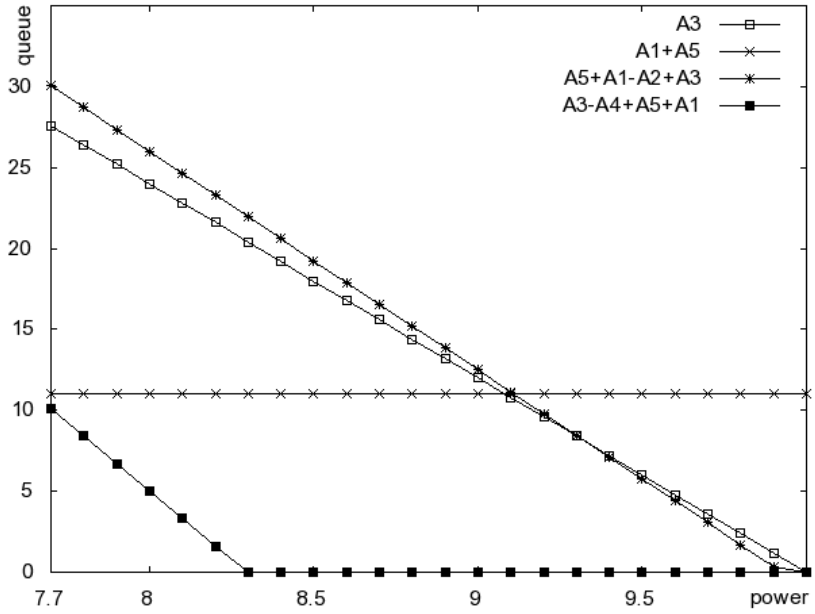

Figure 2.3: Calculated queue sizes for the different areas as a function of processing power

$$P_{min} > \frac{144}{18.5} \approx 7.78. \tag{2.1.19}$$

The maximum queue size can be calculated with (2.1.1)-(2.1.5) and (2.1.11), and represented in Figure 2.3 as a function of P. I have drawn all four values from the *max* function, but only the function with the highest value with a given P shall be used. It can be seen, that we cannot decrease the queue size below 11 but for smaller P values the $Q = A_1 + A_5 - A_2 + A_3$ is dominant. With the minimum required power the system will operate with a queue size around 28.9.

Taking the CDR age into consideration, we have to fulfil two different requirements. The system shall catch-up between 4.5 and 6 o'clock and for the rest of the day, the maximum age in the queue shall not exceed K. It is obvious, that K is a function of P (and vice-versa), and it is represented in Figure 2.4. As it can be seen, the mid-day requirement is stronger, and it requires more power capacity.

Let me do some calculation with the following given requirements: The maximum queue size should not exceed 15, and every CDR should be processed within 1.5 hours. Thus, we have the following equations for queue size:

$$\begin{aligned} Q &= A_1 + A_5 - A_2 + A_3 && (2.1.20)\\ Q &= 9 + 2 - 1.5(P_Q - 2) + 12(10 - P_Q) && (2.1.21)\\ P_Q &= \frac{134 - Q}{13.5} = \frac{134 - 15}{13.5} \approx 8.814, && (2.1.22) \end{aligned}$$

and for the age requirement we get the strongest constraint if $6+K < x \leq 18+K$ when using (2.1.14) since the queue is not empty at m_2:

$$P_K(x - 4.5) = 2 + 12 + 10(x - K - 6) \tag{2.1.23}$$

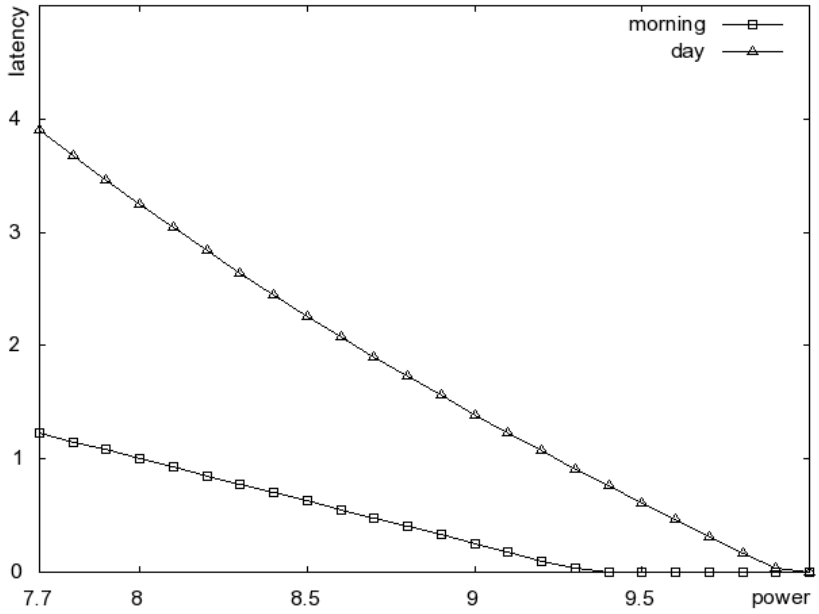

Figure 2.4: Calculated latency of the offline rating system as a function of processing power

$$P_K = 10 + \frac{-10K - 1}{x - 4.5}, \tag{2.1.24}$$

and we need the highest P if $x = 18 + K$, thus:

$$P_K = 10 + \frac{-10K - 1}{13.5 + K} \tag{2.1.25}$$

$$= \frac{134}{13.5 + K} = \frac{134}{15} \approx 8.933. \tag{2.1.26}$$

The required power is the maximum of the above calculated powers, thus

$$P_{total} = max(P_{min}; P_Q; P_K) = P_K \approx 8.933. \tag{2.1.27}$$

2.1.6 Simulation for queue size

A simulation was created to demonstrate the effect of different processing powers. The number of calls on a given time was calculated (assuming that the calls are started with normal distribution over time with $\mu = 12$ and $\sigma = 4.5$), and it was multiplied by a random variable (equally distributed between 0.9 and 1.1). The processing window started at 5:30AM and ended at 11PM, the processing power was 60000 records in the first scenario and 57000 records in the second scenario (the processing power was also multiplied by a random variable between 0.6 and 1.4 in each time interval). The total number of records on a day was one million.

Figure 2.5 and 2.6 represents $C(t)$ and $P(t)$ for the first and second scenario respectively, while in Figures 2.7 and 2.8 shows the backlog size. It can be seen, that in the second scenario the system is unable to process the daily backlog, and the queue size explodes slowly.

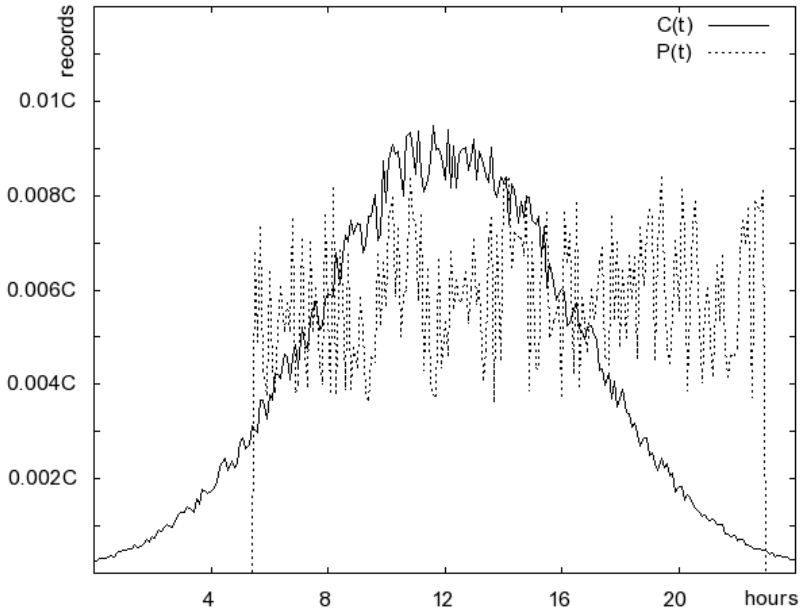

Figure 2.5: Number of records and processing power as a function of time for the first simulation scenario

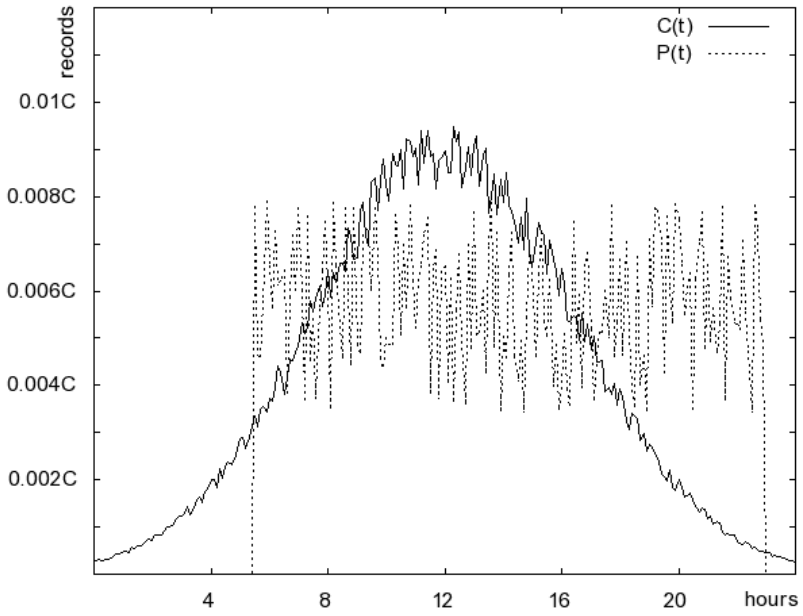

Figure 2.6: Number of records and processing power as a function of time for the second simulation scenario

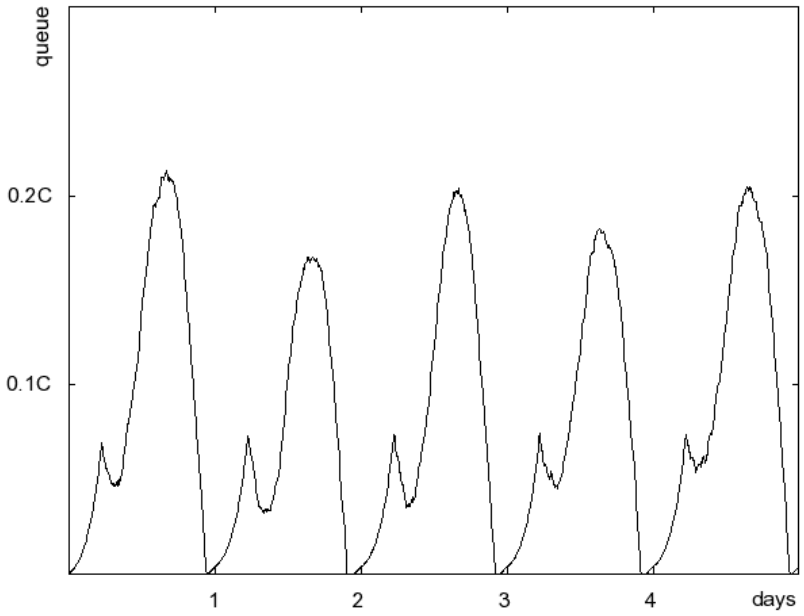

Figure 2.7: Queue size as a function of time for the first simulation scenario

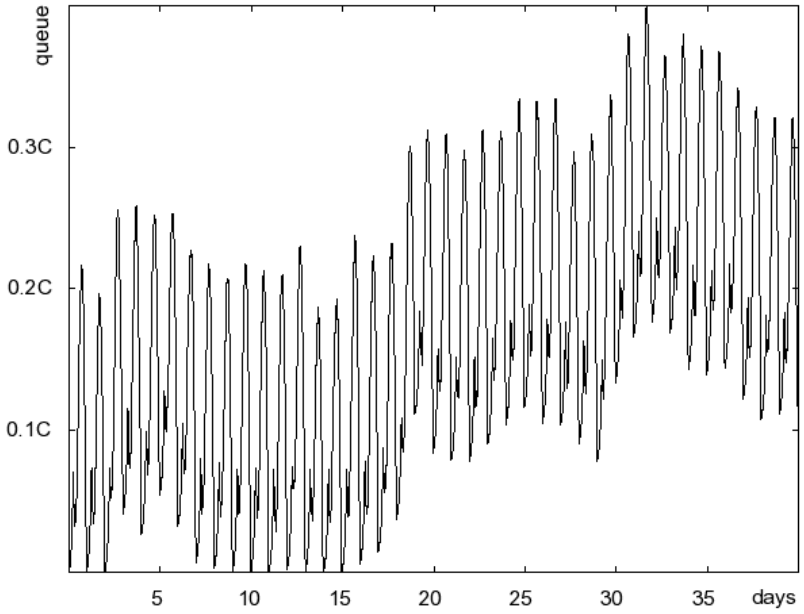

Figure 2.8: Queue size as a function of time for the second simulation scenario

2.2 Number and distribution of partial CDRs

Partial CDRs (FQPCs or RPCs) are generated for long sessions while the call is made, and they are carrying information about the service consumption since the last partial CDR was issued as it was mentioned in Section 1.4.1. In this section I will calculate the CDR distribution for long calls including the partial and the final CDRs. I will assume, that partial CDRs will be generated with exact time intervals, and besides these records only the final CDR is generated - thus no handover, or other behavior results in additional CDR generation. Standards give the possibility to generate partial CDRs at exact time intervals, during handover, when a predefined amount of data transmission reached or if any significant parameter is changed in the service. Considering all of the above parameters would result in a highly complex model (researches are done to model the number of handovers based on the users' movement [16] and several researches are done to model the data transmission for specific content types [80, 22]). Assuming a large number of users, the partial CDR generation can be approximated with fix intervals. I also assume that these records will be sent directly to the billing system one-by-one without further aggregation, buffering or delay (although all the network elements are buffering these records, the delay can be ignored if the corresponding scale is chosen wisely), and that the call start and call length are independent random variables. In the next subsections I will calculate the expected number of partial CDRs, Section 2.2.2 calculates the distribution of the final CDRs if the call start and call length distribution are known, while Section 2.2.3 shows the final distribution of the partial and final CDRs.

2.2.1 Number of partial CDRs

I assume that the call length of the long calls is given with the probability density function $g(t)$. The corresponding cumulative distribution function is denoted with $G(t)$ and the expected value is referred as $E_g(t)$.

The expected number of partial CDRs can be calculated with the following raw approach if the partial CDRs are generated after every K minutes and P_i represents the possibility that the call length is between iK and $(i+1)K$:

$$N = \sum_{i=0}^{\infty} iP_i = \sum_{i=1}^{\infty} iP_i, \tag{2.2.1}$$

and the P_i probability can be calculated as follows:

$$P_i = \int_{iK}^{(i+1)K} g(t)dt = G((i+1)K) - G(iK), \tag{2.2.2}$$

where $g(t)$ represents the probability density function and $G(t)$ the cumulative distribution function of the call length. Please note, that at the end of the session, an additional CDR will be generated as mentioned in Section 2.2, thus (2.2.1) only gives us the number of partial CDRs.

Thesis 1.1: *I have proved, that the average number of partial CDRs sent for a long call is between* $(\frac{E_g(t)}{K} - 1)$ *and* $\frac{E_g(t)}{K}$ *if partial CDRs are generated at every* K *interval and* $E_g(t)$ *represents the expected value of the call length.* [Ary2010ARR]

Later on, I will use the following three equations. Equation (2.2.3) is trivial, once the first few parts of the sum are written in brief format, (2.2.4) is coming from the nature of cumulative distribution functions, while (2.2.5) is coming from the definition of the cumulative distribution function:

$$\sum_{i=1}^{\infty}\sum_{j=i}^{\infty} P_j = \sum_{i=1}^{\infty} iP_i, \tag{2.2.3}$$

$$\sum_{i=j}^{\infty}\Big(G((i+1)K) - G(iK)\Big) = (1 - G(jK)), \tag{2.2.4}$$

$$G((i+1)K) - G(iK) = \int_{iK}^{(i+1)K} g(t)dt. \tag{2.2.5}$$

Because of (2.2.3) and (2.2.4), (2.2.1) can be written as follows:

$$N = \sum_{i=1}^{\infty} iP_i = \sum_{i=1}^{\infty}\sum_{j=i}^{\infty} P_j \tag{2.2.6}$$

$$N = \sum_{i=1}^{\infty} i(G((i+1)K) - G(iK)) \tag{2.2.7}$$

$$= \sum_{i=1}^{\infty}\sum_{j=i}^{\infty}(G((j+1)K) - G(jK)) \tag{2.2.8}$$

$$= \sum_{i=1}^{\infty}(1 - G(iK)). \tag{2.2.9}$$

Let me now calculate the difference between $E_g(t)/K$ and the expected number of partial CDRs:

$$\frac{E_g(t)}{K} - N = \tag{2.2.10}$$

$$\frac{\int_0^{\infty} tg(t)dt}{K} - \sum_{i=0}^{\infty} i(G((i+1)K) - G(iK)) = \tag{2.2.11}$$

$$\sum_{i=0}^{\infty}\int_{iK}^{(i+1)K} \frac{t}{K}g(t)dt - \sum_{i=0}^{\infty} i\int_{iK}^{(i+1)K} g(t)dt = \tag{2.2.12}$$

$$\sum_{i=0}^{\infty}\int_{iK}^{(i+1)K} (\frac{t}{K} - i)g(t)dt. \tag{2.2.13}$$

Since within the boundaries of the integral $iK \leq t \leq (i+1)K$ and

$$0 = \frac{iK}{K} - i \leq \frac{t}{K} - i \leq \frac{(i+1)K}{K} - i = 1, \tag{2.2.14}$$

the following relation is true for the difference:

$$0 \leq \sum_{i=0}^{\infty}\int_{iK}^{(i+1)K} (\frac{t}{K} - i)g(t)dt \leq \sum_{i=0}^{\infty}\int_{iK}^{(i+1)K} g(t)dt = 1, \tag{2.2.15}$$

thus the theorem is proven. □

Let me give an example if the call length distribution is a heavy tailed distribution. I have used the lognormal distribution for this purpose as it is

Table 2.1: Simulated and calculated number of partial CDRs

Scenario	Simulation	Approximation	$E_g(t)/K$
$\mu = 4$; $\sigma = 0.5$; $K = 10$	5.68533731466	5.686761151	6.186780925
$\mu = 3.2$; $\sigma = 0.1$; $K = 7$	3.03301296699	3.03301296699	3.522214288

suggested in the corresponding publication [27]. The lognormal probability density function and cumulative distribution function is

$$g(x;\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\log_n x-\mu)^2}{2\sigma^2}} \tag{2.2.16}$$

$$G(x;\mu,\sigma) = -\frac{1}{2}\mathrm{erf}\Big(\frac{\mu-\log_n x}{\sigma\sqrt{2}}\Big)+\frac{1}{2}, \tag{2.2.17}$$

where *erf()* denotes the error function:

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt. \tag{2.2.18}$$

Using (2.2.9) and (2.2.17) the number of partial CDRs in this special case can be calculated as:

$$N = \sum_{i=1}^{\infty}(1-G(iK)) \tag{2.2.19}$$

$$= \sum_{i=1}^{\infty}\Big[1+\frac{1}{2}\mathrm{erf}\Big(\frac{\mu-\log_n iK}{\sigma\sqrt{2}}\Big)-\frac{1}{2}\Big] \tag{2.2.20}$$

$$= \frac{1}{2}\sum_{i=1}^{\infty}\Big[1+\mathrm{erf}\Big(\frac{\mu-\log_n iK}{\sigma\sqrt{2}}\Big)\Big]. \tag{2.2.21}$$

Using the theorem, the expected value can be easily lower- and upper bounded:

$$\Big(\frac{e^{\mu+\frac{\sigma^2}{2}}}{K}-1\Big) \le N \le \Big(\frac{e^{\mu+\frac{\sigma^2}{2}}}{K}\Big). \tag{2.2.22}$$

I have also created two simulations to confirm the results. I have generated 1 million calls with their length following the lognormal distribution. The used parameters were $\mu = 4$, $\sigma = 0.5$ and $K = 10$ for the first run and $\mu = 3.2$, $\sigma = 0.1$ and $K = 7$ for the second run (see Figure 2.9 for the call length distribution for the first run). I have used the Box-Muller algorithm to generate normal distribution which was later transformed to lognormal distribution. Once the calls were generated, I have calculated the number of partial CDRs for each call ($\lfloor length/K \rfloor$) and compared them to the numeric approximation (summarizing the raw calculation from 1 to 100 – see (2.2.1),(2.2.9), (2.2.17), (2.2.21) and (2.2.23) for the first run) and to the expected value ($e^{\mu+\frac{\sigma^2}{2}}$) divided by K. Table 2.1 summarizes the simulation and numeric results.

$$0.5\sum_{i=1}^{100}\Big[1+erf\Big(\frac{4-\log_n(10i)}{0.5\sqrt{2}}\Big)\Big] = 5.686761151. \tag{2.2.23}$$

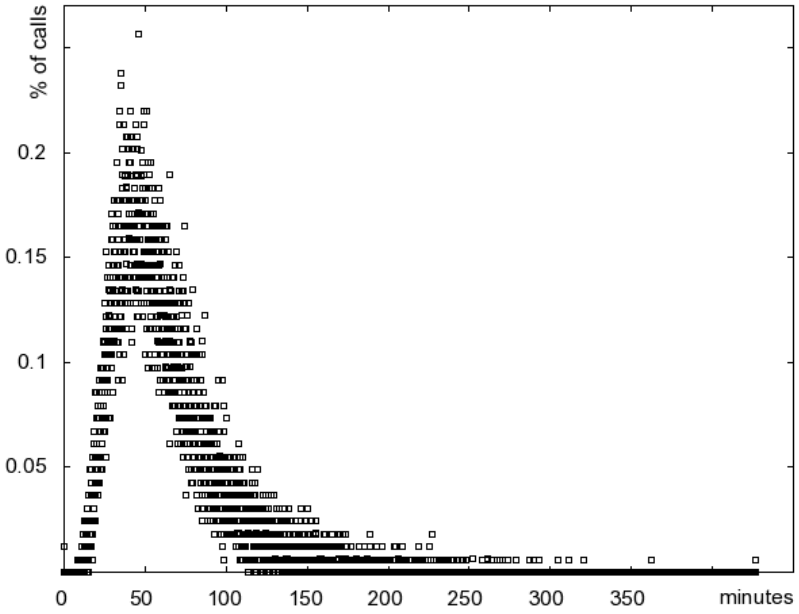

Figure 2.9: Lognormal call length distribution used in the simulation

2.2.2 Final CDR arrival

I assume that the probability density function of calls made on the network is given with $f(\tau)$. If we would like to give a probability density function for the final CDR generation (and arrival to the billing system) we have to include the call length distribution in the equation, since the CDRs arrive to the systems once the calls were finished.

A final CDR is being generated at a given τ time if the call was started t minutes ago, and the call length is exactly t. This probability can be calculated with the multiplication of the two probabilities:

$$f(\tau - t)g(t). \tag{2.2.24}$$

We can calculate the probability that a CDR is generated at τ by summing up (integrating) these probabilities for all valid call length:

$$h_f(\tau) = \int_{0+}^{\infty} f(\tau - t)g(t)dt, \tag{2.2.25}$$

which gives us, that the CDR generation probability density function is the convolution of the call start distribution and the call length distribution since the call length probability distribution has to be zero if t is negative:

$$h_f(t) = (f * g)(t). \tag{2.2.26}$$

This result is straightforward from the assumption that the call start and call length are independent random variables.

Sadly, the call length cannot be modelled with a normal distribution because of the aforementioned restriction. The convolution of a normal and lognormal distribution cannot be given in closed format, thus I have run a simulation to

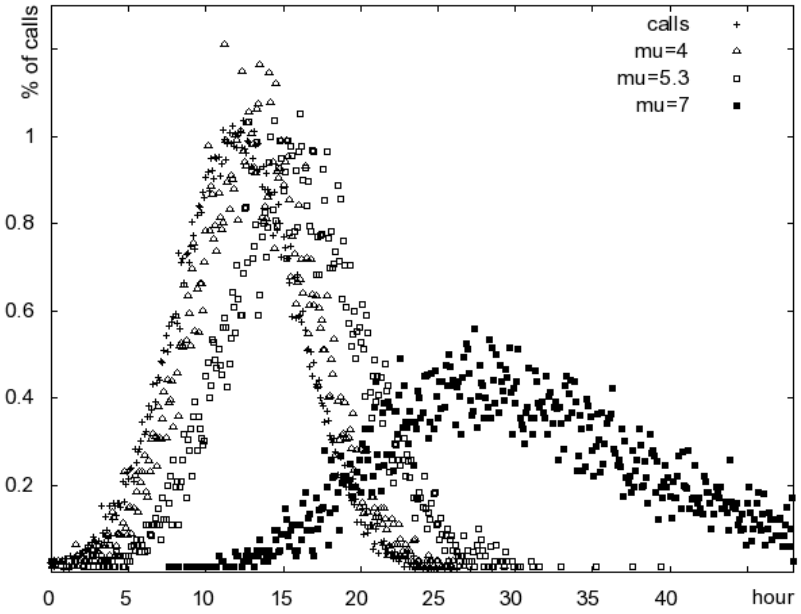

Figure 2.10: Final CDR arrival distribution as a function of time for different call length distributions

observe the final distribution. Nevertheless, the convolution of the normally distributed call start and the lognormally distributed call length can be expressed as follows:

$$h(\tau) = \int_{0+}^{\infty} f(\tau - t)g(t)dt \tag{2.2.27}$$

$$= \frac{1}{2\pi\sigma\rho}\int_{0+}^{\infty}\frac{1}{t}e^{-\frac{(\tau-t-\nu)^2}{2\rho^2}-\frac{(\log_n t-\mu)^2}{2\sigma^2}}dt \tag{2.2.28}$$

In the simulation I have used the Box-Muller algorithm again to generate both the normal, and the lognormal distributions, and I have generated 10000 CDRs with normal distribution on call start and with lognormal distribution on call length. The parameters for the call start distribution was $\mu = 12$ and $\sigma = 4$ (in hours), while for the call length $\sigma = 0.5$ was used and μ varied during the test runs using the values 4, 5.3 and 7 (in minutes). Figure 2.10 represents the call starts and the empirical distribution for CDR arrival for the separate test runs. The expected values for the distributions are listed in Table 2.2.

It can be observed, that for short expected lengths (comparable with the density of the call start), the CDR arrival distribution is not significantly different from the call start distribution, and only slightly shifted to the right. For dimensioning purposes, the original call start can be used, with the mean shifted with the expected length of the calls. However, for larger expected values (e.g.: one day), numerical approximation or simulation is suggested to correctly size the system.

Table 2.2: Expected values for the different parameters

μ	Expected value
4	1.031130154 hour
5.3	3.783522439 hour
7	20.71080278 hour

2.2.3 CDR distribution

If we would like to calculate the probability density function for CDR arrival for long calls, we have to include the final and partial CDRs and weight the probability density function to give 1 when integrated.

Thesis 1.2: *I have proved, that the CDR arrival distribution (including the partial and final CDR) can be calculated as follows:*

$$h(\tau) \quad = \quad \frac{\int_{0+}^{\infty} f(\tau - t)g(t)dt}{1+N} + \frac{\sum_{i=1}^{\infty} f(\tau - iK) \sum_{j=i}^{\infty} P_j}{1+N}, \qquad (2.2.29)$$

where $f(t)$ represents the call start, while $g(t)$ the call length distribution, N is the average number of partial CDRs and P_i is defined in (2.2.2). [Ary2010ARR]

It can be easily understood, that a partial CDR is generated at a given τ time if

- the call was started K minutes ago (at $\tau - K$), and the call length is greater than K.
- the call was started $2K$ minutes ago (at $\tau - 2K$), and the call length is greater than $2K$.
- the call was started $3K$ minutes ago (at $\tau - 3K$), and the call length is greater than $3K$.
- and so on...

The total number of partial CDRs sent is equal to the number of calls (C) multiplied by the expected value of partial CDR per call (N). On the other hand, the number of partial CDRs sent at a given time ($n(t)$) can be calculated with the help of the above mentioned logic:

$$Ch_i(\tau) = n(t) = \sum_{i=1}^{\infty} Cf(t - iK) \sum_{j=i}^{\infty} P_j, \qquad (2.2.30)$$

and the integral of $n(t)$ shall give the total number of partial CDRs, thus:

$$\int_{-\infty}^{\infty} n(t)dt = \int_{-\infty}^{\infty} \sum_{i=1}^{\infty} Cf(t - iK) \sum_{j=i}^{\infty} P_j dt = CN. \qquad (2.2.31)$$

Since $f(t)$ is bounded, $\int_{-\infty}^{\infty} f(t)dt = 1$ and $f(t)$ and $g(t)$ are independent, this equation can be written as

$$\sum_{i=1}^{\infty} C \int_{-\infty}^{\infty} f(t - iK)dt \sum_{j=i}^{\infty} P_j \quad = \quad CN \qquad (2.2.32)$$

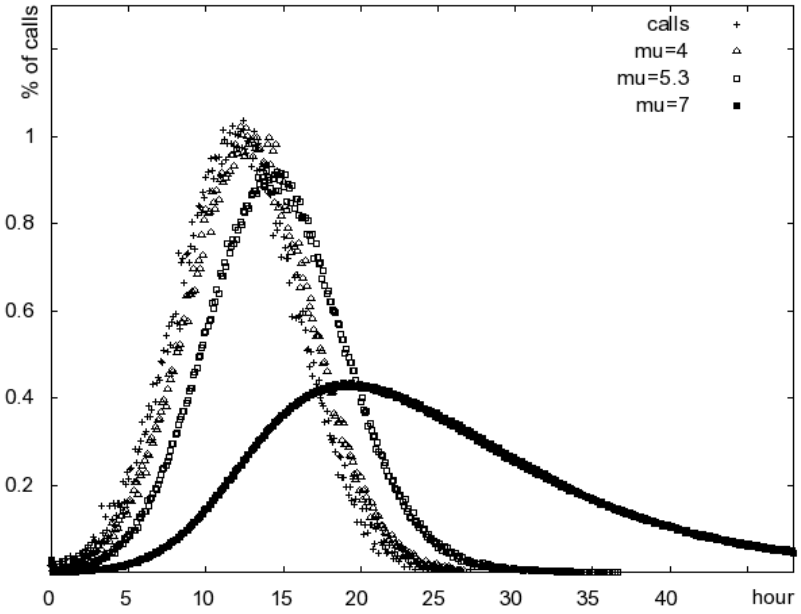

Figure 2.11: Final and partial CDR arrival distribution as a function of time for different call length distributions

$$\sum_{i=1}^{\infty} C \sum_{j=i}^{\infty} P_j = CN \tag{2.2.33}$$

$$C \sum_{i=1}^{\infty} \sum_{j=i}^{\infty} P_j = CN, \tag{2.2.34}$$

which is trivial since (2.2.6).

Combining this with the full CDR distribution, the final CDR probability density function for long calls can be given with

$$h(\tau) = \frac{h_f(\tau) + h_i(\tau)}{1+N} \tag{2.2.35}$$

$$h(\tau) = \frac{\int_{0+}^{\infty} f(\tau - t) g(t) dt}{1+N} \tag{2.2.36}$$

$$+ \frac{\sum_{i=1}^{\infty} f(\tau - iK) \sum_{j=i}^{\infty} P_j}{1+N}, \tag{2.2.37}$$

which complies with (2.2.29). □

Again, I have created a simulation with different parameters to model the CDR arrivals. The approach is identical with the ones in the previous sections. The call start distribution was a normal distribution with $\mu = 12$ and $\sigma = 4$ (in hours), and the call length was lognormal with $\sigma = 0.5$ and μ varied during the test runs (given in minutes). Figure 2.11 displays the results. For short expected lengths, the call start distribution is suggested, where the number of CDRs is $1 + N$. For longer $E_g(t)$, simulation or numerical approximation is suggested.

2.3 Partial CDR database

The partial CDRs are allowing the network operator to gain control over long services. The drawback is that it requires additional database size and processing power to store and correlate these partial CDRs. The required and used database size trivially affects the required processing power and time since the corresponding query execution times are depending on the size of these database tables and thus implicitly on the users behavior such as the number and length of the consumed service, and on some technical parameter such as the partial CDR generation interval.

In this section I will show a method to calculate the required storing capacity if the call start and call length distribution is known and the call length distribution can be approximated with normal, lognormal or Erlang distribution, as it was proved and tested in the corresponding publication [27]. Sadly, in most cases the size of the database cannot be calculated in a closed form, thus I will give a simpler, algorithmic and analytical tool to upper bound the size of the required database. The values calculated with these methods are surely enough to store the partial CDRs if the starting conditions and assumptions (parameters of the distributions) are good and relevant.

In Section 2.3.1 I will give the idea how to calculate the database size in general and introduce some notations and assumptions used later in this section. Section 2.3.2 gives an algorithmic method to calculate the effect of the first few days, while Section 2.3.3 gives us an analytical method to estimate the effect of the rest of the days on the database size. I will make assumptions on the call length distribution only in Section 2.3.3. In Section 2.3.4 I will give a simply usable estimation on the required space in some special cases, while in Section 2.3.5 I will compare the simulation and computed results.

2.3.1 Database size

For the rest of this section, I will use the following notations:

- f denotes the probability density function of the call start distribution.
- g denotes the probability density function of the call length.
- G is the cumulative distribution function of g.
- $H(t)$ denotes $(1 - G(t))$, the probability that a call is longer than t.
- K is the partial CDR generation interval in seconds.
- T_d denotes 24 hour.
- $D(\tau)$ represents the database size at a given τ time.
- C is the amount of calls on one day.

I will also use the following assumptions during the calculations

- Partial CDRs are generated during specific time intervals. However, the corresponding standards allow the operators to set different thresholds (such as transferred data) and partial CDRs are generated during handover, I will calculate with specified time intervals only (see justification in Section 2.2.1).
- $K \leq T_d$. This is a regular best practice among the operators.

- C, f and g is identical every day. I will not calculate with the differences.
- The calls are not aborted by the operator. However, most of the operators are disconnecting the users from the network after a specified interval (e.g.: 24 hour), I will omit this fact. In most cases, the disconnected users are immediately connecting back to the network, thus this case (from partial CDR point of view) is identical to the scenario, if an additional partial CDR is generated at this time, and only decreasing the database size for a short time.
- Partial CDRs are correlated each time they are arriving to the billing system incrementally, thus one session requires only one entry in the partial CDR database. Technically each partial CDR can be stored and correlated at the end of the session; however this approach requires far more disk space and consumes far more processing power. I assume the network operator chooses the first approach to save its resources.

If the subscribers would only be able to start a service on a particular day, the corresponding partial database size could have been calculated as follows:

$$D(\tau) \quad = \quad C \int_K^\tau f(\tau - t) H(t) dt. \tag{2.3.1}$$

The limits of the integral are K and τ, since calls shorter than K does not appear in the database, and there were no calls before this day ($f(t) = 0 \; \forall t < 0$). It can be seen, that the database size is the convolution of the call start probability and the call length probability function. If we would like to take the previous days into consideration, we have to extend the previous equation as follows (if $\tau > K$):

$$\begin{aligned}
D(\tau) = & \; C \int_K^\tau f(\tau - t) H(t) dt \\
& + C \int_0^{T_d} f(T_d - t) H(t + \tau) dt \\
& + C \int_0^{T_d} f(T_d - t) H(t + T_d + \tau) dt \\
& + C \int_0^{T_d} f(T_d - t) H(t + 2T_d + \tau) dt \\
& + ... \qquad (2.3.2) \\
= & \; C \int_K^\tau f(\tau - t) H(t) dt \\
& + C \sum_{i=0}^{\infty} \int_0^{T_d} f(T_d - t) H(t + iT_d + \tau) dt \qquad (2.3.3) \\
= & \; C \int_K^\tau f(\tau - t) H(t) dt \\
& + C \sum_{i=0}^{L-1} \int_0^{T_d} f(T_d - t) H(t + iT_d + \tau) dt \\
& + C \sum_{i=L}^{\infty} \int_0^{T_d} f(T_d - t) H(t + iT_d + \tau) dt. \qquad (2.3.4)
\end{aligned}$$

Let me name two parts of this sum as recent (D_r) and extended (D_e) part as follows:

$$D_r(\tau) = C \int_K^{\tau} f(\tau - t)H(t)dt + C \sum_{i=0}^{L-1} \int_0^{T_d} f(T_d - t)H(t + iT_d + \tau)dt \tag{2.3.5}$$

$$D_e(\tau) = C \sum_{i=L}^{\infty} \int_0^{T_d} f(T_d - t)H(t + iT_d + \tau)dt, \tag{2.3.6}$$

thus

$$D(\tau) = D_r(\tau) + D_e(\tau). \tag{2.3.7}$$

It can be seen, that in order to calculate the database size properly, we have to summarize infinite number of convolutions, moreover in many cases the convolution cannot be given in a closed form. In the above equations L denotes a limit used to separate D_r and D_e, since we will use a different approach to calculate them.

In this section I will estimate the extended ($D_e(\tau)$) and the recent ($D_r(\tau)$) part without calculating the actual convolution. I will calculate the values in a way, that it cannot be less, than the real value. For the estimation of $D_r(\tau)$ I will give a general approach, which does not rely on the density functions, and can be calculated based on empirical results. The infinite sum ($D_e(\tau)$) will be estimated if the call length distribution is normal, lognormal or Erlang.

2.3.2 Calculating the recent part

In this section I will show how to estimate the maximum required database size if we take L days into consideration. The effect of the days after this period can be estimated with the methods detailed in Section 2.3.3. I will not assume anything regarding the distribution of the call length or call start, however, the used values can be calculated easily if the distributions are known or assumed, and can be empirically calculated, if we have the desired information stored in our system.

Thesis 1.3: *I have created an algorithm to calculate (upper bound) the required database size, which takes a predefined number of days into consideration and omitting the rest of the days if partial CDRs are correlated on the fly and we have the required empirical data or we know the distribution class (and parameters) about the call start and call length distributions.* [Ary2012TS]

My approach here is to divide the given days to equal periods (let say hours, or couple of minutes) and I will overestimate the real value with the special case when all the calls in the period are arriving to the system at the very last second (for example, all the calls between $4T$ and $5T$ are arriving to the system at $5T$, if we have divided the days to T length periods). With this approach, the convolution is simplified to a finite summary, and due to the fact that $1 - G(t)$ is monotone decreasing, this calculated value is more or equal to the real value.

The drawback of this method is that we are only able to calculate the database size in finite times (iT). Gladly, it can be easily understood, that the values calculated in these times are more or equal to any of the real values

in the given time period. It is important to note, that with a similar approach (assuming that all the calls in a given interval are started on the very first second) a lower boundary can be given to the database size.

Let us assume, that $T_d|T$, $K \geq T$ and $K|T$. The first condition can be easily fulfilled, since we are choosing T. If the later one is not valid, we can decrease K since by lowering K we are increasing the calculated required database size.

To simplify the equations afterward, let me introduce the following notations:

$$p_d = \frac{T_d}{T} \tag{2.3.8}$$

$$p_K = \frac{K}{T} \tag{2.3.9}$$

$$F_T(i) = \int_{iT}^{(i+1)T} f(t)dt \tag{2.3.10}$$

$$G_T(i) = 1 - G(iT) = H(iT). \tag{2.3.11}$$

I will calculate the values of $D_r(t)$ in given time intervals, more precisely in every T interval ($D_r(iT)$). With these notations, the first part of $D_r(iT)$ (the actual day) is obviously 0, if $i < p_K$, since the calls on this day are not long enough to get in the partial CDR database. In any other case, I will use my approach to batch the calls to the end of the period, and thus the value can be calculated as follows:

$$\sum_{j=p_K}^{i-1} F_T(i-j-1)G_T(j). \tag{2.3.12}$$

The sum of the second day starts from 0 if $i \geq p_K$, or from $p_K - i$ otherwise:

$$\sum_{j=max(0;p_K-i)}^{p_d-1} F_T(p_d-j-1)G_T(kp_d+i+j). \tag{2.3.13}$$

The rest of the days (if any) can be calculated similarly to the second day without the starting limitations. For day k, the equation is as follows:

$$\sum_{j=0}^{p_d-1} F_T(p_d-j-1)G_T(kp_d+i+j). \tag{2.3.14}$$

If the day-limit is L, we have to summarize the days till p_L, thus putting it all together, the recent part can be upper bounded as follows:

$$\begin{aligned} D_r(iT) < & \sum_{j=p_K}^{i-1} F_T(i-j-1)G_T(j)* \\ & + \sum_{j=max(0;p_K-i)}^{p_d-1} F_T(p_d-j-1)G_T(p_d+i+j) \\ & + \sum_{k=1}^{p_L}\sum_{j=0}^{p_d-1} F_T(p_d-j-1)G_T(kp_d+i+j)**. \end{aligned}$$

* if $i \geq P_K$ and 0 otherwise

** if $p_L > 1$ and 0 otherwise

The written equation looks more complicated than it is. I will give an algorithmic approach to demonstrate the above results:

- Calculate all the values of $F_T(i)$.
- Calculate the values of $G_T(i)$ from p_K to Lp_d.
- Multiply the values of $G_T(j)$ with $F_T(k)$ where
  - j goes from p_K to Lp_d in ascending order in every step.
  - k goes periodically in descending order between 0 and p_d in every step, and starts from $i - p_K$.

If $getG(j)$ would return $G_T(j)$, and $getF(j)$ would return $F_T(j)$, then the following pseudo code represents how to calculate the database size at $D_r(iT)$:

```
function getDr(i,C,pk,pd,L)
{
  size=0;
  for (n=pk;n<(L*pd);n++)
  {
    g=getG(n);
    f=getF(((L+1)*pd+i-pk-n) mod pd);
    size+=C*g*f;
  }
  return size;
}
```

Once we have all the calculated values of $D_r(iT)$ for every i, we can get the maximum database size by selecting the maximum value among them. □

I have created a simulation to demonstrate how the maximum value and the position of the peak value changes with the different parameters of the distribution. In the simulation I have generated the $F_T(j)$ data offline, using the limited normal distribution with $\mu_s = 12$ and $\sigma_s = 6$. Let me define the limited normal distribution as follows: the distribution function is 0 if $t < 0$ or $t > 24$, and the rest of the values are upscaled to give $\int f(t)dt = 1$. For call length we have used the lognormal distribution and the $G_T(i)$ data was calculated online, changed as the different parameters of the distribution have changed. Both data was calculated based on the probability density and cumulative distribution functions of the distribution. During the simulations I have used $K = 1$ and $T = 0.1$ and L changed with every simulation run as it is detailed in Section 2.3.3. I have used the above represented algorithm to calculate the maximum value and the element ($D_r(iT)$) which caused the maximum value. I created four different simulation runs. All of them are calculated with σ going from 0.1 to 2 with 0.1 intervals, while μ had a value of 0,1,2 and 3 during the runs, where μ and σ are the parameters of the call length distribution. Figure 2.12 represents the peak position changes, while Figure 2.13 shows us how the peak values are changed.

It was expected, that if the call start distribution is symmetric, and has a peak value (as in case of the limited normal distribution), then the peak position will be between the peak of the call start distribution and the end of the day. If μ of the call length is relatively low, the peak value moves away from the peak value of the call length as σ is increased. The peak size is exponentially

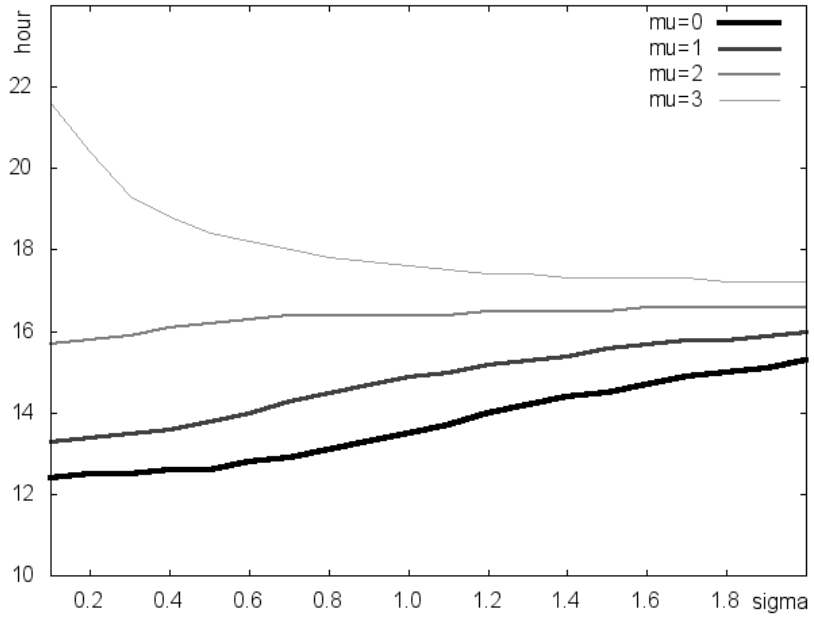

Figure 2.12: The hour of the day when the partial CDR database size peaks as a function of σ for different μ values

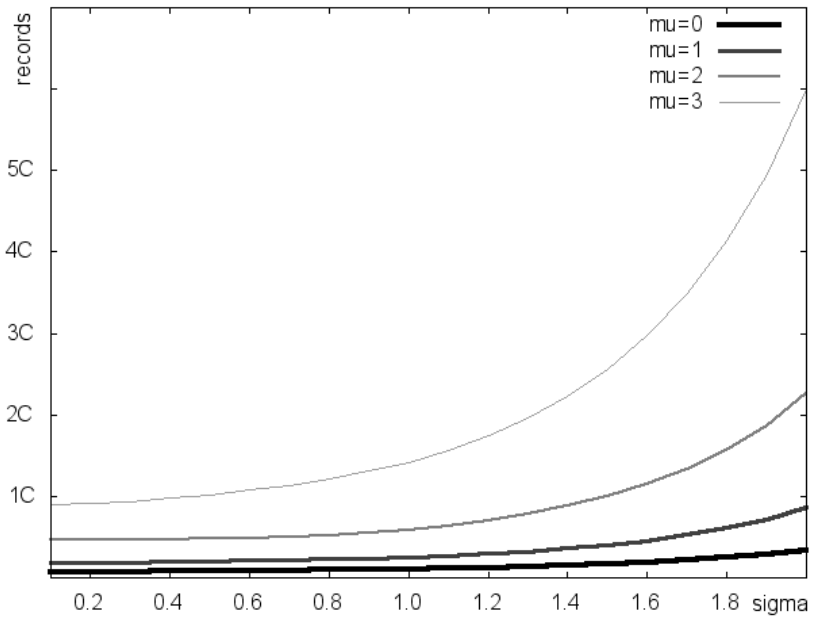

Figure 2.13: Partial CDR database peak values as a function of σ for different μ values

increasing as the average call length is increased. This is due to the fact, that the average call length is an exponential function of μ and σ, thus it is adding exponentially more days to the summary.

2.3.3 Calculating the extended part

Let me define the positive normal distribution as follows:

$$g_m(t) = \begin{cases} 0 & \text{if } t < 0 \\ \frac{g(t)}{1-G(0)} & \text{if } t \geq 0, \end{cases} \tag{2.3.15}$$

where $g(t)$ denotes the density function and $G(t)$ denotes the cumulative distribution function of the normal distribution.

Thesis 1.4: *I have proved, that if the call length either follows the* positive normal, lognormal *or* Erlang *distributions, then the effect of long calls started more than L days ago on the database size can be upper bounded if L is large enough. The L_{min} and the additional S_L database requirement are displayed in Table 2.3 for the mentioned distribution classes.* [Ary2012TS]

I will show how to estimate $D_e(\tau)$, and how to calculate the required (minimal) L limit to separate $D_e(\tau)$ and $D_r(\tau)$. In this section, I will not assume anything regarding the distribution of the call start, since I will overestimate $D_e(\tau)$ with the scenario, when all the calls are arriving to the system at the end of the day as follows:

$$f(t) = \begin{cases} 0 & \text{if } t \neq T_d \\ 1 & \text{if } t = T_d. \end{cases} \tag{2.3.16}$$

The extended part from (2.3.6) can be upper bounded as

$$\begin{aligned} D_e(\tau) &= C \sum_{i=L}^{\infty} \int_0^{T_d} f(T_d - t) H(t + iT_d + \tau) dt \\ &\leq C \sum_{i=L}^{\infty} H(iT_d + \tau) \qquad (2.3.17) \\ &\leq C \sum_{i=L}^{\infty} H(iT_d). \qquad (2.3.18) \end{aligned}$$

During the equation from (2.3.17) to (2.3.18), I have used the fact, that $H(t)$ is monotone decreasing.

In this section, I will estimate $\sum_{i=L}^{\infty} H(iT_d)$ with three different distribution classes: if the call length is normally, lognormally or Erlangly distributed. Please note (as mentioned above), that I will in neither case assume anything about the distribution of the call start.

Normal distribution

I have proved, that if $L \geq \lceil \frac{\mu + 2\sqrt{2}\sigma}{T_d} \rceil$, then

$$\sum_{i=L}^{\infty} H(iT_d) < \frac{0.25 e^{\frac{\sqrt{2}\mu}{\sigma}}}{\sqrt{\pi}} \frac{e^{\frac{-\sqrt{2}LT_d}{\sigma}}}{1 - e^{\frac{-\sqrt{2}T_d}{\sigma}}}, \tag{2.3.19}$$

where μ and σ is the mean and variance, $G_m(t)$ is the cumulative distribution function of the positive normal distribution, T_d is $24h$, and $H(t)$ is $1 - G_m(t)$.

The cumulative distribution function of the normal distribution is

$$G(t) = 0.5 + 0.5erf\Big(\frac{t-\mu}{\sqrt{2}\sigma}\Big), \tag{2.3.20}$$

where $erf()$ denotes the error function. The cumulative distribution function of the positive normal distribution is

$$G_m(t) = \frac{G(t)}{1 - G(0)} \text{ if } t > 0, \tag{2.3.21}$$

thus if the call length distribution follows the positive normal distribution and $G_m(t)$ denotes the cumulative distribution function of this distribution then

$$\begin{aligned} H(t) &= 1 - G_m(t) < 1 - G(t) \\ &= 1 - \Big(0.5 + 0.5erf\Big(\frac{t-\mu}{\sqrt{2}\sigma}\Big)\Big), \end{aligned} \tag{2.3.22}$$

since $G(0) > 0$.

If $u >> 1$, then the error function can be underestimated as follows [75]:

$$erf(u) = 1 - \frac{e^{-u^2}}{\sqrt{\pi}u}\Big(1 - \frac{1}{2u^2} + \frac{3}{4u^4} - \frac{15}{8u^6} + ...\Big) \tag{2.3.23}$$

$$> 1 - \frac{e^{-(u)^2}}{\sqrt{\pi}u}. \tag{2.3.24}$$

If we use $u = \frac{t-\mu}{\sqrt{2}\sigma}$ and if we assume, that

$$t > LT_d = \lceil \frac{\mu + 2\sqrt{2}\sigma}{T_d} \rceil T_d \geq \mu + 2\sqrt{2}\sigma, \tag{2.3.25}$$

then

$$u = \frac{t-\mu}{\sqrt{2}\sigma} > 2 > 1 \tag{2.3.26}$$

and so:

$$erf\Big(\frac{t-\mu}{\sqrt{2}\sigma}\Big) > 1 - \frac{e^{-(\frac{t-\mu}{\sqrt{2}\sigma})^2}}{\sqrt{\pi}\frac{t-\mu}{\sqrt{2}\sigma}} > 1 - \frac{e^{-(\frac{t-\mu}{\sqrt{2}\sigma})^2}}{\sqrt{\pi}2}. \tag{2.3.27}$$

Substituting this into the definition of $H(t)$, we get:

$$H(t) < \frac{0.25}{\sqrt{\pi}} e^{-(\frac{t-\mu}{\sqrt{2}\sigma})^2}. \tag{2.3.28}$$

Equation (2.3.26) further implies, that

$$\Big(\frac{t-\mu}{\sqrt{2}\sigma}\Big)^2 > 2\Big(\frac{t-\mu}{\sqrt{2}\sigma}\Big), \tag{2.3.29}$$

and thus $H(t)$ can be overestimated as follows:

$$H(t) < \frac{0.25}{\sqrt{\pi}} e^{-(\frac{t-\mu}{\sqrt{2}\sigma})^2} \tag{2.3.30}$$

$$< \frac{0.25}{\sqrt{\pi}} e^{-2(\frac{t-\mu}{\sqrt{2}\sigma})} \tag{2.3.31}$$

$$= \frac{0.25 e^{\frac{\sqrt{2}\mu}{\sigma}}}{\sqrt{\pi}} e^{\frac{-\sqrt{2}t}{\sigma}}. \tag{2.3.32}$$

Since we would like to calculate the sum of $H(t)$ at certain points (namely at T_d, $2T_d$, $3T_d$ and so on), let me introduce $t = iT_d$. Please note, that I do not assume at this point, that i is an integer. With this notation, $H(t)$ (or $H(iT_d)$) can be written as follows:

$$H(iT_d) < \frac{0.25 e^{\frac{2\mu}{\sqrt{2}\sigma}}}{\sqrt{\pi}} e^{\frac{-\sqrt{2}iT_d}{\sigma}} \tag{2.3.33}$$

$$= \frac{0.25 e^{\frac{2\mu}{\sqrt{2}\sigma}}}{\sqrt{\pi}} \Big(\frac{1}{e^{\sqrt{2}T_d/\sigma}}\Big)^i. \tag{2.3.34}$$

Let me now write the equation to summarize $H(iT_d)$:

$$\sum_{i=L}^{\infty} H(iT_d) < \frac{0.25 e^{\frac{2\mu}{\sqrt{2}\sigma}}}{\sqrt{\pi}} \sum_{i=L}^{\infty} \Big(\frac{1}{e^{\frac{\sqrt{2}T_d}{\sigma}}}\Big)^i. \tag{2.3.35}$$

From (2.3.35) we can see that it is a geometric progression, where the first element is

$$a_1 = \frac{0.25 e^{\frac{2\mu}{\sqrt{2}\sigma}}}{\sqrt{\pi}} e^{-\frac{\sqrt{2}LT_d}{\sigma}} \tag{2.3.36}$$

and the quotient is

$$q = e^{-\frac{\sqrt{2}T_d}{\sigma}}. \tag{2.3.37}$$

Since $e^{-\frac{\sqrt{2}T_d}{\sigma}} < 1$, the progression is convergent, and its value is

$$\sum_{i=L}^{\infty} H(iT_d) < \frac{a_1}{1-q} \tag{2.3.38}$$

$$= \frac{0.25 e^{\frac{\sqrt{2}\mu}{\sigma}}}{\sqrt{\pi}} \frac{e^{\frac{-\sqrt{2}LT_d}{\sigma}}}{1 - e^{\frac{-\sqrt{2}T_d}{\sigma}}}, \tag{2.3.39}$$

which conforms with my original statement in (2.3.19). □

Lognormal distribution

In case of lognormal distribution, the calculation of $H(iT_d)$ slightly differs if $\sigma < \sqrt{2}$ and if $\sigma \geq 1$.

For the sake of simplicity, let me introduce the following function here:

$$\varsigma(r, L) = \sum_{i=L}^{\infty} \frac{1}{i^r}. \tag{2.3.40}$$

If $r > 1$, then $\varsigma(r, L)$ is finite, since the $L = 1$ case gives us the Riemann-Zeta function.

$$\varsigma(r, 1) = \zeta(r). \tag{2.3.41}$$

From this, it is straightforward, that if $r > 1$ and $L > 1$, then

$$\varsigma(r, L) = \zeta(r) - \sum_{i=1}^{L} \frac{1}{i^r}. \tag{2.3.42}$$

I will use this new function in my calculation.

I have proved, that if $\sigma < \sqrt{2}$ and $L \geq \lceil \frac{e^{\mu+2\sqrt{2}\sigma}}{T_d} \rceil$, then

$$\sum_{i=L}^{\infty} H(iT_d) < \frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}} \varsigma(\frac{\sqrt{2}}{\sigma}, L), \tag{2.3.43}$$

where μ and σ is the mean and variance, $G(t)$ is the cumulative distribution function of the lognormal distribution, T_d is $24h$, $H(t)$ is $1 - G(t)$ and $\varsigma(r, L)$ is defined as $\sum_{i=L}^{\infty} \frac{1}{i^r}$.

The cumulative distribution function of the lognormal distribution is:

$$G(t) = 0.5 + 0.5erf\Big(\frac{\log(t) - \mu}{\sqrt{2}\sigma}\Big), \tag{2.3.44}$$

and similar to the normal distribution (see (2.3.22)-(2.3.31)), if $t > jT_d \geq e^{\mu+2\sqrt{2}\sigma}$, then $\frac{\log(t)-\mu}{\sqrt{2}\sigma} > 2$ and

$$H(t) < \frac{0.25}{\sqrt{\pi}} e^{-2\Big(\frac{\log(t)-\mu}{\sqrt{2}\sigma}\Big)}. \tag{2.3.45}$$

Let me use $t = iT_d$ from now one. Due to the properties of the logarithm function $\log(t)$ can be written as $\log(i) + \log(T_d)$, and so:

$$\begin{aligned} H(iT_d) &< \frac{0.25}{\sqrt{\pi}} e^{-2\Big(\frac{\log(t)-\mu}{\sqrt{2}\sigma}\Big)} && (2.3.46)\\ &= \frac{0.25}{\sqrt{\pi}} e^{-2\Big(\frac{\log(i)+\log(T_d)-\mu}{\sqrt{2}\sigma}\Big)} && (2.3.47)\\ &= \frac{0.25}{\sqrt{\pi}} e^{-\log(i)\frac{\sqrt{2}}{\sigma}} e^{-\sqrt{2}(\frac{\log(T_d)-\mu}{\sigma})} && (2.3.48)\\ &= \frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}} \frac{1}{i^{\frac{\sqrt{2}}{\sigma}}}. && (2.3.49) \end{aligned}$$

The sum of $H(iT_d)$ can be expressed as follows:

$$\begin{aligned} \sum_{i=L}^{\infty} H(iT_d) &< \frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}} \sum_{i=L}^{\infty} \frac{1}{i^{\frac{\sqrt{2}}{\sigma}}} && (2.3.50)\\ &= \frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}} \varsigma(\frac{\sqrt{2}}{\sigma}, L), && (2.3.51) \end{aligned}$$

and so, my statement is proved. □

I have also proved that if $\sigma \geq 1$ and $L \geq \lceil \frac{e^{\mu+2\sqrt{2}\sigma^2}}{T_d} \rceil$, then

$$\sum_{i=L}^{\infty} H(iT_d) < \frac{0.25e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}} \varsigma(\sqrt{2}, L), \tag{2.3.52}$$

where μ and σ is the mean and variance of the lognormal distribution. Please note, that if $1 \leq \sigma < \sqrt{2}$, then both (2.3.52) and (2.3.43) can be used. Please note, that σ is on the square this time in the condition.

Similar to the previous cases, if

$$t > jT_d \geq e^{\mu+2\sqrt{2}\sigma^2}, \tag{2.3.53}$$

then $\frac{\log(t)-\mu}{\sqrt{2}\sigma} > 2\sigma > 2$ and

$$H(t) < \frac{0.25}{\sqrt{\pi}} e^{-2\sigma\left(\frac{\log(t)-\mu}{\sqrt{2}\sigma}\right)}, \tag{2.3.54}$$

and if we use $t = iT_d$ again, then

$$\begin{aligned} H(t) &< \frac{0.25}{\sqrt{\pi}} e^{-2\sigma\left(\frac{\log(t)-\mu}{\sqrt{2}\sigma}\right)} && (2.3.55)\\ &= \frac{0.25}{\sqrt{\pi}} e^{-2\sigma\left(\frac{\log(i)+\log(T_d)-\mu}{\sqrt{2}\sigma}\right)} && (2.3.56)\\ &= \frac{0.25 e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}} \frac{1}{i^{\sqrt{2}}} && (2.3.57) \end{aligned}$$

The sum of $H(iT_d)$ can now be written as

$$\begin{aligned} \sum_{i=L}^{\infty} H(iT_d) &< \frac{0.25 e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}} \sum_{i=L}^{\infty} \frac{1}{i^{\sqrt{2}}} && (2.3.58)\\ &< \frac{0.25 e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}} \varsigma(\sqrt{2}, L), && (2.3.59) \end{aligned}$$

which equals to (2.3.52). □

Erlang distribution

If the call length distribution is Erlang with k and λ parameter, then the cumulative distribution function is

$$G(t) = 1 - \sum_{n=0}^{k-1} e^{-\lambda t} \frac{(\lambda t)^n}{n!}, \tag{2.3.60}$$

thus if we introduce $t = iT_d$, then

$$\begin{aligned} H(t) &= \sum_{n=0}^{k-1} e^{-\lambda t} \frac{(\lambda t)^n}{n!} && (2.3.61)\\ &= \sum_{n=0}^{k-1} e^{-\lambda i T_d} \frac{(\lambda i T_d)^n}{n!}. && (2.3.62) \end{aligned}$$

I have showed, that if $L \geq \lceil \frac{1}{e^{\lambda T_d/k}-1} \rceil$, then

$$\sum_{i=L}^{\infty} H(iT_d) \leq \frac{H(LT_d)}{1 - e^{-\lambda T_d/k}}, \tag{2.3.63}$$

where k and λ are the parameters, $G(t)$ is the cumulative distribution function of the Erlang distribution, T_d is $24h$ and $H(t)$ is $1 - G(t)$.

For the proof, I will use the fact that $\sum_{n=0}^{\infty} \frac{1}{n!} < 3$, and I will distinguish two different cases. If $\lambda T_d \geq 1$ and i is a positive integer, then

$$\begin{aligned} H(iT_d) &= \sum_{n=0}^{k-1} \frac{e^{-\lambda T_d i}(\lambda T_d)^n (i)^n}{n!} && (2.3.64) \\ &< \sum_{n=0}^{k-1} \frac{e^{-\lambda T_d i}(\lambda T_d)^{k-1} (i)^{k-1}}{n!} && (2.3.65) \\ &< e^{-\lambda T_d i}(\lambda T_d)^{k-1}(i)^{k-1} \sum_{n=0}^{k-1} \frac{1}{n!} && (2.3.66) \\ &< 3e^{-\lambda T_d i}(\lambda T_d)^{k-1}(i)^{k-1} = a_i && (2.3.67) \end{aligned}$$

With the help of the division criteria we know, that $\sum H(iT_d)$ convergent, if $\frac{a_{i+1}}{a_i} < 1$. In the current scenario, this is:

$$\begin{aligned} \frac{a_{i+1}}{a_i} &= \frac{3e^{-\lambda T_d (i+1)}(\lambda T_d (i+1))^{k-1}}{3e^{-\lambda T_d i}(\lambda T_d i)^{k-1}} && (2.3.68) \\ &= \frac{3e^{-\lambda T_d} e^{-\lambda T_d i}(\lambda T_d)^{k-1}(i+1)^{k-1}}{3e^{-\lambda T_d i}(\lambda T_d)^{k-1} i^{k-1}} && (2.3.69) \\ &= \frac{e^{-\lambda T_d}(i+1)^{k-1}}{i^{k-1}} = e^{-\lambda T_d}\Big(1 + \frac{1}{i}\Big)^{k-1}, && (2.3.70) \end{aligned}$$

and we get the same result if $\lambda T_d < 1$, since

$$\begin{aligned} H(iT_d) &= \sum_{n=0}^{k-1} \frac{e^{-\lambda T_d i}(\lambda T_d)^n (i)^n}{n!} && (2.3.71) \\ &< \sum_{n=0}^{k-1} \frac{e^{-\lambda T_d i}(i)^{k-1}}{n!} && (2.3.72) \\ &< e^{-\lambda T_d i}(i)^{k-1} \sum_{n=0}^{k-1} \frac{1}{n!} && (2.3.73) \\ &< 3e^{-\lambda T_d i}(i)^{k-1} = a'_i && (2.3.74) \end{aligned}$$

and

$$\begin{aligned} \frac{a'_{i+1}}{a'_i} &= \frac{3e^{-\lambda T_d (i+1)}(i+1)^{k-1}}{3e^{-\lambda T_d i} i^{k-1}} && (2.3.75) \\ &= \frac{3e^{-\lambda T_d} e^{-\lambda T_d i}(i+1)^{k-1}}{3e^{-\lambda T_d i} i^{k-1}} && (2.3.76) \\ &= \frac{e^{-\lambda T_d}(i+1)^{k-1}}{i^{k-1}} = e^{-\lambda T_d}\Big(1 + \frac{1}{i}\Big)^{k-1}. && (2.3.77) \end{aligned}$$

We would like to summarize the values from L, thus according to the condition we can assume, that $i > \lceil \frac{1}{e^{\lambda T_d / k} - 1} \rceil$, and by substituting this to the division criteria, we get:

$$\frac{a_{i+1}}{a_i} = e^{-\lambda T_d}\Big(1 + \frac{1}{i}\Big)^{k-1} \quad (2.3.78)$$

$$< \quad e^{-\lambda T_d}\left(1+e^{\lambda T_d/k}-1\right)^{k-1} \tag{2.3.79}$$

$$= \quad e^{-\lambda T_d}e^{\frac{\lambda T_d(k-1)}{k}} = e^{-\lambda T_d+\lambda T_d-\lambda T_d/k} \tag{2.3.80}$$

$$= \quad e^{-\lambda T_d/k}, \tag{2.3.81}$$

which is obviously smaller than 1.

It means, that we can overestimate $\sum H(iT_d)$ with a geometric progression, where the first element is $H(LT_d)$, and the quotient is $e^{-\lambda T_d/k}$, hence the value of the sum is:

$$\sum_{i=L}^{\infty} H(iT_d) \leq \frac{H(LT_d)}{1-e^{-\lambda T_d/k}}, \tag{2.3.82}$$

as it was stated in (2.3.63). □

Summary

In this section I have showed how to estimate the effect of very long calls on the partial CDR database size. Most of the calls started days ago are ended, but some of them are still active, and the corresponding partial CDR is still stored in the database. If the call lengths are following one of the discussed distributions, then the number of days to be taken into consideration to calculate the required database size is infinite, however this is impossible in the real life.

Calculating or estimating the required database size for partial CDRs is a pretty hard analytical task. However, in these section I have showed, that after L days, the required additional resource can be analytically estimated. I know that the equations represented in these sections are vague, and highly overestimating the number of required records, but they still give some hints to do the dimensioning of the required database.

In this section I have discussed the normal, lognormal and Erlang distributions. The limits and the estimated database sizes are represented in Table 2.3 for every case. Please note, that I discussed two different cases if the call length distribution is lognormal. If the variance of the distribution is $1 \leq \sigma < \sqrt{2}$, then both results are applicable. Higher variances will give unnecessary high values in the first variant, but also it gives lower limits, while the second variant would result in an unpractical high limit in some cases.

The limits calculated in these sections are the minimum limits. The longer we are calculating the database size with the algorithmic approach ($D_r(t)$), the more accurate the sizing will be since L is a parameter in the equations. Our task is to find the proper equilibrium between the analytical and algorithmic calculation based on the parameters of the system and the users' behavior.

2.3.4 Maximum database size in some special cases

I have showed how to calculate the database size if the call start and call length distribution is known. We saw that generally we have to iterate through some finite sums and we can calculate the effect of the infinite summary part analytically. In this section I will show, that the database size can be calculated more easily in some special cases. In this section I will upper bound the database size in such special cases, and with concrete values.

Table 2.3: Database size calculations for old records with given distributions

Distribution	L_{min}	D_e
normal(μ,σ)	$\lceil \frac{\mu + 2\sqrt{2}\sigma}{T_d} \rceil$	$C\frac{0.25e^{\sqrt{2}\mu/\sigma}}{\sqrt{\pi}} \frac{e^{-\sqrt{2}LT_d/\sigma}}{1-e^{-\sqrt{2}T_d/\sigma}}$
lognormal(μ,σ) $\sigma < \sqrt{2}$	$\lceil \frac{e^{\mu+2\sqrt{2}\sigma}}{T_d} \rceil$	$C\frac{0.25e^{\sqrt{2}(\mu-\log(T_d))/\sigma}}{\sqrt{\pi}} \varsigma(\frac{\sqrt{2}}{\sigma}, L)$
lognormal(μ,σ) $\sigma \geq 1$	$\lceil \frac{e^{\mu+2\sqrt{2}\sigma^2}}{T_d} \rceil$	$C\frac{0.25e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}} \varsigma(\sqrt{2}, L)$
Erlang(λ,k)	$\lceil \frac{1}{e^{\lambda T_d/k} - 1} \rceil$	$C\frac{H(LT_d)}{1-e^{-\lambda T_d/k}}$

Thesis 1.5: *I have calculated, that if the call length follows the positive normal distribution and $T_d > K > \mu + 2\sqrt{2}\sigma$, then the database size required to store partial CDRs is less than $0.00732C$, where C is the total number of calls on one day. If the call length follows the lognormal distribution and $T_d > K > e^{\mu+2\sqrt{2}\sigma}$, then the database size required to store partial CDRs is less than $0.01502C$.* [Ary2012TS]

I have proved, that if the call length distribution follows the positive normal distribution, and $T_d > K > \mu + 2\sqrt{2}\sigma$, then the required database size is less or equal to $D(\tau) < 0.00732C$, where C is the total number of calls on one day.

In this case $L = 1$ and $1 - G(t) = H(T) < 0.5 - 0.5erf(2) < 0.00234$, thus

$$\begin{aligned} D(\tau) &= C\int_K^\tau f(\tau - t)H(t)dt && (2.3.83)\\ &+ C\int_0^{T_d} f(T_d - t)H(t + \tau)dt + D_e(\tau) && (2.3.84)\\ &< CH(K) + CH(K) + D_e(\tau) && (2.3.85)\\ D(\tau) &< 2C \cdot 0.00234 + D_e(\tau) && (2.3.86) \end{aligned}$$

We can also estimate the value of $D_e(\tau)$ as follows:

$$D_e(\tau) \leq C\frac{0.25e^{\frac{\sqrt{2}\mu}{\sigma}}}{\sqrt{\pi}} \frac{e^{\frac{-L\sqrt{2}T_d}{\sigma}}}{1-e^{\frac{-\sqrt{2}T_d}{\sigma}}} \quad (2.3.87)$$

$$= C\frac{0.25e^{\frac{\sqrt{2}\mu}{\sigma}}e^{\frac{-\sqrt{2}T_d}{\sigma}}}{\sqrt{\pi}(1-e^{\frac{-\sqrt{2}T_d}{\sigma}})} \quad (2.3.88)$$

$$= C\frac{0.25e^{\frac{\sqrt{2}\mu-\sqrt{2}T_d}{\sigma}}}{\sqrt{\pi}(1-e^{\frac{-\sqrt{2}T_d}{\sigma}})} \quad (2.3.89)$$

$$< C\frac{0.25e^{-4}}{\sqrt{\pi}(1-e^{\frac{-\sqrt{2}T_d}{\sigma}})} \quad (2.3.90)$$

$$< C\frac{0.25e^{-4}}{\sqrt{\pi}(1-e^{\frac{-\sqrt{2}\mu}{\sigma}-4})}, \quad (2.3.91)$$

and since

$$\frac{1}{(1-e^{\frac{-\sqrt{2}\mu}{\sigma}-4})} < \frac{1}{(1-e^{-4})} \text{ and} \quad (2.3.92)$$

$$\frac{0.25e^{-4}}{\sqrt{\pi}}\frac{1}{(1-e^{-4})} \leq 0,00264 \quad (2.3.93)$$

we can overestimate the database size with

$$D(\tau) < 2C \cdot 0.00234 + C * 0.00264 \quad (2.3.94)$$

$$= 0.00732C \quad (2.3.95)$$

which means, that maximum 0.732% of the incoming calls is required. For a daily 10 million calls this is approximately 73200 records. □

I have also proved, that if the call length distribution follows the lognormal distribution and $T > K > e^{\mu+2\sqrt{2}\sigma}$ and $\sigma \leq 1$ or $T > K > e^{\mu+2\sqrt{2}\sigma^2}$ and $\sigma > 1$, then the required database size is less or equal to $D(\tau) < 0.01502C$, where C is the total number of calls on one day.

If $T > K > e^{\mu+2\sqrt{2}\sigma}$ and $\sigma \leq 1$, then $L = 1$ and we get similar conditions, thus:

$$D(\tau) < 2C * 0.00234 + D_e(\tau) \quad (2.3.96)$$

$$< 0.00468C + C\frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}}\varsigma(\frac{\sqrt{2}}{\sigma}, L) \quad (2.3.97)$$

$$\leq 0.00468C + C\frac{0.25e^{\sqrt{2}(\frac{\mu-\log(T_d)}{\sigma})}}{\sqrt{\pi}}\zeta(\sqrt{2}) \quad (2.3.98)$$

$$\leq 0.00468C + C\frac{0.25e^{-4}}{\sqrt{\pi}}\zeta(\sqrt{2}) \quad (2.3.99)$$

$$\leq 0.00468C + C\frac{1}{e^4\sqrt{\pi}} \quad (2.3.100)$$

$$< 0.00468C + 0.01034C = 0.01502C, \quad (2.3.101)$$

since $\zeta(\sqrt{2}) < 4$. If $T > K > e^{\mu+2\sqrt{2}\sigma^2}$ and $\sigma > 1$, then $L = 1$ and we get similar conditions and similar results since:

$$D(\tau) < 2C * 0.00234 + D_e(\tau) \quad (2.3.102)$$

$$< 0.00468C + \frac{0.25e^{\sqrt{2}(\mu-\log(T_d))}}{\sqrt{\pi}}\varsigma(\sqrt{2}, L) \quad (2.3.103)$$

Table 2.4: Partial CDR database size simulation parameters

Simulation	μ	σ	K
$S1$	2	0.7	1
$S2$	3	1.2	1
$S3$	4	0.9	1

$$= 0.00468C + \frac{0.25e^{\sqrt{2}(\mu - \log(T_d))}}{\sqrt{\pi}}\zeta(\sqrt{2}) \tag{2.3.104}$$

$$< 0.00468C + \frac{1}{e^{4\sigma}\sqrt{\pi}} \tag{2.3.105}$$

$$\leq 0.00468C + C\frac{1}{e^{4}\sqrt{\pi}} \tag{2.3.106}$$

$$< 0.00468C + 0.01034C = 0.01502C. \tag{2.3.107}$$

This means, that approximately 1.502% of the incoming calls is required. For a daily 10 million calls this is approximately 150200 records. □

2.3.5 Simulations for database size calculation

Thesis 1.6: *I have created a simulation to calculate the required database size for a given call length distribution and to prove the analytic calculations.* [Ary2012TS]

First of all I have generated 100000 calls where the call start followed the limited normal distribution with $\mu = 12$ and $\sigma = 6$ (detailed in Section 2.3.2 and Section 2.3.3). I have done this, by generating a normally distributed random variable with the Box-Muller algorithm, and I have omitted the call if the call start was below 0 or above 24. Once the proper amount of calls was generated with their call start, I have calculated their length based on the lognormal distribution. The Box-Muller algorithm was used again and the value was transformed to let the distribution follow the lognormal distribution with the given parameters. To satisfy the condition on K (the partial CDR generation threshold) I have thrown away the calls, that were shorter than $K = 1$. For rest of the calls, I have written down the call start time increased by K and the call end time. Once these timing (with the proper markings: START and END) were written down, they were sorted by their timestamps, and so the proper modeling of the partial CDR effects on the database size was created. I have then created a script to calculate the database size for one day by going through the list, and increasing the hypothetical database size with 1 if a timestamp was found with $START$, and decreasing if a timestamp was found with END marking. This database size (in total call percentage) is represented in Figure 2.14 for the three simulation runs. The parameters of the simulation are presented in Table 2.4.

Once the database size was calculated for one day, I have calculated the proper limit with the equations shown in Section 2.3.3 and summarized the database size effect till this limit. A loop was then created which runs from 0 till the limit multiplied by 24 with 0.1 intervals (denoted with i) and if $getDay(i)$ represents the function that returns the database size effect for one day at time i, then the proper database size was calculated as follows:

- if the time is less than 24, then return $getDay(i)$

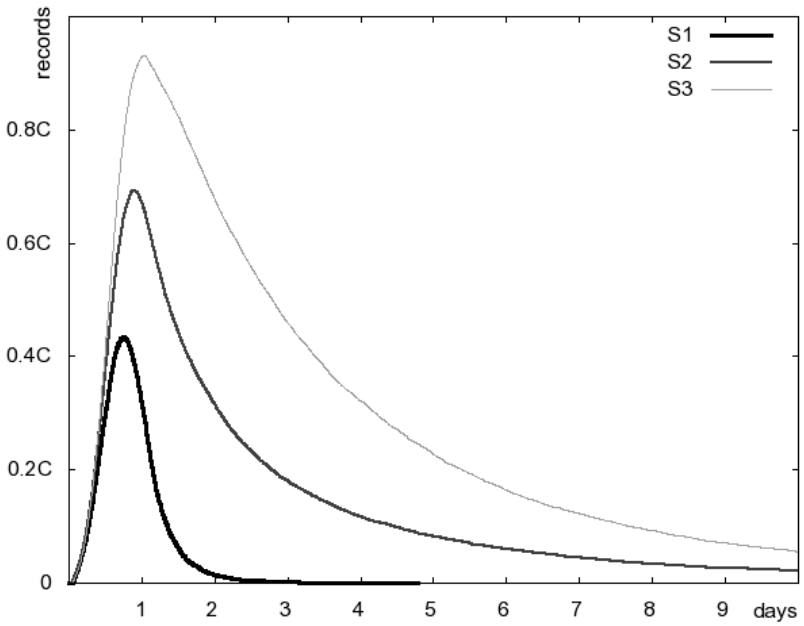

Figure 2.14: Partial CDR database size as a function of time for one day traffic

Table 2.5: Calculating L, D_r and D_e

Simulation	L	D_e	D_r	Total
$S1$	3	$0.004C$	$0.493C$	$0.497C$
$S2(a)$	25	$0.355C$	$1.778C$	$2.133C$
$S2(b)$	50	$0.055C$	$1.802C$	$1.857C$
$S3$	30	$0.134C$	$3.478C$	$3.612C$

- if the time is between 24 and 48 then return $getDay(i-24) + getDay(i)$
- if the time is between 48 and 72 then return $getDay(i-48) + getDay(i-24) + getDay(i)$
- and so on.

This simulation shows us the hypothetical case, when a service is introduced to the customers on day 0 and represents how the database size is increased as we are taking more and more days into consideration (assuming, that the number of calls and the distribution parameters are unchanged). The result of this calculation is presented in Figure 2.15. On the other hand, I have calculated the database size with the methods represented in Section 2.3.2 and 2.3.3 (with simulation interval $T = 1$). The results can be seen in Table 2.5. For the second simulation run ($S2$) I have used both variants, since σ is between 1 and $\sqrt{2}$. To allow easy comparison, I have drawn the calculated L limits and the calculated totals ($D_r(t) + D_e(t)$) in Figure 2.15 with vertical and horizontal dashed lines respectively. It can be seen, that in all cases our estimation was correct, and for $S2$, the (b) variant gave better estimation. □

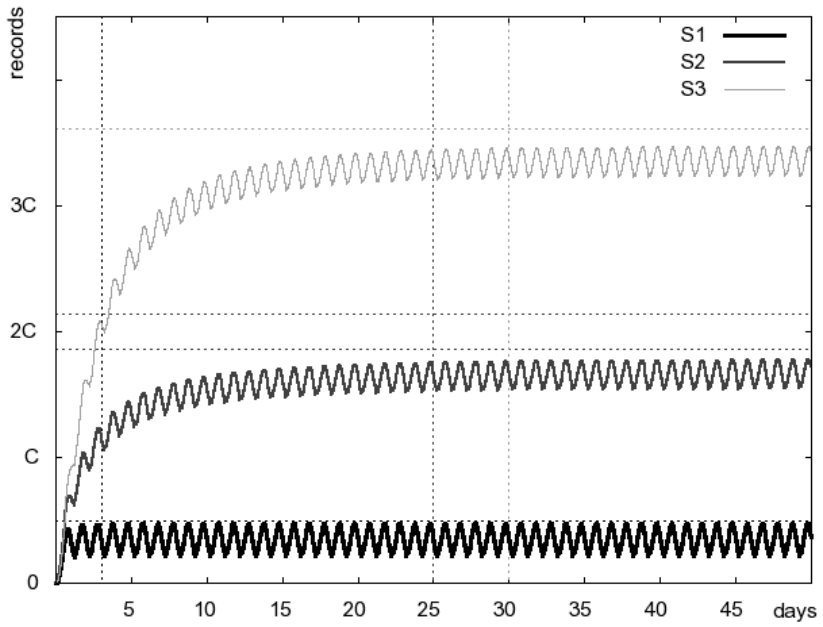

Figure 2.15: Simulation results for the total partial CDR database size as a function of time for consecutive days for different simulation parameters

2.4 Dimensioning online charging systems

During session based services the serving network elements are asking for predefined measurable units from the billing system. Once the billing system ensures that the subscriber's balance covers the requested amount, it allows the network element to serve the requested amount of service to the end-user. If the customer does not end the service before the requested amount is exhausted, the network element asks for additional units from the pre-paid billing system. When the service ends, the serving element reports the total consumed unit which will be re-rated by the billing system and the final price of the service is deducted from the subscribers balance while the possible additionally reserved units are released [5]. The more the reserved units are, the more money will remain on the subscribers' account, since this amount of money will not cover the requested amount of service. On the other hand, small amount of reservations results in high signaling (reservation) traffic and frequent ratings, which puts a high load on the billing system.

Proper dimensioning (sizing) of the online billing systems requires a lot of information such as (but not limited to) the number of subscribers, the number and distribution of the calls and call lengths, the reservation messages in case of session based services and the required number of ratings. In this section I will estimate the average number of reservation messages for a call if the call length distribution is known. In addition I will show how the number of required ratings is depending on the number of unit reservation messages. I will calculate the number of reservation messages in case of session based services where no inverse rating is implemented and the amount of reserved units are fix.

In order to calculate the average number of unit reservation messages for a given call length distribution, we have to observe and understand the protocol of

the session based services. When a call is initiated, the serving network element is reserving the predefined amount of units and once this amount is consumed, it reserves another amount. At the end of the session it reports back the total consumed service. I will include this final reporting in my calculations.

Thesis 1.7: *I have proved, that if the reservation is done with predefined intervals (K), then the average amount of reservation messages for a call is less than* $\frac{E_g(t)}{K} + 2$. *This result was proved with a simulation.* [Ary2011PP]

I have proved, that the number of unit reservation messages (including the final reporting message) is less than the expected value of $g(t)$ divided by K plus 2, where K denotes the reserved units. Moreover, if the expected value is denoted with $E_g(t)$ and all the calls are completed (not even the last message is aborted), then

$$\frac{E_g(t)}{K} + 1 \leq N \leq \frac{E_g(t)}{K} + 2. \tag{2.4.1}$$

Let P_{iK} represent the possibility that the session is longer than iK, then the amount of reservation messages can be calculated as follows:

$$N = \sum_{i=0}^{\infty}(i+2)P_{iK} = \sum_{i=0}^{\infty}2P_{iK} + \sum_{i=0}^{\infty}iP_{iK} = 2 + \sum_{i=0}^{\infty}iP_{iK}. \tag{2.4.2}$$

If $g(t)$ represents the probability density, while $G(T)$ the cumulative distribution function of the call length distribution, then P_{iK} can be calculated as follows:

$$P_{iK} = \int_{iK}^{(i+1)K} g(t)dt = G((i+1)K) - G(iK). \tag{2.4.3}$$

Let me calculate the difference between $E_g(t)/K$ and the expected number of partial CDRs:

$$\frac{E_g(t)}{K} - N = \tag{2.4.4}$$

$$\frac{\int_0^{\infty} tg(t)dt}{K} - \sum_{i=0}^{\infty} i(G((i+1)K) - G(iK)) - 2 = \tag{2.4.5}$$

$$\sum_{i=0}^{\infty}\int_{iK}^{(i+1)K}\frac{t}{K}g(t)dt - \sum_{i=0}^{\infty} i\int_{iK}^{(i+1)K} g(t)dt - 2 = \tag{2.4.6}$$

$$\sum_{i=0}^{\infty}\int_{iK}^{(i+1)K}(\frac{t}{K} - i)g(t)dt - 2. \tag{2.4.7}$$

Since within the boundaries of the integral $iK \leq t \leq (i+1)K$ and

$$0 = \frac{iK}{K} - i \leq \frac{t}{K} - i \leq \frac{(i+1)K}{K} - i = 1, \tag{2.4.8}$$

the following relation is true:

$$0 \leq \sum_{i=0}^{\infty}\int_{iK}^{(i+1)K}(\frac{t}{K} - i)g(t)dt \leq \sum_{i=0}^{\infty}\int_{iK}^{(i+1)K} g(t)dt = 1, \tag{2.4.9}$$

thus

$$-2 \leq \frac{E_g(t)}{K} - N \leq -1 \tag{2.4.10}$$

$$\frac{E_g(t)}{K} + 1 \leq N \leq \frac{E_g(t)}{K} + 2, \tag{2.4.11}$$

which was my theorem in (2.4.1). For the simulation please check Section 3.2.5. □

Please note, that the lower boundaries in (2.4.1) is not valid, if the last call is aborted due to low balance. In this case, the lower boundary shall be downscaled to $(1 - \frac{1}{C})$ where C denotes the average number of calls. The average number of calls can be easily calculated with $\frac{U}{E_g(t)}$ where U denotes the total consumable service, thus (2.4.1) shall be modified as follows:

$$\Big(1 - \frac{E_g(t)}{U}\Big)\Big(\frac{E_g(t)}{K} + 1\Big) \leq N \leq \frac{E_g(t)}{K} + 2. \tag{2.4.12}$$

Sadly, the operators are only aware of the available balance, and to define the total consumable service from the balance requires the inverse rating functionality.

Chapter 3

Reducing charging overhead

In this chapter I will present two different methods to reduce online charging overhead. In Section 3.1 I will present the *mode-switching* model, while Section 3.2 will detail the effect of dynamic and preemptive reservation methods.

3.1 Dynamic mode change

Mobile network operators in general are serving pre-paid and post-paid users with two different rating systems. Pre-paid users are served by an online rating engine (often referred as IN), while post-paid users are rated, charged and billed by an offline billing system. Operating, developing and maintaining two different systems for more or less the same thing might seem to be an unnecessary expenditure. The idea of convergent charging (serving both the pre- and post-paid users with one platform) is in the air since the millennium [50, 45, 47, 49]. It must be noted, that such a single platform is still using online and offline charging for the different subscribers, but the shared rating engine and logic and the shared data model allows the communication service providers to reduce the IT footprint and lower their total cost of ownership. By 2010, mainly every major billing system provider developed its solution for convergent charging and offering a single platform for offline and online rating [69, 25, 13].

The idea behind these solutions is the data model and the rating engine is shared between the offline and online charging system allowing the operator to offer hybrid subscriptions (see Section 1.3) [78, 76, 70] and to ease the prepaid-postpaid migration processes [38]. One of the inventions is automatically switching the user's service consumption from pre-paid to post-paid, if the pre-paid limit is exhausted [77]. As noted above, the main drawback of these inventions is that the pre-paid service consumptions are still rated and charged by an online charging system resulting in high number of ratings and in a huge signaling overhead.

My mode-switching model extends these solutions. The idea behind the mode-switching model is not to glue the charging mode to the type of the payment/service consumption (pre-paid, post-paid), but to dynamically switch between offline and online charging (if online charging is required) considering the user's account as well. Moreover, the overhead of the continuous unit reservation can also be reduced, by granting units only once. The quality of service should also be supervised, in order to charge services properly

In this model, we can assign a service specific limit to every service offered. If the user's account is above this limit, then charging is done in offline mode. If

the subscriber's account drops below this limit, the online charging mechanism is applied (if required), and we grant all the consumable credits to the serving network element. In multi-task systems, it is possible to access more than one service. In such cases, when the account drops below the limit, we shall delegate the credits to multiple network elements. A good solution is to distribute the account among the services with statistical methods, considering the money-consumption and properties of the services and the behavior of the user.

Let me use d_{data} for the amount of useful data transmitted. If the size of a CDR is d_{cdr} and the amount of data that triggers the CDR generation is referred as t_{cdr}, then the $O_{offline}$ charging overhead can be calculated as the quotient of the charging messages and the useful data as:

$$O_{offline} = \frac{\frac{d_{data}}{t_{cdr}} d_{cdr}}{d_{data}} = \frac{d_{cdr}}{t_{cdr}} \tag{3.1.1}$$

in offline mode (if we use the same unit for d_{data}, d_{cdr} and t_{cdr}). Similar to this, the quotient of the charging messages and the useful data in online mode (O_{online}) is:

$$O_{online} = \frac{\frac{d_{data}}{t_{ur}} d_{ur}}{d_{data}} = \frac{d_{ur}}{t_{ur}}, \tag{3.1.2}$$

where d_{ur} is the size of the unit reservation message and t_{ur} is the amount of granted data. Since the unit reservation message should contain more or less the same information as the CDRs, I will assume that d_{ur} is equal to d_{cdr}. In online charging, if the service reserves a large amount of credit from the user's account, access to additional, parallel resources could be denied, because there is no credit left on the account for another resource usage request; even if some service terminates afterwards, and the unused credits are returned to the users. In light of this, a more frequent unit reservation, with a smaller amount of credit should be applied. Because CDRs indicate the used services/data, this problem doesn't occur during offline charging. As follows, t_{ur} is smaller than t_{cdr}, and thus online charging causes bigger network overhead, than offline charging:

$$O_{offline} = \frac{d_{cdr}}{t_{cdr}} < \frac{d_{ur}}{t_{ur}} = O_{online} \tag{3.1.3}$$

□

My idea is to apply offline charging for pre-paid users if their account is far above zero and for post-paid users with credit threshold, if their account is far from the specified limit. If the user's account is close to zero (or to the specified credit limit) online charging should be applied. A crucial question is to determine the threshold limit to switch between offline and online charging.

Thesis 2.1: *I have proposed a mode-switching model, where the system dynamically switches between offline and online charging.* [Ary2005HTS, Ary2005EUN, Ary2005CON]

Let me define a function called unit consumption speed as $C(t)$, having the measure of [unit/sec], which represents the consumed units in one second. Let T_c represent the CDR generation interval and the time needed to rate the call and charge it on the user's account. With these notations and definitions the limit for mode-switch can be calculated. In ideal case it is:

$$L = C(t) \cdot T_c. \tag{3.1.4}$$

If we own more units on our account than L, the charging is done offline with small network overhead; otherwise accounting is done online, with unit reservation. If we require more than one service at a time, the limit can be calculated by the sum of the limits of the services:

$$L = \sum L_i. \tag{3.1.5}$$

The events in a distributed, wide network (signaling, queries) have propagation delay, which is not constant in general. If we want to determine the mode-switching threshold properly, we have to consider the time to switch between modes (T_d), and the variations (jitters) of these values (T_{cj} and T_{dj}) as well:

$$L = C(t) \cdot (T_c + T_{cj} + T_d + T_{dj}). \tag{3.1.6}$$

To ensure accurate charging, we should count with the maximum values of the jitters (T_{ci} and T_{di}). If we want to reduce the values of the mode-switching limits (in order to reduce the network overhead), we shall count with smaller values (with the expected value for example). In this case the possibility of users gaining more service than they paid for can be calculated from the distributions of the jitters.

In case of re-sharing the control messages should be labeled with proper timestamps to be able to charge the services gained during the retransfer and mode switching process. The mode-switching thresholds can be calculated offline for every service offered, and the system can use these pre-calculated values to switch between the charging modes. However, the actual limit can depend on the time of the day and on the user profile (discounts for group of users, statistical behavior for interactive content).

The mode-switching model requires the modification of the corresponding network elements and IT systems. Sadly, the used communication/signaling protocol have to be enhanced as well. The following new functionality is required in the serving network elements:

- During the call admission control, the network element has to query the billing system the required charging method (online or offline) for the given subscriber and service.
- Based on the returned value offline or online charging shall be applied.
- If a mode-switching message is received, the network element has to switch to the desired charging mode.
- During change from offline to online mode, if the required amount of unit cannot be reserved from the subscriber's account, the service shall be terminated and a last resort CDR shall be sent in offline mode (such a scenario can only be caused by bad limit choice).
- A new interface is required that allows the billing system to query if a given service is active for the user or not.

On the other side, the Billing system has to be enhanced with the following items:

- The mode-switching limit has to be calculated for each service for each user. This limit has to be recalculated if a new service or allowance is bought by the subscriber (or an existing service is cancelled).

- The billing system has to calculate for each service the required charging method. This indicator has to be re-calculated if the account of the user changes (during service or top-up).

- If the charging method changes for a service and the service is active, a mode-switching message shall be sent to the appropriate network element.

- A new interface is required that allows the network elements to query the charging method for a given user for a given service.

3.1.1 Mode-switching algorithm

In this section I will present the algorithm/protocol for the mode-switching model.

1. The subscriber requests a service.

2. During CAC, the charging system decides whether online or offline charging shall be used.

3. If offline charging is used, the serving network element starts the service and creates partial CDRs periodically.

4. When a partial CDR arrives to the billing system, the price is reserved from the user's account

5. If the account drops below the limit, then the offline charging system sends a mode-switching command to the network element.

6. When the mode-switching command arrives to the network element, the network element tries to reserve a predefined amount of units from the users' account (the amount of units shall be greater than the amount of services consumed since the last partial CDR).

7. If the account reservation does not succeed, the network element tears down the service and create a final (last-resort) CDR which represents the amount of service used since the last partial CDR was issued.

8. If the account reservation succeeds, the network element deducts the amount of used service units (since the last partial CDR was sent) from the requested units and continues with online charging.

Please note, that if during the CAC process, the account is below the limit then the regular online charging shall be applied (not detailed in the flow above). If the subscriber's account does not cover the requested service, then a final (last-resort) CDR shall be sent. It gives a slight possibility, that the subscribers' account drops below 0, when this CDR is rated and charged to the account. □

3.1.2 Simulation for mode-switching

Thesis 2.2: *I have created a simulation to show the advantage of the mode-switching model.* [Ary2005HTS, Ary2005EUN, Ary2005CON]

During the simulation, I have created several objects as displayed on Figure 3.1. Each of them represented a network or IT element and they were communicating through communication channels, where delays and jitters were applied to the messages. I have simulated the monthly spending for a 100 subscribers, each of them had a starting balance of 10000 credits. The unit price of the call was 30 credit/unit (a rather simple rating logic) and the call length was a random variable with lognormal distribution ($\mu = 2.8$ and $\sigma = 0.7$). The time between the calls was a random variable with uniform distribution (between 1 and 10), the propagation delay was a normally distributed random variable as follows:

- Between the network element and the offline charging system: $\mu = 0.2$ and $\sigma = 0.1$.
- Between the network element and the hot billing charging system (used in case of mode-switching as an offline charging system): $\mu = 0.1$ and $\sigma = 0.1$.
- Between the hot billing charging system and the network element: $\mu = 0.01$ and $\sigma = 0.1$.

The communication channel was assumed as an offline channel between the network element and the hot billing system and an online channel in the other direction. The result of the simulation and the advantage of the mode-switching model depend on the communication delay between the network element and the hot-billing system. The higher the delay, the higher the mode-switching limit will be, and the faster the subscribers will cross this threshold, which eventually will result in a higher number of online ratings. The delay chosen in the simulation is a realistic (even an overestimated) value for such a communication method.

Figure 3.2 shows the average final remaining accounts in case of the different charging modes as a function of partial CDR triggering interval. The mode-switching mode is represented on the figure with the average and minimum accounts. The mode-switching limit was defined as 500 credits in this case. In Figure 3.3 the number of required ratings are represented (which correlates with the number of charging overhead) for the postpaid and mode-switching mode. For the latter one, the different ratings (online or hot billing) are represented separately as well. The pure online ratings are left out of the graph, but had a constant value above 350. Figure 3.4 and 3.5 represents the remaining account and number of ratings as the mode-switching limit changes. The partial CDR interval was set to 10 units in these cases.

Regarding the remaining account, it can be seen that online charging assures that the subscribers' account cannot drop below zero. Due to the nature of offline charging the final account will be significantly below the required threshold, this is why offline charging is not used for pre-paid subscribers. If the mode-switching method is used and the partial CDR generation interval and the mode-switching threshold are correctly chosen, then the account remains non-negative as in the online case. When the threshold was set to a fix value and the interval increased (and the propagation delay remained as is) the subscribers' account started to drop below 0. However, a significantly larger interval still caused less loss to the operator than the offline charging in average. As an example, 17 units is the flawless value in our simulation. For a fix partial CDR generation interval and propagation delay (which is the usual case) a proper threshold can be found. It can be seen, that a mode-switching limit of 350 is a safe choice in our example.

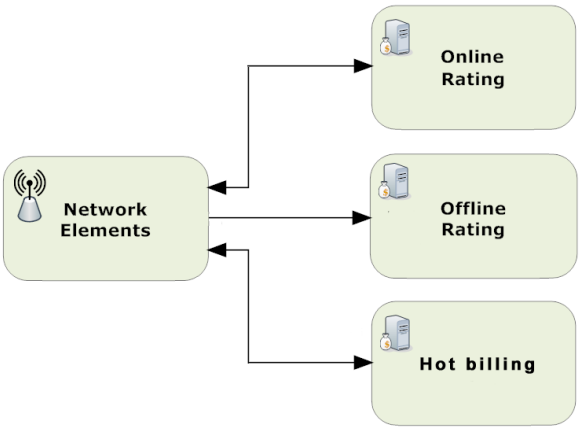

Figure 3.1: Simulation setup for the mode-switching model

Regarding the number of ratings, the mode-switching and offline charging uses less resource than the online charging in our scenarios. Smaller intervals and higher thresholds caused higher resource utilization as expected, but assured proper charging as it was explained before. Please note, that due to the algorithm, the number of ratings with the mode-switching method cannot be higher than the number of ratings in online case if the partial CDR generation interval is less than the units reserved during the online charging. □

3.2 Reducing online charging overhead

Although using the mode-switching limit results in a very low network overhead the corresponding network elements and protocols has to be updated. Considering an existing architecture, this change might be too difficult. In this section I will present some techniques to reduce the charging overhead of online charging without modifying the corresponding protocols. The gain of this model is lower than the gain with the mode-switching model.

The consumed service, the used protocol, the rating logic and the serving network (or IT) element determines whether the amount of unit reserved in each transaction is static or dynamic [44, 20, 43]. If inverse rating exists (see Section 1.4.2) the system and protocol shall be capable to derive and return the available units from the customer's available credits, thus eliminating the remaining balance issue even with high reservation amounts. Another solution would be to define different tiers for reservation (8, 4, 2 and 1 units for example). The billing system would try to reserve the highest defined amount of units, and in case of failure (due to low balance) it will try to reserve the next amount until it succeeds. The frame of this algorithm is known as exponential backoff (or decay) and widely used in the IT industry (e.g.: the Slow-start algorithm in TCP/IP for congestion avoidance [34]).

Such approach would put additional rating load on the system, however assuming that this case would only occur during a low-balance period (until the subscriber refills her balance) and efficient caching mechanism can be introduced, this load can be kept relatively low. This approach will be detailed in Section 3.2.2.

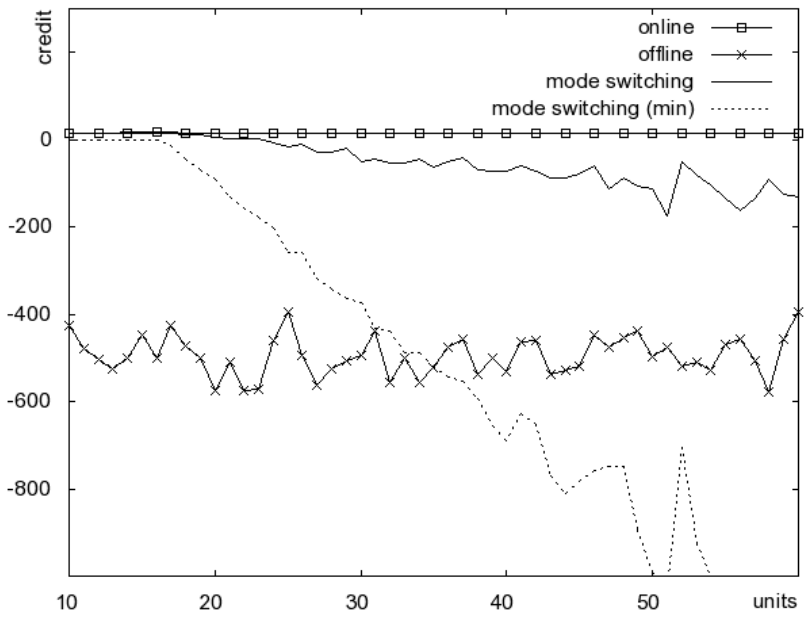

Figure 3.2: Remaining account value as a function of CDR generation interval for the different charging methods

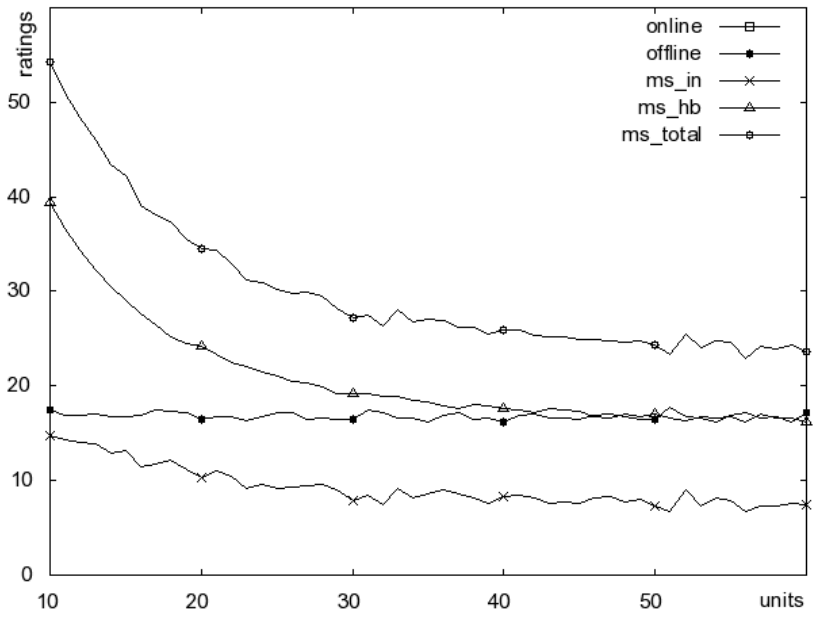

Figure 3.3: Number of ratings as a function of CDR generation interval for the different charging methods

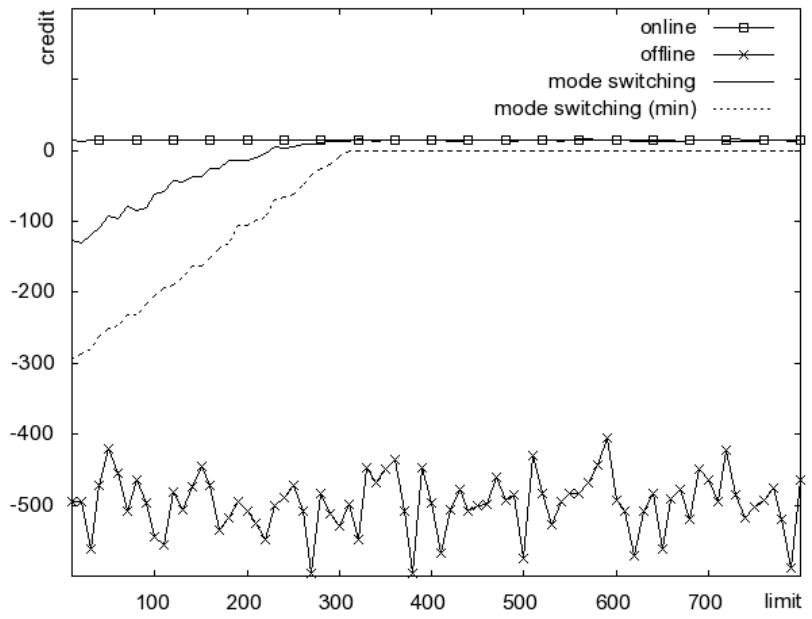

Figure 3.4: Remaining account as a function of mode-switching threshold for the different charging methods

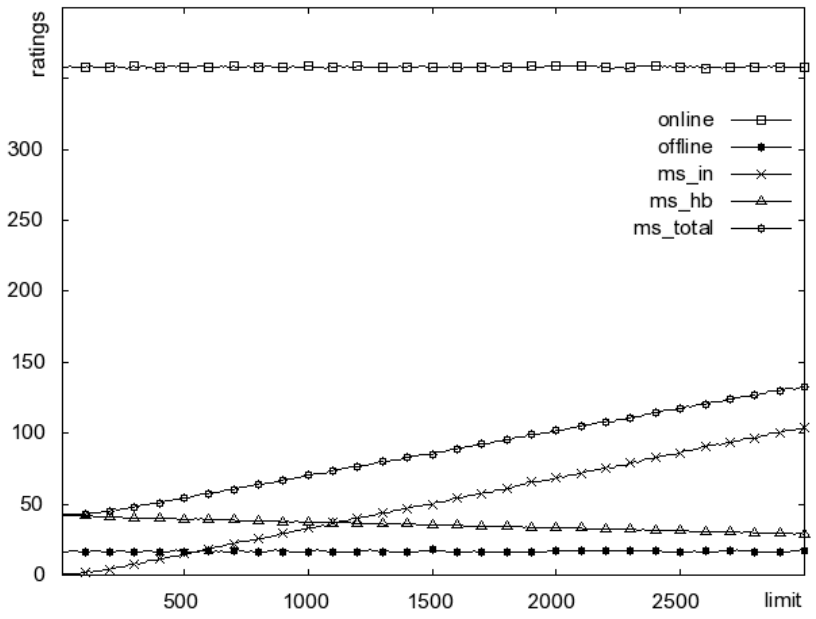

Figure 3.5: Number of ratings as a function of mode-switching threshold for the different charging methods

The consumed service, the used protocol and the serving network (or IT) element determines whether the unit reservation is preemptive or not. Preemptive unit reservation means that even though a predefined amount of unit was reserved for a particular service, the billing system shall ask the serving network (or IT) system to report back the consumed units so far, and ask for another chunk of units to be reserved. This behavior is required if the user shall have more than one active service at a time, and priorities exists among the services. Please note, that the priorities does not have to be hard coded between the services. For example the service started earlier shall have priority over the services started later [39] to assure customer satisfaction.

Imagine that a HSDPA session is initiated during a voice call and the users balance is relatively low. The HSDPA session shall reserve the whole account once the PDP context is activated. In such case, the already active voice call will be terminated when the next reservation occurs, unless some of the reserved account is *taken back* from the HSDPA service.

Without preemptive reservation, the system shall be capable to somehow divide the available units among the active services preventing the HSDPA service (to stick with the previous example) to reserve the whole account. Several techniques are available on the market for such division [39].

Thesis 2.3: *I have created an algorithm, that assures proper charging with dynamic unit reservation, and the average amount of unit reservation messages per calls is less or equal to* $\frac{E_g(t)}{K} + 2 + \frac{LP}{C}$ *where* L *denotes the levels of dynamic unit reservations,* P *denotes the maximum number of service types and* C *denotes the average amount of charged calls or service requests in a topup-period. If preemptive unit reservation is used, then the average amount of unit reservation messages per calls is less or equal to* $\frac{E_g(t)}{K} + 2 + \frac{DLP+LP}{C}$ *where* D *represents the average number of preemptive calls.* [Ary2011PP]

In some special cases there is no need to exchange frequent reservation messages. This particular case can happen when the billing system can measure the consumed service without interacting with the serving network element. Basically this happens, if the measured length of the service can be derived from the length (minute) of the session. This is trivial in case of voice calls, but also possible if the data service assures some QoS and thus rated according to the length (minute) of the session. In some pre-paid billing embodiments, the system calculates the end of the service in advance, when the session is started, and only a tear-down message is sent to the serving network element if the user does not end the service till that moment [1].

3.2.1 Online charging examples

Let me give some example to light the previously detailed characteristics. GPRS sessions are initiated by the users and the SGSN (Serving GPRS Support Node) measures the service. Each time, the SGSN reserves 10KB in the billing system. This scenario is a session based service with static reservation amounts. If the billing system is capable to translate the last few credits to kilobytes, then inverse rating is implemented, and practically there will be no unused credit on the subscribers' amount. If inverse rating does not exists then a few credit will remain on the account.

If a voice session is initiated, and the pre-paid billing system calculates the end of the service and sends a tear-down message to the MSC, then this

Table 3.1: Static unit reservation scenarios

Time	Balance (A)	Event (A)	Balance (B)	Event (B)
0	850 → 770	R1(8)	850 → 830	R1(2)
1				
2			830 → 810	R1(2)
3				
4			810 → 790	R1(2)
5				
6			790 → 770	R1(2)
7	770 → 450	R2(8)	770 → 690	R2(2)
8	450 → 370	R1(8)	690 → 670	R1(2)
9			670 → 590	R2(2)
10			590 → 570	R1(2)
11			570 → 490	R2(2)
12			490 → 470	R1(2)
13			470 → 390	R2(2)
14			390 → 370	R1(2)
15	370 → 50	R2(8)	370 → 290	R2(2)
16		END1	290 → 270	R1(2)
17			270 → 190	R2(2)
18			190 → 170	R1(2)
19			170 → 90	R2(2)
20			90 → 70	R1(2)
21				END2
22			70 → 50	R1(2)
23		END2		
24			50 → 30	R1(2)
25				
26			30 → 10	R1(2)
27				
28				END1
29				

scenario is a session based service with reservation control and inverse rating. The reservation amount is not relevant in such cases.

I have created a few scenarios to demonstrate the differences between the approaches. In each scenario I have assumed, that the user has 850 credits on her account and starts service 1 at $t = 0$ and service 2 at $t = 7$. The price of the services are 10 and 40 for each time interval respectively. $R_i(t)$ means, that the corresponding serving network element is reserving t amount of unit from the subscribers balance for service i, while END_i means, that the serving network element is aborting service i because the subscriber's balance does not cover further reservation. I assumed that there is no inverse rating despite of the fairly simple rating logic. The tables representing the scenarios are showing the time, the balance change and the event that occurs at that given time.

With these notations and assumptions I have modelled the static reservations in Table 3.1. In the A variant, the unit reservation was 8 units for both service, while I have applied a static, 2 unit reservations in variant B. It can be seen, that the A variant used only a few reservation message, but left a fairly huge amount of unused credit on the subscribers account.

Table 3.2: Dynamic unit reservation scenarios

Time	Balance (C)	Event (C)	Balance (D)	Event (D)
0	850 → 770	R1(8)	850 → 770	R1(8)
1				
2				
3				
4				
5				
6				
7	770 → 450	R2(8)	770 → 450	R2(8)
8	450 → 370	R1(8)	450 → 370	R1(8)
9				
10				
11				
12				
13				
14				
15	370 → 50	R2(8)	370 → 50	R2(8)
16	50 → 10	R1(4)	50 → 10	R1(4)
17				
18				
19				
20	10 → 0	R1(1)	10 → 0	R1(1)
21		END1	0 → 80	REALLOCATE
			80 → 0	R1(8), END2
22				
23		END2		
24				
25				
26				
27				
28				
29				END1
30				

In Table 3.2 I have calculated the required messages if dynamic unit reservation apply without (C) and with preemptive allocation (D). In both cases the amount of reservable units were 8, 4, 2 and 1 for both service. Each time the reservation with a higher amount does not succeeds, the system tries to reserve a smaller amount and tears down the service if not even the reservation of the smallest amount succeeds. During the preemptive reservation I have assumed, that service 1 has higher priority (since it was started earlier), and when neither unit reservation succeeds at $t = 21$ it requests the second service to release the unused credits. Since there were two unused credits at that moment for the second service, its price (80) was released, and allowed service 1 to continue. Sadly, the first service consumed the whole amount, thus service 2 was aborted.

I have summarized the amount of reservation messages, the total served units for both service as well as the unused credits in each scenarios in Table 3.3.

Table 3.3: Unit reservation summary

	A	B	C	D
final remaining balance	50	10	0	0
reservation messages	4	21	6	7+
service 1 length	16	28	21	29
service 2 length	16	14	16	14

3.2.2 Additional messages in case of dynamic reservation

From Table 3.3 and (2.4.1) it can be seen, that longer reservation units results in fewer reservation messages but leaves more unused credits on the subscribers account. Smaller credits are eliminating this problem but require more signaling traffic. Dynamic unit reservation is capable to solve both issues but requires a more complex mechanism and protocol. In light of reservation messages the upper boundary of (2.4.1) shall be extended, since the last few calls are issuing more signaling traffic.

If the units of the dynamic reservation are wisely chosen, the number of additional messages per call shall not exceed the number of available reservation steps. Moreover, if additional caching mechanism is introduced, then the total amount of additional messages shall not exceed this limit. In order to achieve this, we have to:

- Choose the step in a way, that each step shall be the half of their preceding step. To give an example for voice calls, the available reservation steps shall be: 8, 4, 2 and 1 minutes. The last step shall be the minimum consumable service.
- Introduce a caching mechanism, so the system will remember the lowest step used (for each service type). This cache shall be reset, when the subscriber top-ups her balance.

According to the previous points, the algorithm will be as follows:

1. Try to reserve the maximum possible amount of units for the call. If it succeeds, then proceed with the next step. If not, then go to step 3.
2. Assure the service and go to step 1 if the call is not ended before the reserved units are consumed.
3. Try to reserve a smaller amount and mark that this is the highest reservable amount for this service and proceed with the next step.
4. If it succeeds, assure the service and go to step 3 if the call is not ended before the reserved units are consumed. If it does not succeed, go to step 3 immediately.

It can be easily understand, that with these innovations, the upper boundary of the number of reservation messages is

$$N \leq \frac{E_g(t)}{K} + 2 + \frac{LP}{C}, \tag{3.2.1}$$

where L denotes the number of reservation steps, P denotes the maximum number of service types and C represents the average number of charged calls or service requests in a topup-period. For a normal call maximum $\frac{E_g(t)}{K} + 2$

Table 3.4: Parameters for online charging simulations

Parameter	Values
balance	100, 1000, 2000, 4000
μ	0.5, 1, 1.5, 2, 2.5 ,3
σ	0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6
reserved unit	0.5, 2, dynamic

reservations are used. If the subscriber's account is low, some (but maximum L) re-reservation may occur for a particular call type, hence the additional LP messages.

3.2.3 Additional messages in case of preemptive reservation

During preemptive reservation (a service with higher priority requests the redistribution of the available balance) an additional unit reservation message is expected from the interrupted call. If we also use dynamic reservation units and we denote the expected number of preemptive reservations with D, then the total reservation messages can be overestimated with

$$N \leq \frac{E_g(t)}{K} + 2 + \frac{DLP + LP}{C}, \tag{3.2.2}$$

since each preemptive reservation can trigger the redistribution and reservation unit calculation. □

3.2.4 Number of ratings

Due to the implementation and behavior of the protocol, it can be understood, that the maximum number of ratings does not exceed the maximum number of messages. Please note that this does not mean that in an actual scenario the number of ratings cannot exceed the number of messages. The dynamic reservation is a perfect example, since interim steps (4 and 2 in the example) has to be rated to check whether they can be applied or not, but is has to be reported back to the serving network element only if the subscriber's balance covers that step. Thus the maximum number of ratings can be calculated with (2.4.1), (3.2.1) or (3.2.2) for normal, dynamic and preemptive reservations respectively.

3.2.5 Simulations for online charging

I have created a simulation to demonstrate my calculations. I have implemented a stripped down version of the unit reservation protocol mentioned in the previous sections and calculated the average number of unit reservation messages and number of ratings for 10000 subscribers. The calls were following the lognormal distribution, while the price of the call was set to 20 *credit/unit*. During the simulations I have varied the available balance, the parameters (μ, σ) of the distribution and the unit reservation amount as displayed in Table 3.4. The dynamic reservation was used with 8, 4, 2, 1 and 0.5 units.

In Figure 3.6 I have shown the average number of unit reservation messages when the reserved unit was 0.5 and the balance, median (μ) and variation (σ) have changed during the simulation runs. The results of the simulations is represented with small black squares, the maximum value (from (2.4.1)) is

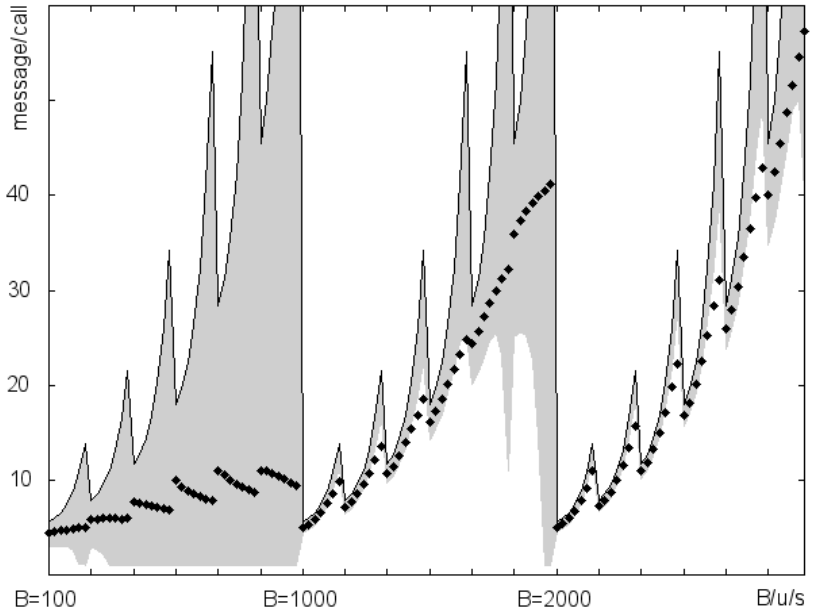

Figure 3.6: Simulated number of unit reservation messages and their calculated ranges for different simulation parameters

Table 3.5: Selected simulation results

Parameter	S1	S2	S3	S4	S5	S6
Balance	1000	1000	1000	4000	4000	4000
Median (μ)	1	1	1	2.5	2.5	2.5
Reserved unit	0.5	2	dynamic	0.5	2	dynamic

represented with a solid line while the minimum value (as calculated in (2.4.12)) is displayed as the lower boundary of the grey area. On the x axis the different parameters of the simulation was represented. The balance is explicitly stated, the minor tics representing the median change ($\mu = 0.5, 1, 1.5, 2, 2.5, 3$), while the variance change ($\sigma = 0.4, 0.6, 0.8, 1, 1.2, 1.4, 1.6$) is displayed between the tick marks. We can observe, that the simulation results were always below the estimated maximum, however, in some cases (when σ and the expected length of the call was high) the results were below the expected minimum. This is due to the fact, that in these cases the total number of calls was less than the calculated value because of the small number of calls and the high variance.

In Figure 3.7 I have plotted six simulation results as displayed in Table 3.5 to let us compare the effect of the used unit reservation amount. The x axis represents the variance (σ) change, while the y axis shows the average number of reservation messages. The simulation results confirm my speculation in Section 3.2.2.

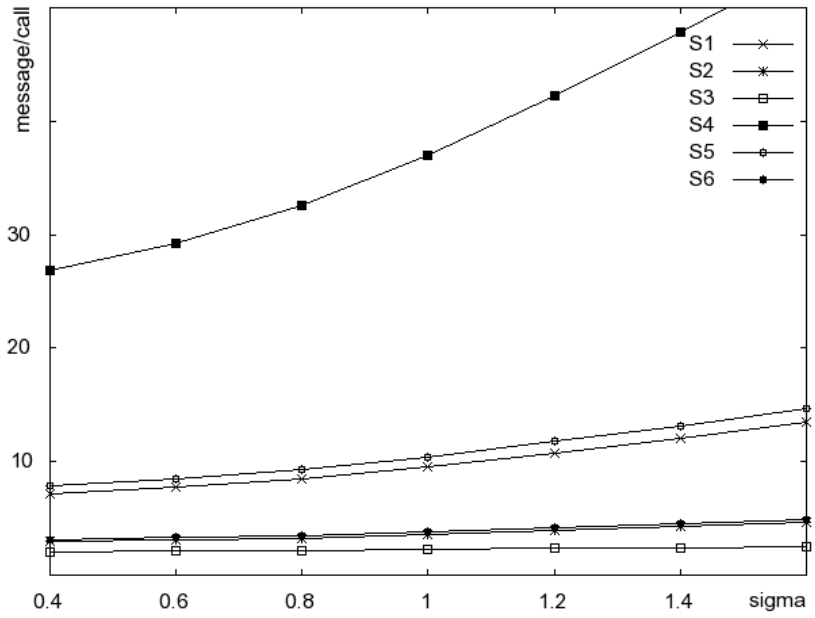

Figure 3.7: Unit reservation messages as a function of σ for the different simulation parameters

Chapter 4

New rating approach

The technological evolution and the stressed market-competition of the telecommunication industry lead to the change of approach in the price calculation of the different services in the last decade. Ten years ago, the price of the voice call (which was the only available service that time) depended mainly only on the time of the call, the called party and the duration of the call. The latter was a simple linear component in the price calculation.

Nowadays, the available tariff packages are much more complex than it used to be in a few years ago. The price of the service depends on a lot of parameters, such as the called party, the calling party, the accumulated service consumption in the current billing period, the bought services, the type of the customer, the duration of the call and many other parameters. Not even mentioning the different allowances and discounts that can be purchased separately or can be bundled with other services. The tariff packages are so complex, that it is actually hard to compare them to each other.

From the billing system point of view, there are several possible solutions to keep up with the market driven evolution of complexity[59]. One is to keep the old, legacy billing system and try to squeeze the complex pricing mechanism into it. Another one is to redesign the whole system, and take the new circumstances and the new marketing approaches into consideration. Of course, there are several companies (like Amdocs, Siemens or Ericsson) that are selling their own solutions or products for this problem[35]. These new, off-the-shelf billing systems are quite flexible, and provide the possibility to the end-users (the mobile network operators) to develop almost any marketing driven price calculation with minimal manual work. In the current billing system implementations, the pricing (rating) is performed with an (almost) arbitrary program-code. As the CDR comes in, the billing system starts to query the different database tables and applies a highly flexible algorithm to calculate the final price. Such an approach is very convenient and efficient, as developers can almost do anything and the way they want. Sadly, this flexibility does not necessarily come together with a suitable mathematical model, and does not provide the possibility to do extra calculations with the tariff packages.

With the thousands of new services and with the advent of third party providers it is crucial to be able to define the price of the service in advance, and to display it to the end users[52]. In this section, I will represent a high-level model, which is flexible enough, but holds quite a strong mathematical frame at the same time and thus, it guarantees the advice of charge (AoC) functionality and allows the communication service provider to do extra calculations

(such as income prediction) with the developed / requested offers. Such an advantage might worth the telecommunication company to replace its current billing system even if such a project is a very complex and long task and even if the performance of such a system cannot compete with the performance of an optimized program code.

4.1 The main tasks of rating

In the last few years, the task of rating has expanded. It is no longer limited to derive only the price of the service, but to calculate the *values* of the service consumption from every aspect. The value can be price, loyalty points, free gifts or anything else. These values are calculated from the deficient call detail records, the users' accumulated service consumption, the business logic, and from the user information stored in the databases of the billing- and operational system[64]. Although this seems to be a trivial problem, the technological implementation is much harder than it looks at the first glance. Finding the relevant subscriber in a database containing more than 3-5 million other entries requires huge number of reading operations, even with the correct database-indices. Moreover, not only the subscriber, customer, but the relevant services, prices, discounts and allowances should by determined, which requires even more operations from the database, and calculation from the billing system itself. Since pre-paid subscribers need a real-time approach, this must be done in milliseconds[60, 59].

Thesis 3.1: *I have identified the main functionalities of a rating module and created a new flexible yet powerful rating model which is based on the state-graph approach.* [Ary2007MS]

The following steps shall be executed when rating a call detail record.

1. Load and parse the incoming CDR.

2. Define the calling scenario and the target (chargeable) user from the information and numbers (mobile originated call, mobile terminated call, forwarded call, reverse charging, conference call, etc.).

3. Look up the target customer in the database. This step is often referred as: *guiding to customer*.

4. Look up the relevant tariff package, allowances and discounts in the database that can modify the price of the service. This step is often referred as: *guiding to service*.

5. Load the relevant accumulations that can alter the price of the service.

6. Apply the pricing logic on the call detail record based on the available information. This is the narrow-sense *rating* process.

7. Update the accumulation based on the call detail record.

8. Write and commit the rated CDR and the accumulation to the database and create a charge on the target customer's account.

Giving a mathematical formulation, rating can be described as the following function for a specific (s) service:

$$\bar{v} = r_s(\bar{a}, \bar{c}, \bar{d}), \tag{4.1.1}$$

where $\bar{v}$ is the value vector of the call, $\bar{d}$ refers to the user information and business logic stored in the databases, $\bar{c}$ denotes the actual, and $\bar{a}$ denotes the accumulated service consumption information. After a call detail record ($\bar{c}$) arrives, the accumulated values should be updated as follows:

$$\bar{a}' = a_s(\bar{a}, \bar{c}, \bar{d}). \tag{4.1.2}$$

These two high level functions shall be implemented in any rating module or system. If the prices of the calls are independent from the accumulation, then the latter equation can be omitted.

4.2 Novel rating approach

The input of function $r_s()$ can be several dimensions deep and thus, it cannot be published to the subscribers. However, we can use two abstractions, in order to display the business logic in an understandable format, and to introduce a further mathematical computation method.

AS1 Since the values of the service (the elements of $\bar{v}$) are independent from each other, these values can be calculated independently, and (more important) in parallel. These values should be published to the subscribers separately, and thus, it is enough if we focus on the computation of $v_i = r_{si}(\bar{a}, \bar{c}, \bar{d})$.

AS2 Even if the input of $r_s()$ is quite complex, the return value is discrete with a finite value-space. For a given $\bar{d}$, the number of possible return values are finite (maximum 20-30 different values are realistic). The reason is that a specific service for a specific subscriber should not have more than 20-30 different unit prices or values, in order to remain clear for the subscribers.

With these abstractions, we can use a state-graph instead of a regular function. The states in the state-graph representing the service with *similar* conditions, where the price of the service is constant u. The transitions between the states are triggered by the accumulated and by the current parameters of the call (let me denote them as transition-conditions). According to this, the price of the service can be calculated with the following steps:

S1 Calculate the actual state in the state graph, using the state-graph definition, the accumulated, and the current values (such as the length) of the call:

$$\bar{\alpha} = g(\bar{a}, \bar{c}, G_{d,s}). \tag{4.2.1}$$

S2 Calculate the value of the service from the state ($\bar{\alpha}$), from the current values of the call ($\bar{c}$) and from the unit-price vector ($\bar{u}$) using some simple function:

$$v = p_s(\bar{\alpha}, \bar{c}, \bar{u}). \tag{4.2.2}$$

S3 Update the accumulated values with the values of the current call according to the business logic:

$$\bar{a}' = a_s(\bar{a}, \bar{c}, \bar{d}), \tag{4.2.3}$$

where $G_{d,s}$ denotes the state-graph for a given service and for a given subscriber, and $\bar{\alpha}$ stands for the state indicator vector (where every element is 0 except one, which is 1). The state-graph shall be calculated for every subscriber and service and shall be updated if the customer requests or cancels a service or an allowance.

The advantage of this representation, is that $p_s()$ is much more simple that $r_s()$, because the business logic is pushed into the standard state-graph, and not into this function. Mainly $p_s()$ is a very simple function, where the value of the state (u) is multiplied by the duration of the call. The other thing is that $g()$, the state definition function, is uniform, and can be used for every service and for every subscriber. □

4.3 Advice of charge

Since the number of available services in a 3rd generational mobile network is significantly higher, than it was / it is in the GSM era, the price of the services should be presented to the subscribers before the actual service consumption in order to let the end-user decide whether to consume or cancel the requested service[52]. Since this information, this advice, should hold some kind of guarantee, our task is to approximate the price of the service as good as possible. The difference between the advised and the real price of the service can be paid by the subscriber or by the telecommunication company. Taking the first option, the subscribers should be aware, that the advice of charge has no guarantee, and it is only used as an informative service. If the deviation is paid by the provider (which means, that the AoC has its guarantee), the company should calculate this loss into its prices, in order to avoid loss of income. In this section, we will show how our model can aid the advice of charge functionality.

Basically we can divide the available services in the mobile telecommunication industry into two sets: Event and Session based services (as detailed in Section 1.4.2). The advice of charge functionality is slightly different in these cases. For the event based services, the calculation of the price is quite easy. First we calculate the actual state in the state-graph with (4.2.1), and return the value of the actual state. Computing the expected value of the service in advance is much more complex with session based services, moreover, since the length of these calls and thus, their prices can have quite a large deviation, it is better, if we calculate the advice of charge for a fixed length call only. However, calculating the expected value with variable length (with a given probability density function) may aid the economical planning of prices, this option is out of the scope of the current paper, and may be a subject for further research.

I will introduce three different models for the AoC for session based services, which share the idea of using some well-known solutions from queuing theory, but they differ in the computation requirements and in applicability. I will compare two from these three models in Section 4.4 with a simulation.

Thesis 3.2: *I have showed, that if* Π_r *denotes the transition probability matrix for a given* r *rating logic, then the price of a service can be estimated as follows:*

$$v(t) = \sum_{k=0}^{t-1} \bar{\alpha}\Pi_r^k \bar{u}, \tag{4.3.1}$$

where $\bar{\alpha}$ *is the state indicator vector,* $\bar{u}$ *is the price-vector, and* t *denotes the length of the call.* [Ary2007MS]

Since we have defined the different tariff packages with a state-graph (see Section 4.2), it is obvious to represent this graph with a state transition matrix. The elements of this matrix are the transition probabilities from one state (state i) to another (state j). If we denote this probability with p_{ij}, then the state transition matrix can be defined as:

$$\Pi = [p_{ij}]. \tag{4.3.2}$$

In this case, the transition probabilities and thus, the state-transition matrix is a function of the state-graph, the length and other parameters of the call, and the accumulated values in the current period:

$$\Pi = \Pi(G_{d,s}, \bar{c}, \bar{a}). \tag{4.3.3}$$

The value (unit price) of the service in the different states is represented with a value-vector, where the ith element of the vector represents the unit price in state i. The matrix can be calculated whenever an advice of charge functionality is initiated, and if $\bar{u}$ denotes the value-vector of the state-graph, and $\bar{\alpha}$ denotes the initial state, which is defined by (4.2.1) the expected value of a service can be evaluated with the following equation, if the call length is t ($t > 0$):

$$v(t) = \sum_{k=0}^{t-1} \bar{\alpha}\Pi^k\bar{u}. \tag{4.3.4}$$

Please note, that in case of the event based services $t = 1$, and the price of the service is $v(1) = \bar{\alpha}\bar{u}$, which is equal to the result mentioned earlier in this section. □

Regarding the performance of the AoC functionality, it requires pre-processing and offline computation in order to have the state graphs to be built up (if it is not in this format from the beginning). Once the state graphs are ready the functionality requires smaller amount of resources (CPU power) as a normal rating even if we give the theoretically optimal approximation, since the actual rating functionality is simplified to a predefined number of addition.

Thesis 3.3: *I have introduced three different models to enhance the quality of price prediction by coding the transition conditions to different states.* [Ary2007MS]

The main problem with the raw model is that the exact transition conditions are mapped into simple probabilities, and this simplification may ruin the result of the AoC functionality and may raise the deviation significantly. The approximation of the price may be much better, if we can somehow code the transition-conditions in the state-graph or in the model itself. The following subsections will give some solution for this problem.

4.3.1 Advice of charge with exploded state transition matrix

The idea of advice of charge with exploded state transition matrix (ESTM) is to code the transition-conditions into the states of the state-graph. To be more exact, we create a new graph, were one state is exploded into as many states, as the pricing logic requires it, to hold some memory for the transition condition. For example, if the transition condition is to move from state i to state j after

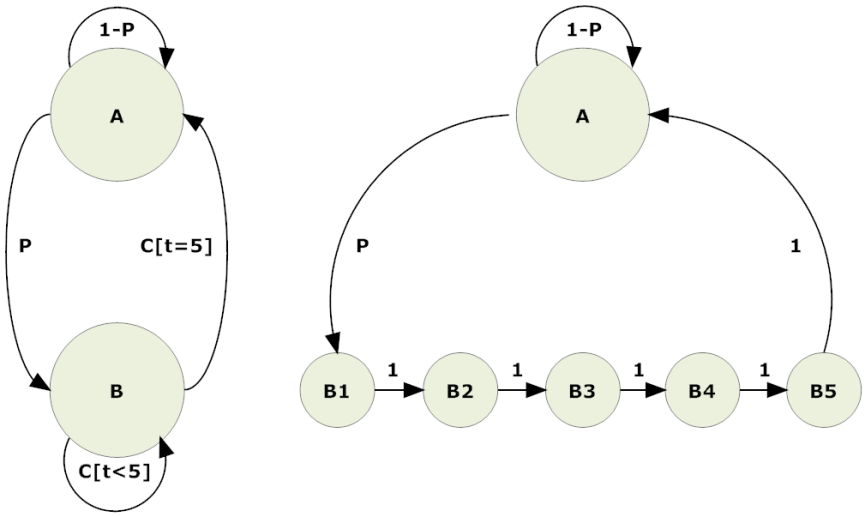

Figure 4.1: Example for AoC with ESTM

5 units (seconds), then we substitute state i with a chain of five states (from state i_1 to state i_5). The probabilities of these transitions are 1. Once the new graph is created the price of the service can be calculated with:

$$v(t) = \sum_{k=0}^{t-1} \bar{\alpha}' \Pi'^k \bar{u}'. \tag{4.3.5}$$

Let us imagine a simple service, where the price of the service is 1 EUR per second, but if we receive an MMS during the service, the price will be 0.5 EUR for the next 5 minutes. This pricing graph is represented on the left side, while the exploded one represented on the right side of Figure 4.1.

The problem of this solution is that the state-graph may be enormous, if the required memory is huge and thus, it may have a significant impact on the computation speed and on the required memory from the IT system.

4.3.2 Advice of charge with time layered model

The advice of charge with time layered model (TLM) assumes that the transition-conditions can be either mapped into probabilities without causing significant deviation from the theoretical optimum, or they are depending only on the length of the call. Nowadays, most of the available tariff packages fulfil this requirement.

We divide the state-graph into different layers. The transitions inside a layer will be mapped into probabilities, the transitions between the layers are deterministic, and they are depending only on the length of the call. If we divide the graph into G_1, G_2, G_3, ... layers with T_1, T_2, T_3, ... time, then the price of the service is:

$$v(t) = \sum_{k=0}^{T_1-1} \bar{\alpha_1} \Pi_1^k \bar{u_1} + \sum_{k=T_1}^{T_2-1} \bar{\alpha_2} \Pi_2^k \bar{u_2} + \sum_{k=T_2}^{T_3-1} \bar{\alpha_3} \Pi_3^k \bar{u_3} + \dots \tag{4.3.6}$$

where Π_i is the standard state transition matrix representation of G_i, and its value vector is $\bar{u_i}$. $\bar{\alpha_i}$ represents the initial state transition probabilities in the

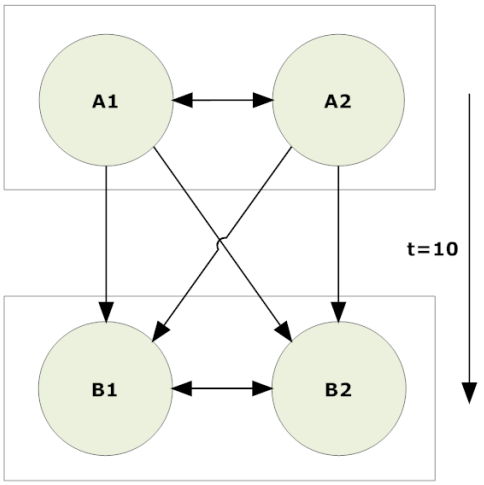

Figure 4.2: Example for AoC with TLM

ith layer. Figure 4.2 shows a simple example for this model, where we have layered the graph into $\{A1, A2\}$ and $\{B1, B2\}$ states, and $T1 = 10$.

The advantage of this model is that it uses less memory and less CPU resources, since the number of states remains low. Sadly, we cannot use this model directly, when the assumption on the transition-condition is not met. However, it is possible to use this model together with the ESTM model with the following simple algorithm: divide the graph into as many layers as possible and apply the ESTM model where necessary.

4.3.3 Stratified advice of charge

The main idea of the stratified advice of charge model (SAoC) is that some transitions may be left out from the transition graph, when we calculate the AoC. This simplification / condition should be published to the customers. Moreover, if the transition probability is quite small, but the value of the new state differs significantly from the previous one, then the precise advice of charge may even cause problems. Let us see the following example:

- The price of the voice call is 1 EUR per minute.
- If the subscriber receives an SMS during the service, then the voice call is free.

If we would calculate the AoC for a 3 minutes long call, it would be less than 3 EUR, which may cause some confusion. So, we may cut down these transitions, publish to the subscribers, that the AoC does not take the SMS into consideration, and give 3 EUR as the AoC. This idea can be used for all the above mentioned models. □

4.4 Simulation for advice of charge

Let me give a simple example. I will calculate the price of a data service (a video on demand service for instance) in advance. This service has kilobyte (KB) as unit of measure, has three different quality levels (high, medium and low) and the unit-price depends on the QoS. Let me define the tariff package as follows:

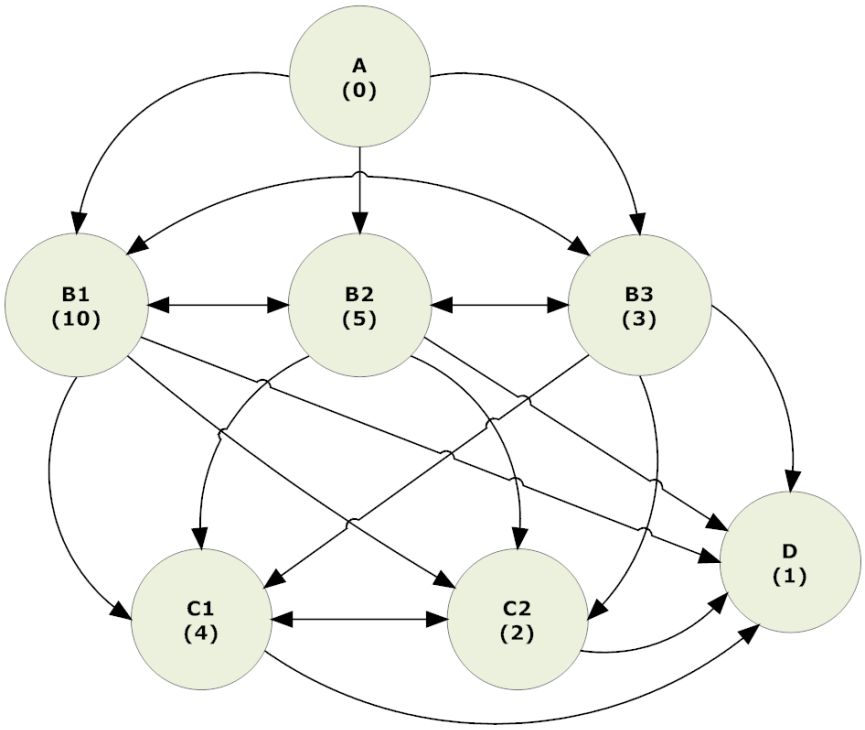

Figure 4.3: Price-graph of the tariff package

- The first 200KB is free in the month.
- The first 300KB of every call can have three different unit-prices according to the quality: 10, 5 and 3 cents per KB for high, medium and low quality level.
- The price of the rest of the call has two different prices: 4 cents per KB if the call has high or medium, and 2 cents per KB if the call has low quality level.
- The price of the service is 1 cent per KB if the subscriber receives a special SMS from someone. This price is only applied, after the first 200KB allowance consumed (if remained any).

Thesis 3.4: *I have created a simulation to show the results of our price prediction and to show the differences of the different enhancements.* [Ary2007MS]

According to this specification, the price-graph of this service has 7 different states as displayed in Figure 4.3. I have created a simulation (with 4000 runs), and I have calculated the price of the service with the Time Layered Model and with the Stratified Advice of Charge model. In the SAoC model, I have omitted the probability of receiving an SMS during the service. However, for the computations I had to make some further assumptions for the properties of the service and for the subscriber as well. I have assumed that the hidden

Table 4.1: Advice of charge simulated and calculated results

Method	Result	Approximation
Simulation	1256.10875	1260
TLM	1240.1455	1240
SAoC	1413.3333	1410

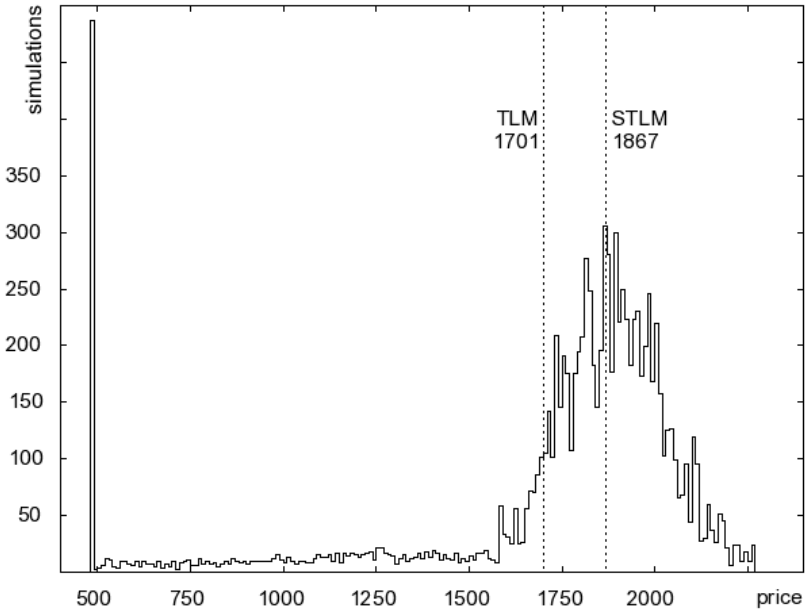

Figure 4.4: Number of simulation runs resulted the given price and the calculated expected price for the different models

state-transition matrix of the service quality is the following:

$$P = \begin{pmatrix} 0.89 & 0.1 & 0.01 \\ 0.1 & 0.8 & 0.1 \\ 0.01 & 0.1 & 0.89 \end{pmatrix} \tag{4.4.1}$$

and the subscriber already consumed 80KB from the 200KB allowance. The probability of receiving an SMS is $p = 0.0004$ during every unit.

The simulation was done in two phases. First, some simplified CDRs (Call Detail Records) were generated with a script. The mentioned parameters (probabilities, state-transition matrix, etc.) were used in order to generate life-like CDRs. As a second phase, these CDRs were rated by a simplified rater. In parallel, the TLM and SAoC models were used to predict the price of the services. The result of the simulation (average price) and the results of the two models (TLM and SAoC) are given in Table 4.1.

Figure 4.4 displays the histogram of the simulation run. The X axis represents the price of the service, and the Y axis represents the number of simulation run resulted the given price. The local peak value on the left indicates the minimum price of the service, when the subscriber received the special SMS during the first 120KB traffic. The low, constant values on the left are caused by the received SMS after 120KB. The deviation of the histogram is determined by

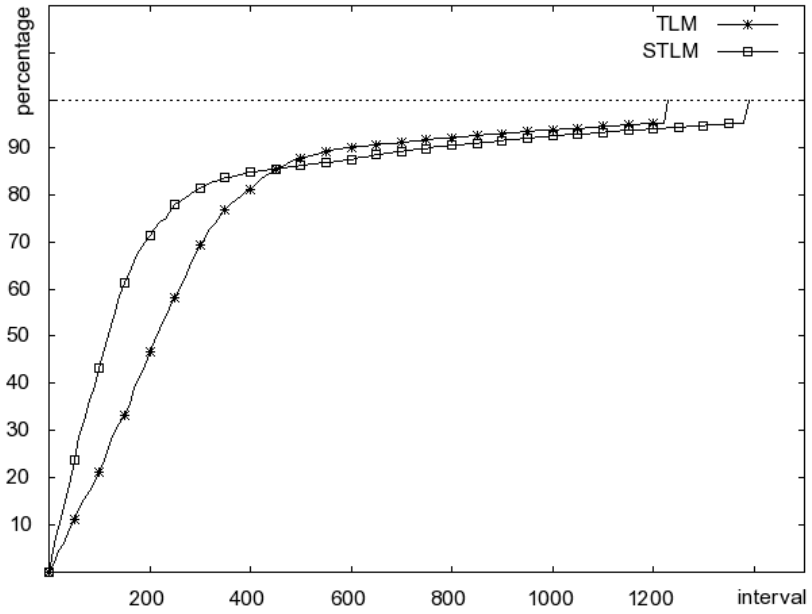

Figure 4.5: Percentage of simulation runs covered by a given threshold for the SAoC (line) and TLM (dots) model

the probability of receiving an SMS (p), the state transition matrix (P) of the quality and the value vector ($\bar{u}$) of the service.

From Table 4.1 it is clear, that the simple TLM is more efficient, if we would like to measure the efficiency with the average, however, I think, that a much better quality indicator is the *number of subscribers / simulation runs within a given threshold*. Figure 4.5 represents the percentage of simulation runs within a given tolerance. It can be seen, that the TLM model reaches the 100% confidence with a lower threshold (960 eurocents instead of 1130 eurocents), but the SAoC model reaches the 80% confidence earlier (300 eurocents instead of 390 eurocents). Since I defined the SAoC model, not to take the received SMS in consideration, and it is published to the customers, it can be more precise than the simple TLM model, since the variation caused by the SMS is removed. □

Chapter 5

Summary

In this Ph.D. dissertation I gave a comprehensive overview about the billing and charging systems used in the mobile telecommunication industry. The overview summarizes the history of the Hungarian mobile telecommunication market from 1990 to 2013, gives an insight to the used business model patterns and definitions, details the offline and online charging system architecture with the connected IT systems, lists the corresponding standardization organizations and regulations and tries to outline the most important business parameters of these systems.

In the second chapter I presented the mathematical tools and models I have created to aid the dimensioning (sizing) process of the offline and online charging systems. I presented a model and the required equations to dimension the offline charging module, if the maximal queue size and maximal record age constraint are given. I gave equations to compute the number of partial CDRs if the call length distribution is known and to define the distribution of the call detail record (CDR) arrivals. I also computed the database size required to store these partial CDRs if different attributes of the call start and call length distribution are known. Regarding online charging systems I computed the number of unit reservation messages for session based services, if some parameters of the system and the call length distribution are known. These sizing results can be used to forecast the impacts of a new service on the billing system or it can be used during a green field implementation, when a new service provider tries to penetrate the market.

In chapter three I presented a novel model and algorithm that switch between offline and online rating and thus assures proper charging in most cases with smaller administrative overhead (considering CPU power and network traffic). If the online charging protocol and services are enhanced with dynamic and preemptive unit reservation, then the amount of overhead can be reduced without the proposed mode-switching model. An algorithm was proposed and the amount of reservations and ratings were calculated in such cases. The results from the overhead reduction can be used to fine-tune or enhance the system from charging overhead point of view and reduce the required hardware and thus the different software licensing fees.

In chapter four I identified the main steps of rating, and proposed a novel rating approach, that is based on state-graphs. The new rating method determines the price of the call by defining the current state in the graph based on the call detail record and the accumulation and returns the state specific price. The model gives the possibility for the service providers and network operators

to assure the Advice of Charge functionality. A billing system was developed in a project at the Mobile Innovation Center based on this approach.

All the results presented in this dissertation were confirmed by different simulations. The results of the simulations are presented in the corresponding sections.

During my research I was endeavor to give fairly simple, yet effective mathematical formulas and algorithms. I strongly believe that the simpler a solution is, the easier it can be understood and applied in real implementations.

Bibliography

[1] Alcatel-lucent convergent payment, April 2010. http://tinyurl.com/39thbut.

[2] Huawei pre-paid platform, April 2010. http://tinyurl.com/33bey3t.

[3] UMTS Forum Report No 21. Charging, Billing and Payment Views on 3G Business Models. Technical report, UMTS Forum, 2002.

[4] 3GPP. Charging implications of IMS architecture (Release 5). TR 23.815, 3rd Generation Partnership Project (3GPP), March 2002.

[5] 3GPP. Charging architecture and principles. TS 32.240, 3rd Generation Partnership Project (3GPP), December 2008.

[6] 3GPP. Charging Data Record (CDR) file format and transfer. TS 32.297, 3rd Generation Partnership Project (3GPP), June 2009.

[7] 3GPP. Charging Data Record (CDR) parameter description. TS 32.298, 3rd Generation Partnership Project (3GPP), December 2009.

[8] 3GPP. Charging Data Record (CDR) transfer. TS 32.295, 3rd Generation Partnership Project (3GPP), October 2009.

[9] 3GPP. Customised Applications for Mobile network Enhanced Logic (CAMEL) Service description. TS 22.078, 3rd Generation Partnership Project (3GPP), December 2009.

[10] 3GPP. IP Multimedia Subsystem (IMS) charging. TS 32.260, 3rd Generation Partnership Project (3GPP), December 2009.

[11] 3GPP. Packet Switched (PS) domain charging. TS 32.251, 3rd Generation Partnership Project (3GPP), December 2009.

[12] 3GPP. Diameter Charging Applications. TS 32.299, 3rd Generation Partnership Project (3GPP), April 2010.

[13] Amdocs. Amdocs convergent charging. http://www.amdocs.com/Products/Revenue-Management/Service-Monetization/Pages/telecom-expense-management.aspx.

[14] Gregory L. Anderson, Andrew D. Flockhart, Robin H. Foster, and Eugene P. Mathews. Queue waiting time estimation, Aug 2003.

[15] GSM Association. TAP3 Implementation Handbook 1.10. TD 4006, GSM Association, November 2009.

[16] Qian bin Chen, Yu Liu, Lun Tang, and Rong Chai. Handover algorithm based on movement state for cellular relaying networks. *Proceedings of Future Computer and Communication (ICFCC)*, pages 83–85, 2010.

[17] Zsolt Butyka and Sándor Imre. Charging Concepts in UMTS Mobile Networks. *13th International Conference On Software, Telecommunications and Computer Networks*, pages 1–5, 2005.

[18] Zsolt Butyka, Tamás Jursonovics, and Sándor Imre. New fair QoS-based charging solution for mobile multimedia streams. *International Journal of Virtual Technology and Multimedia*, 1(1):3–22, 2008.

[19] Yigang Cai. Charging for long duration sessions in communication networks, March 2009.

[20] Yigang Cai and Sunil Thadani. Credit reservation transactions in a prepaid electronic commerce system, July 2004.

[21] Yigang Cai and Chun Guang Xu. System and method for communicating charging data records, April 2008.

[22] Jahangir Dadkhah Chimeh, Mohammad Hakkak, and Paeiz Azmi. Internet traffic modeling and capacity evaluation in umts. *International Journal of Hybrid Information Technology*, 1(2):109–120, April 2008.

[23] John N. Daigle. *Queueing Theory with Applications to Packet Telecommunication*. Springer Science, University of Mississippi, 2005.

[24] Mobile Europe. Billing and roaming - bill shock – sun, sea, sand and..., September 2009. `http://www.mobileeurope.co.uk/features/115067/Billing_and_roaming_-_Bill_Shock_-_Sun,_sea,_sand_and_.html`.

[25] Craig Galbraith. Amdocs, Ericsson, Huawei Lead Convergent Charging Market. *Billing and OSS world*, July 2012.

[26] Stephen C. Graves. The application of queueing theory to continuous perishable inventory systems. *Management Science*, 28(4), April 1982.

[27] Junqiang Guo, Fasheng Liu, and Zhiqiang Zhu. Estimate the Call Duration Distribution Parameters in GSM System Based on K-L Divergence Method. *International Conference on Wireless Communications, Networking and Mobile Computing, WiCom*, pages 2988–2991, September 2007.

[28] Sági Gyöngyi. Számlázási és CRM-rendszerét is integrálja a Magyar Telekom, April 2013. `http://www.bitport.hu/vezinfo/szamlazasi-es-crm-rendszeret-integralja-a-magyar-telekom`.

[29] H. Hakala, L. Mattila Category, J-P. Koskinen, M. Stura, and J. Loughney. Diameter Credit-Control Application. RFC 4006, The Internet Society, August 2005.

[30] T-Mobile Hungary. Cégtörténet, June 2010. `http://t-mobile.hu/t-mobile/ceginfo/cegtortenet`.

[31] Telenor Hungary. Cégtörténet, June 2010. `http://www.telenor.hu/telenor-magyarorszag/ceginformacio/tortenet/`.

[32] Vodafone Hungary. Cégtörténet, June 2010. http://www.vodafone.hu/egyeni/postpaid/vodafonerol/magyarorszagon/tortenet_hu.html.

[33] index.hu. Magyar mobilszolgáltató lett a Red Bull, July 2010. http://index.hu/tech/cellanaplo/2010/07/01/magyar_mobilszolgaltato_lett_a_red_bull.

[34] Van Jacobson and Michael J. Karels. Congestion avoidance and control. November 1988. http://www.cord.edu/faculty/zhang/cs345/assignments/researchPapers/congavoid.pdf.

[35] Bradley Johnson. Telecom Billing Systems - Build, Buy, Outsorce, 2003. viewed on 02/04/2005 15:39.

[36] Tamás Jursonovics, Zsolt Butyka, and Sándor Imre. Examination and New Charging Solution for Multimedia Streams over Mobile Network. *Proceedings of 3rd International Conference On Advances in Mobile Multimedia*, pages 311–320, 2005.

[37] Tamás Jursonovics and Sándor Imre. Charging, Accounting and Billing of Multimedia Streaming in 3G Mobile Networks. *Proceedings of 14th IST Mobile Summit 2005*, pages 1–5, 2005.

[38] Hemlata S. Kadia, Lior Auslander, Benger Geoffrey Coleman, Allen Christopher Doehler, Bruce Frankel, Gabriel Matsliach, David M. Sell, and Pankaj Trivedi. Network communication service using multple payment modes, March 2012.

[39] Stefan Karlsson. Optimized reservation for multi-session and/or multi-unit types, December 2009.

[40] F. Khan and Baker. N. Charging data dimensioning in 3G mobile networks. *Proceedings of 3G Mobile Communication Technologies*, April 2004.

[41] Maria Koutsopoulou, Alexandros Kaloxylos, Athanassia Alonistioti, and Lazaros Merakos. Charging, Accounting and Billing Management Schemes in Mobile Telecommunication Networks and the Internet. *IEEE Communications Surveys, First Quarter 2004*, 6(1), 2004.

[42] Dr. Matthew Lucas. Where Should Rating be Implemented. *Billing World and OSS today*, October 2004.

[43] Michael Joseph Magnotta, Derron Keith Newland, Frank Clifford Perkins, and Joseph Difonzo. Prepaid reservation-based rating system, April 2008.

[44] Mark Myatt, Felim O'neill, Malcolm Crouch, Michael Jenvey, Graham Agnew, and Michael Rosenberg. Real-time reservation of charges for pre-paid services, September 2002.

[45] Keith Newman. Single face on multiple billing streams. *Telecommunications Review*, April 2004.

[46] NHH. Mobilhang havi gyorsjelentés, April 2013. http://nmhh.hu/cikk/157387.

[47] Jarmo Niemi. Convergent charging - unlocking value. *Latin America 2007*, 2007.

[48] Origo.hu. Már minden magyarra jut egy mobil, May 2007. http://www.origo.hu/uzletinegyed/hirek/20070525marminden.html.

[49] Ericsson White Paper. Prepaid postpaid convergent charging. April 2005.

[50] Raghu Prasad. Comverse client & partner success conference 2007. 2007.

[51] Ali Rajabi and Farhad Hormozdiari. Time constraint m/m/1 queue, 2006.

[52] Alejandro Ramirez. Advise of Charge: Implications for Mobile Data Strategies, 2002. CSG Mobile Series Whitepaper.

[53] B. Rottembourg. Call center scheduling, 2002.

[54] BME GTK Kommunikáció és médiatudomány (BA) szakosok oldala. 0660, June 2010. http://p2p-fusion.mokk.bme.hu/w2/index.php/-_0660.

[55] DJuice sajtóközlemény. A djuice kiválik a pannonbol, April 2010. http://www.djuice.hu/a-djuice-kivalik-a-pannonbol.

[56] Magyar Telekom sajtóközlemény. Megszűnik a westel 0660 szolgáltatása, April 2003. http://sajtoszoba.magyartelekom.hu/process?action=notice&id=1572.

[57] Vodafone Hungary sajtóközlemény. A mobil távközlési piac új szereplővel bővül, amelynek szolgáltatását 721 postán lehet igénybe venni, November 2009. http://www.vodafone.hu/egyeni/prepaid/vodafonerol/magyarorszagon/sajtoanyagok/sajtokozlemenyek/091120_hu.html.

[58] Tor Schoenmeyr and Stephen C. Graves. Strategic safety stocks in supply chains with capacity constraints, 2009.

[59] Susana Schwartz. Next-Gen Rating: It Will Be Only As Good as the Network. *Billing World & OSS Today*, February 2003.

[60] Susana Schwartz. Prepaid's Untapped Potential. *Billing World and OSS today*, July 2003.

[61] Mallász Judit Sciamus. Számlázási anomáliák, September 2007. http://www.sciamus.hu/public/doc/szamlazasi_anomaliak.pdf.

[62] Yuri Shtivelman. Method for estimating telephony system-queue waiting time in an agent level routing environment, May 2005.

[63] Sok-Ian Sou, Hui-Nien Hung, Yi-Bing Lin, Nan-Fu Peng, and Jeu-Yih Jeng. Modeling credit reservation procedure for umts online charging system. *IEEE Transactions On Wireless Communications*, 6(11):4129–4135, November 2007.

[64] David Stark. The Changing Role of Billing. *CRM Today*, December 2002.

[65] Syniverse. Roaming solutions, June 2010. http://www.syniverse.com/business-solutions/roaming-solutions.

[66] Koi Tamás. A legolcsóbb feltöltőkártyás tarifa lehet a Blue Mobile, January 2012. http://www.hwsw.hu/hirek/48009.

[67] Telecoms.com. EU data roaming regulations and the rise of personalised user policies, June 2010. http://www.telecoms.com/15041/.

[68] tescomobile.hu. Március 1-jén indul a Tesco Mobile, February 2012. http://www.tescomobile.hu/usr/sajtokozlemeny-12-02-28.pdf.

[69] Robert Tornkvist and Ralph Schubert. Ericsson convergent charging and billing. *Ericsson Review 2009*, 2009.

[70] Dara Ung. Prepaid/postpaid automatic change of payment option, April 2003.

[71] Wikipedia. List of mobile network operators, June 2010. http://en.wikipedia.org/wiki/List_of_mobile_network_operators.

[72] Wikipedia. Mobile virtual network operator, June 2010. http://en.wikipedia.org/wiki/Mobile_virtual_network_operator.

[73] Wikipedia. Parlay, June 2010. http://en.wikipedia.org/wiki/Parlay_Group.

[74] Wikipedia. Telecoms & internet converged services & protocols for advanced networks, June 2010. http://en.wikipedia.org/wiki/Telecoms_%26_Internet_converged_Services_%26_Protocols_for_Advanced_Networks.

[75] WolframAlpha. Error Function - ERF, June 2010. http://mathworld.wolfram.com/Erf.html.

[76] Zhiquan Yi, Fei Zhang, and Wei Zhang. Charging system and charging method, December 2010.

[77] Jin Zhang. A method for selecting/switching the charging mode and the device thereof, January 2010.

[78] Jian Zhu. Seamless switching between pre-paid and post-paid charging, August 2010.

[79] Yuanping Zou and Bo Jia. A New Method for CDR Processing in IP Multimedia Subsystem. *Business and Information Management, International Seminar on*, 2:221–224, 2008.

[80] M. Zukerman, T.D. Neame, and R.G. Addie. Internet traffic modeling and future technology implications. *Proceedings of INFOCOM 2003*, pages 587–596, 2003.

Publications

Book chapters

[Ary2010ENC] Bálint Dávid Ary, and Dr. Sándor Imre. Charging and Rating in Mobile Telecommunication Networks, *Next Generation Mobile Networks and Ubiquitous Computing*, Editor: Samuel Pierre, 2010, pp43-50. ISBN: 9781605662503.

International journal papers

[Ary2012TS] Bálint Dávid Ary, and Dr. Sándor Imre. Partial xDR database dimensioning, *Telecommunication Systems, Springer Verlag*, 2012. ISSN: 1018-4864.

[Ary2011PP] Bálint Dávid Ary, and Dr. Sándor Imre. Comparison of Pre-paid rating methods, *Periodica Polytechnica - Electrical Engineering*, 2011, Vol.55, No.1, pp5-11. ISSN: 0324-6000.

[Ary2005GESTS] Bálint Dávid Ary, and Dr. Sándor Imre. Reduction of Charging Overhead in Mobile Telecommunication Networks, *GESTS International Transactions*, 2005, Vol.20, No.1, pp126-136. ISSN: 1738-6438.

International conference proceedings

[Ary2010ARR] Bálint Dávid Ary, and Dr. Sándor Imre. xDR Arrival Distribution. *33rd International Conference on Telecommunications and Signal Processing 2010.* Appears in Proceedings of TSP 2010, pp417-420. ISBN: 978-963-88981-0-4.

[Ary2010SIZ] Bálint Dávid Ary, and Dr. Sándor Imre. Sizing of xDR Processing Systems. *5th International ICST Conference on Access Networks 2010.* Appears in Proceedings of AccessNets 2010, Springer, pp62-70. ISBN: 978-3-642-20930-7.

[Ary2007MS] Bálint Dávid Ary, and Dr. Sándor Imre. Advice of Charge in Telecommunication Services, *The 16th IST Mobile and Wireless Communications Summit 2007.* Appears in Proceedings of IST Mobile Summit, 2007. ISBN: 963-8111-66-6.

[Ary2005CON] Bálint Dávid Ary, Gábor Debrei, and Dr. Sándor Imre. Real-Time Charging in Third-Generation Mobile Networks, *CONTEL 2005 Conference.* Appears in Proceedings of CONTEL, 2005, pp239-245. ISBN: 9789531840828.

[Ary2005EUN] Bálint Dávid Ary, Gábor Debrei, and Dr. Sándor Imre. Overhead Reduction for Real-Time Charging in UMTS Networks, *EUNICE 2005 Conference*. Appears in Proceedings of EUNICE, 2005, pp67-71. ISBN: 978-0387308159.

[Ary2004DT] Bálint Ary, and Dr. Sándor Imre. Real-time Charging in UMTS Environment, *Digital Technologies Conference: Optical and wireless technologies*. Appears in Proceedings of Digital Technologies, 2005, pp140-145. ISBN: 80-8070-334-5.

Hungarian journal papers

[Ary2006HT] Bálint Dávid Ary, and Dr. Sándor Imre. Számlázás Újgenerációs Telekommunikációs Hálózatokban, *Híradástechnika*, 2006, Vol.LXI, No. 2006/10, pp40-45. ISSN: 0018-2028.

[Ary2005HTS] Bálint Dávid Ary, and Dr. Sándor Imre. Real-time Charging in Mobile Environment, *Híradástechnika - Selected Papers*, 2005, Vol.LX, No.6., pp54-59. ISSN: 0018-2028.

[Ary2005HT] Bálint Ary, and Dr. Sándor Imre. Valós idejű számlázás mobil környezetben, *Híradástechnika*, 2005, Vol.LX, No.2005/1, pp25-29. ISSN: 0018-2028.

[Ary2004MT] Bálint Ary, and Dr. Sándor Imre. UMTS rendszerek valós idejű számlázásának problémái, *Magyar Távközlés*, 2004, Vol.XV. No.2, pp29-32. ISSN: 0865-9648.

Hungarian conference proceedings

[Ary2004TDK] Bálint Ary, and Gábor Debrei. UMTS rendszerek valós idejű számlázása, *Students' Scientific Conference*, Organised by BUTE in 2004 and by ELTE in 2005.

Citations

[Ary2005CON] Kashyap dhruve, Prakash S, and C B Akki. Unified Billing-Realization of convergent architecture for charging and billing in 4G networks, *International Journal on Information Technology* 2011, Vol. 1, Nr. 3, p7. ISSN: 2158-012X.

[Ary2005CON] Mark de Reuver, Tim de Koning, Harry Bouwman and Wolter Lemstra. How new billing processes reshape the mobile industry. *Info - The journal of policy, regulation and strategy for telecommunications*, 2009, Vol.1, pp78-93. ISSN: 1463-6697.

[Ary2005CON] SuJung Yu, SungMin Yoon, JungKap Lee, HyoJin Kim, and JooSeok Son. Service-Oriented Issues: Mobility, Security, Charging and Billing Management in Mobile Next Generation Networks, *Broadband Convergence Networks 2006, - BcN 2006*, 2006, pp1-10, ISBN: 1-4244-0146-1.

[Ary2005GESTS] Balázs Leitem, and Zoltán Windisch. Rating in Next Generation Mobile Networks based on a directed graph, *48th International Symposium ELMAR-2006 focused on Multimedia Signal Processing and Communications*, 2006, ISBN: 953-7044-03-3.

[Ary2005HTS] Zsolt Butyka, Tamás Jursonovics, and Dr. Sándor Imre. New fair QoS-based charging solution for mobile multimedia streams, *International Journal of Virtual Technology and Multimedia*, 2008, Volume 1, Issue 1, pp:3-22, ISSN:1741-1874.

[Ary2004DT] Asad Ahmad Khan, Mo Adda, and Carl Adams. Convergence of terrestrial and satellite mobile communication systems: an operator's perspective, *International Journal of Mobile Communications*, 2009, Volume 7, Issue 3, pp:308-329, ISSN:1470-949X.