Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Prof. Dr.-Ing. P. Tran-Gia

Modeling and Evaluation of Multi-Stakeholder Scenarios in Communication Networks

Christian Schwartz

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Bericht 3/15

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Herausgeber

Prof. Dr. Ing. P. Tran-Gia
Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Am Hubland
D-97074 Würzburg

Tel.: +49-931-31-86630
Fax.: +49-931-31-86632
email: trangia@informatik.uni-wuerzburg.de

Satz

Reproduktionsfähige Vorlage vom Autor.
Gesetzt in LaTeX Linux Libertine 10pt.

ISSN xxxx-xxxx

Modeling and Evaluation of Multi-Stakeholder Scenarios in Communication Networks

Dissertation zur Erlangung des
naturwissenschaftlichen Doktorgrades
der Julius–Maximilians–Universität Würzburg

vorgelegt von

Christian Schwartz

aus

Würzburg

Würzburg 2015

Eingereicht am: xx.xx.2015
bei der Fakultät für Mathematik und Informatik
1. Gutachter: Prof. Dr.-Ing P. Tran-Gia
2. Gutachter: Prof. Dr. Franco Davoli
Tag der mündlichen Prüfung: xx.xx.2015

Contents

1 Introduction

1.1 Considered Stakeholders

1.2 Scientific Contribution

1.3 Outline of Thesis

2 Network

2.1 Background and Related Work

2.1.1 UMTS Networks and the RRC Protocol

A Third Generation (3G) Universal Mobile Telecommunications System (UMTS) mobile network consists of three main components, which are depicted in Figure 2.1: The User Equipment (UE), the Radio Access Network (RAN), and the Core Network (CN). The RAN is used to establish connectivity between the UE and the CN, which in turn can establish connectivity to the Internet, if required.

UE consists of devices used by end users, i.e. smartphones, tablets or data card enabled notebooks, but can also include Machine to Machine (M2M) devices. The RAN is, amongst other tasks, responsible for Radio Resource Control (RRC), packet scheduling and handover control. It includes network entities such as the NodeB and the Radio Network Controller (RNC). The CN provides the backbone network of the UMTS network and provides connectivity to the Internet and the Public Switched Telephone Network (PSTN). Furthermore, functionality such as billing, authentication and location management is provided by the CN.

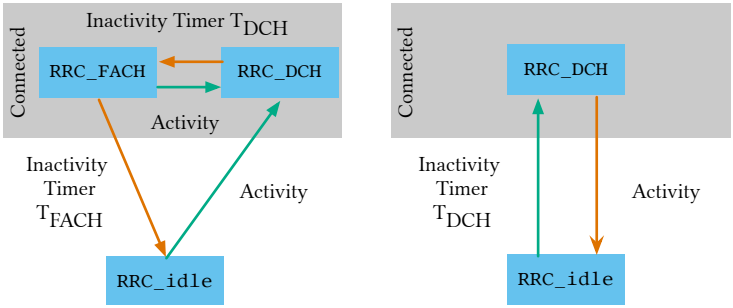
In UMTS networks, the radio resources in the RAN between base station and UE are controlled and managed by the RRC protocol [35]. The protocol offers services such as broadcast of network information, maintenance of a connection between the UE and RAN, establishment of point-to-point radio bearers for

Figure 2.1: Overview of Mobile Network

data transmission, Quality of Service (QoS) control, and reporting and cell selection management. The protocol is divided into different parts: services for upper layers, communication with lower layers, protocol states, RRC procedures, and error control. In particular, RRC also participates in the co-ordination of other resource management operations such as channel measurements and handovers. All RRC procedures rely on protocol states which are defined to trigger action should be applied and which information must be signaled. The state are defined per UE and for the connection between the UE and the NodeB station. Typically there are five RRC states characterizing a connection between UE and NodeB: `idle`, URA_PCH, CELL_PCH, RRC_DCH, and RRC_FACH. Whether a specific RRC state is used in a specific mobile network depends on the configuration of the network by the provider. In the following we concentrate on the most commonly observed [21] RRC states idle mode (`idle`), Dedicated channel (CELL_DCH), and Forward access channel (CELL_FACH). We neglect URA_PCH and CELL_PCH in this study. While URA_PCH plays only a role in scenarios of high mobility, CELL_PCH is not yet widely implemented. Our results are still of general nature and do not depend on the limited number of considered RRC states.

If the UE is switched on and no connection to the mobile network is established, the UE is in `idle` state. If the UE wants to send data, radio resources are allocated by the NodeB for the handset and the UE will transition to either the CELL_FACH or the CELL_DCH state. Then, a corresponding channel for data transmission is assigned to the UE. The CELL_FACH and the CELL_DCH state can be distinguished in that way that in CELL_DCH state a high-power dedicated channel for high speed transmission is allocated whereas in CELL_FACH state a shared access channel for general sporadic data transmission is used. Thus, CELL_FACH consumes significantly less power than the CELL_DCH state.

The possible transitions between the different states are defined by the network operator and the RRC protocol stack. Typically, the following state transitions are included: `idle` $\rightarrow$ CELL_FACH, CELL_FACH $\rightarrow$ CELL_DCH to switch from lower radio resource utilization and low UE energy consumption to an-



(a) Three State Scenario *(b) Two State Scenario*

Figure 2.2: RRC State Machine Diagrams

other state using more resources and energy, and CELL_DCH → CELL_FACH, CELL_FACH → idle, CELL_DCH → idle to switch to lower resource usage and energy consumption. According to [21, 22], the transitions are triggered by user activity and radio link control buffer level. A transition from CELL_DCH to CELL_FACH usually occurs when the buffer is empty and a threshold for a release timer is exceeded, resulting into the corresponding RRC protocol message flow. A transition in the reverse direction is triggered if the buffer level exceeds a specified threshold value for a predefined time period. The UE will transition into idle state if the RNC detects overload in the network or no data was sent by the UE for a specified time.

In the following, we consider two different state transition models, depicted in Figure 2.2, based on the RRC protocol. The first model includes the idle, CELL_FACH, and CELL_DCH states is shown in Figure 2.2a and is in the following called the three state model. If the UE is in the idle state and activity is detected, i.e. a packet is sent or received, the connection transitions to CELL_DCH state. After each transmission a timer T_{DCH} is started and reset whenever a new packet is sent or received. If the timer expires, the connection transitions to the CELL_FACH state Upon entering, the T_{DCH} timer is started. If a new transmis-

sion occurs, the connection again transitions to the CELL_DCH state. If T_{DCH} expires, the connection transitions to `idle` state.

The second model, denoted as the two state model, and shown in Figure 2.2b, only includes the `idle` and CELL_DCH state. If the UE is in the `idle` mode and a packet is sent or received, the connection transitions to the CELL_DCH state. Once in CELL_DCH mode, the T_{DCH} timer is started and it is reset whenever a new packet is sent or received. If the timer expires, the UE transitions back to `idle` state.

While the three state model is closer to the specified RRC protocol is similar to some proprietary *Fast Dormancy* implementations used by UE vendors. In these Fast Dormancy implementations, the UE tears down the connection to the network state as soon as no data is ready to be sent for a certain time, i.e., it forces the network to transition to `idle` state. In contrast to the three state model, there is no transition to the CELL_FACH state. If a device disconnects from the network by transitioning to the `idle` state, it has to be reauthenticated before another transition to the CELL_DCH state can occur. This results in additional signalling traffic and causes more load on the network [36] due to frequent re-establishments of the RRC connection. These proprietary Fast Dormancy algorithms do not adhere to the RRC specification [37], but nontheless exist in the real world and have been identified as possible causes for signalling storms. The major reason for Fast Dormancy implementations is the decrease in power consumption on the UE, since the transmission unit of the UE consumes only 1 % to 2 % of the energy in `idle` state compared to the CELL_DCH state. Thus, both models warrant further investigation.

2.1.2 Measurements of RRC Parameters and Optimisation of Resource Consumption

In the literature, the configuration of the inactivity timers used for the RRC protocols have been investigated in detail. In [22] a measurement tool for RRC protocol states is presented. It is used to determine RRC state transition param-

eters, channel setup delays, and paging delay by measuring the one-way round trip time of data packets. The results are validated by monitoring the energy consumption in different RRC states. One outcome is that UMTS network configurations vary significantly by network operator. CELL_DCH release timer as well as the inactivity timer value triggering transition to `idle` state were measured. The values range from 1.2 s for the CELL_DCH release timer to more than one minute for the `idle` timer. Similar results are presented in [21]. Here, the observed values vary between 5 s and 12 s. Additionally, they also determined the exact RRC state transitions for two networks such as `idle` → CELL_FACH → CELL_DCH or `idle` → CELL_DCH directly without transitioning through the CELL_FACH state. The 3rd Generation Partnership Project (3GPP) has released a technical report [38] about the adverse impact of mobile data applications. This report states that frequent connection re-establishments due to small data packets caused e.g. by status updates of social network or instant messaging apps can lead to problems of increased signalling load. This highlights the importance of this topic.

Furthermore, there are papers that propose optimizing strategies that take the RRC states into account. In [23] the impact of different application traffic patterns is studied to reveal resource usage in mobile networks. By identifying packet bursts, they infer the RRC states of the UE. Radio resources are quantified by channel occupation time and energy consumption. They propose an algorithm that tries to optimize application traffic patterns by e.g. piggybacking, batching up data, or decreasing the update rate of an application. The algorithm is evaluated for six applications, two news applications, Pandora streaming application, Google search, Tune-In radio and Mobelix. In [24] also RRC states are studied for network optimization. The authors optimize the inactivity timers to allow a better resource utilization. They propose a application-to-network interface to avoid unnecessary timer periods after data transmission.

2.1.3 Smartphone Power Consumption and QoE

Power consumption of the UE varies according to the devices current RRC state. The power consumption caused by CELL_DCH mode was measured at about 600 mW to 800 mW [21, 23]. In CELL_FACH mode, the consumption was measured at about 400 mW to 460 mW depending on the UE and the network operator [21]. A precise measurement of the power consumption of different RRC states is performed in [21, 25, 26]. The authors report, that the energy drain depends on two factors: *a*) user interactions and applications *b*) platform hardware and software.

In [6] the authors performed a 4 week long study with 29 participants to identify factors influencing Quality of Experience (QoE) of mobile applications. The study comprises *a*) data from context sensing software, *b*) user feedback using an experience sampling method several times per day, and *c*) weekly interviews of the participants. To determine the factors of influence, the authors analyze the frequency of specific keywords in the interviews and the surveys. They find, that the term *battery* has the highest frequency. According to the authors this is reasonable since the battery efficiency has a strong impact on the user perceived quality, in particular when it the UE is nearly discharged.

2.2 Inferring Signalling Frequency and Power Consumption from Network Traces

[4]

2.2.1 Performance Evaluation

Measurement Procedure and Setup

To investigate the behaviour of the application under study, we capture traffic during a typical use of the application on a *Samsung Galaxy SII* smartphone. The smartphone runs the Android operating system and is connected to the

3G network of a major German network operator. To obtain the network packet traces we use the `tcpdump` application. This application requires *root* privileges which are obtained by rooting the device and installing the custom *cyanogenmod* ROM [1]. Once `tcpdump` is installed and running, we start the application under study and capture packet traces while the application is running. Then, the *android debugging bridge* is used to copy the traces to a workstation. The traces contain Internet Protocol (IP) packets as well as Linux Cooked Captures. We only require the IP packets, thus we filtered the traces for IP packets which are used during the following analysis.

Inferring Network State

In this section we study the influence of the application traffic on RRC state transitions and signalling messages. Since RRC state transitions can not be captured using commonly available tools, we introduce an algorithm to infer RRC state transitions from IP packet traces. Using this algorithm we analyse the RRC state transition frequency and signalling message load for the two state model and three state model.

Traffic below the network layer can not be measured without specific equipment, which is often out of reach for developers interested in assessing the impact of their applications on the network. Based on the two state and three state models introduced in Section 2.1.1 we process `tcpdump` captures of the application traffic. However, it should be noted that this method is not restricted to a specific network model, but can be extended to any other network model, as well. Using these captures, we extract the timestamps when IP packets are sent or received. Furthermore, we require the timer values of the transition from CELL_DCH state to CELL_FACH state, T_{DCH}, and the timer for the transition between CELL_FACH and `idle` states, T_{DCH}. Based on these informations Algorithm 1 infers the timestamps of state transitions according to the 3GPP specification [35] for the three state model. This algorithm can be simplified to also

[1] http://www.cyanogenmod.org

work for the two state model. Alternatively, a method to post process the results of the algorithm to obtain results for the two state model is given at the end of this section. The algorithm first computes the inter-arrival times of all packets. Then, each timestamp is considered. If the UE is currently in `idle` state, a state transition to CELL_DCH occurs at the moment the packet is sent or received. If the inter-arrival time exceeds the T_{DCH} timer the UE transitions to CELL_FACH T_{DCH} seconds after the packet was sent or received. Similarly, if the inter-arrival time exceeds both the T_{DCH} and T_{DCH} timers, a state transition to `idle` occurs T_{DCH} seconds after the state transition to CELL_FACH.

Algorithm 1 Inferring RRC State Transitions Based on IP timestamps

Require: Packet arrival timestamps *ts*
  CELL_DCH to CELL_FACH timer T_{DCH}
  CELL_FACH to `idle` timer T_{DCH}
Ensure: Times of state transition *state_time*
  New states after state transitions *state*
  `interarrival(i)` ← *ts*(i+1) - *ts*(i)
  `index` ← 0
  for all ts(i) **do**
    if `state(index)` = `idle` **then**
      `index` ← `index` + 1
      `state(index)` ← CELL_DCH
      `state_time(index)` ← ts(i)
    end if
    if `interarrival`(i-1) $> T_{\text{DCH}}$ **then**
      `index` ← `index` + 1
      `state(index)` ← CELL_FACH
      `state_time(index)` ← ts(i) $+T_{\text{DCH}}$
    end if
    if `interarrival`(i-1) $> T_{\text{DCH}} + T_{\text{DCH}}$ **then**
      `index` ← `index` + 1
      `state(index)` ← `idle`
      `state_time(index)` ← ts(i) $+T_{\text{DCH}} + T_{\text{DCH}}$
    end if
  end for

UE vendors always search for ways to decrease energy consumption of their

from/to	idle	CELL_FACH	CELL_DCH
idle	–	28	32
CELL_FACH	22	–	6
CELL_DCH	25	5	–

Table 2.1: Number of Signalling Messages per RRC state transition perceived at the RNC (Taken From [35])

devices. A straightforward way to achieve this, if only the wellbeing of the UE is considered, is to transition from CELL_DCH state to idle as soon as no additional data is ready for sending. While this transition is not directly available in the 3GPP specification for the RRC protocol [35], a UE may reset the connection, effectively transitioning from any state to idle. This behaviour can be modelled using the two state model introduced in Section 2.1.1.

State transitions for the two state model can be calculate using a similar algorithm. Alternatively, the behaviour of the two state model can be emulated using Algorithm 1 if T_{DCH} is set to 0 s and all state transitions to CELL_FACH are removed in a post processing step.

Calculating Signalling Frequency and Power Consumption

In reality, the number of state transitions is not the metric of most importance if network signalling load should be evaluated. Each state transition results in a number of RRC messages between the UE and different network components. For this study we consider the number of messages observed at the RNC, which can be found in [35] and is summarized in Table 2.1. It can be seen that transitions from or to the idle state are especially expensive in terms of number of messages sent or received. This is due to the fact that upon entering or leaving the idle state authentication has to be performed. Note that for the two state model only transitions from or to the idle state occur. This results in the fact that for the same network packet trace the number of signalling messages occurring in the two state model is generally higher than in the three state model. To

RRC State	Power Consumption
idle	0 mW
CELL_FACH	650 mW
CELL_DCH	800 mW

Table 2.2: Power Consumption of the UE Radio Interface Depending on Current RRC State (Taken From [23])

obtain the total number of signalling messages, we weight the number of state transitions with the number of messages sent per state transitions. Then, we average the number of state transitions over the measurement duration to obtain a metric for the signalling load at the RNC. The inference algorithm does not differentiate between state changes caused by upstream or downstream traffic. State changes caused by downstream traffic usually generate some additional signalling messages, as paging is involved. The inference algorithm can be easily enhanced to support this behaviour. However, the results discussed in the next section would only change quantitatively. Furthermore, the inference of signalling messages can be easily adapted to new networking models or signalling numbers.

From a users point of view, the signalling message frequency is of little importantance. The user is interested in a low power consumption because this increases the battery life of the device. To calculate the battery life, we use the time when state transitions occurred, and the information about the state the transition was to, to calculate the relative amount of time that was spent in each state. Given the relative time spent in each state, we use Table 2.2 (taken from [23]) to compute the power consumption of the radio interface during the measurement phase. We focus on the power consumption of the radio interface, as it is possible to measure the aggregated power consumption using out of the box instrumentation techniques provided by the hardware vendor.

Application	Traffic Characteristic	Application Use	Required Bandwidth
Angry Birds	Interactive	Foreground	Low Bandwidth
Aupeo	Interactive	Background	High Bandwidth
Twitter	Periodic, Low frequency	Background	Low Bandwidth
Skype	Periodic, High frequency	Background	Low Bandwidth

Table 2.3: Qualitative Characterization of Applications Under Study

2.2.2 Numerical Results of Measurement Study

Characterization of Traffic Patterns for Selected Applications

For this study we chose four specific applications in order to cover a broad spectrum of traffic characteristics, as described in Table 2.3. First, we discuss said characteristics for these applications. We differentiate between applications, where the user interaction causes the generation of traffic, and such, where the application periodically sends or receives traffic. Finally, we consider the amount of bandwidth used by the application.

Angry Birds for Android is a popular *interactive* free-to-play game and runs in the *foreground*. To finance the game, an advertisement is shown once the player starts or restarts a level. Advertisements are downloaded on demand by the application, but require *low bandwidth*. Thus, the time between two advertisements depends on the frequency of the player advancing to the next level or deciding to restart the current one.

Aupeo is an Internet radio application, allowing a user to listen to content from personalised radio stations, while running in the *background*. Content is not streamed but downloaded at the beginning of the track. The exact duration depends on the radio stations chosen by the user and is thus *interactive*. This results in large times of inactivity during the playback of the track itself. Due to the fact that audio files are downloaded, there is a *high bandwidth* requirement.

The **Twitter** client is used to send and receive new short messages from the user's Twitter account. Transferring these messages requires relatively *low*

bandwidth. To this end, the user can specify an update frequency when to pull new messages in the *background*. Thus, the downloads occur with a *periodic behaviour of low frequency*, where the client sends an HTTPS request to the Twitter server and in return receives new Tweets for the user's account. We do not consider an active user who is publishing new Tweets. Such behaviour would manifest as additional traffic to the periodic one generated by the status updates. Due to the fact that publishing updates occurs relatively infrequent, and updating the feed occurs more often, the traffic generated by publishing updates is dominated by that occurring due to updates, and thus can be neglected.

Finally, we consider the **Skype** application. We do not consider any VoIP calls, but the application's idle behaviour, i.e. when the application is running in the *background*. During this time, the application sends keep-alive messages to the network. These keep-alive messages are sent with a *high frequency* and require *low bandwidth*.

In addition to the applications considered, there exist other categories of applications which are running in the *foreground* and *interactively* require a *high bandwidth*. One example for such an application is Skype while taking a VoIP call. These applications are not considered in this study, because this kind of behaviour causes the UE to be always online. This results the minimal amount of signalling messages to be sent and a maximal power consumption at the UE, independent of network model or used parameters. Other combinations of traffic criteria also exist. However, from both a signalling load as well as a power consumption point of view, they can be mapped to one of the discussed cases. For example, if an application is sending periodic updates with low bandwidth without user interaction, then the fact that the application is running in the foreground or the background is without consequence for the generated signalling load or power consumption. However, these cases should be considered when optimisation strategies for message sending are under study. For example background applications could allow for the batching of messages, because the transmission is usually not urgent, while foreground applications do not allow for such behaviour because it would decrease QoE.

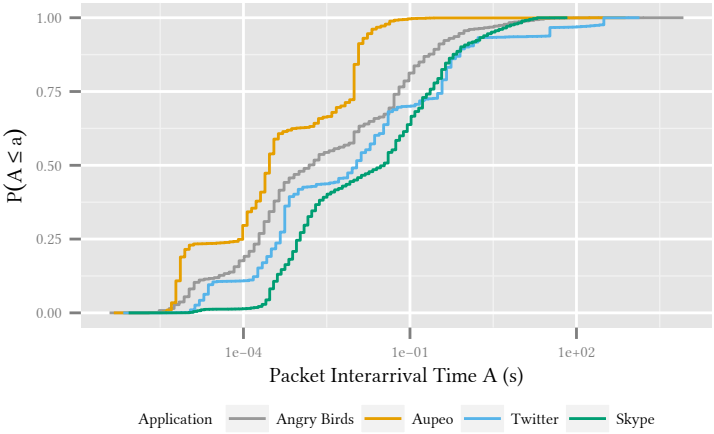


Figure 2.3: CDF of Interarrival Times for Considered Applications

Next, we describe the applications under study in more detail. For each application we show the Culmulative Density Function (CDF) of the interarrival times in Figure 2.3 and give information about the mean values and standard deviation of both interarrival times and bandwidth in Table 2.4, respectively.

Let us again begin with the Angry Birds application. We see that there are no distinct peaks in interarrival time, which would hint at periodic behaviour. Furthermore, we see that $5\,\%$ of all interarrival times are greater than $1\,\mathrm{s}$. As we consider only T_{DCH} values above $1\,\mathrm{s}$, those are candidates for triggering state transitions. The mean interarrival time is $0.66\,\mathrm{s}$, with a relatively high standard deviation of $15.90\,\mathrm{s}$. This is caused by the low interarrival times in one advertisement request and the relatively large interarrival times between two advertisements. Mean bandwidth is relatively low with $4.42\,\mathrm{kbit\,s^{-1}}$ and a high standard deviation of $4.5\,\mathrm{kbit\,s^{-1}}$. These differences can be explained by considering the behaviour of the application. During long phases of use no traffic is sent, and after a level is restarted, a new advertisement has to be obtained,

Application	Interarrival Time (s)		Bandwidth ($\mathrm{kbit\,s^{-1}}$)	
	Mean	Standard Deviation	Mean	Standard Deviation
Angry Birds	0.66	15.90	4.42	4.50
Aupeo	0.06	3.06	129.76	482.63
Twitter	8.91	44.09	0.27	0.04
Skype	0.55	1.95	1.30	1.84

Table 2.4: Mean and Standard Deviation of Interarrival Time and Bandwidth for Considered Applications

causing the transmission of data.

Next, we study the behaviour of the Aupeo application. We see that the application generates packets with relatively small interarrival times. This finding is backed up by the small mean interarrival time of $0.06\,\mathrm{s}$. The high standard deviation of $3.06\,\mathrm{s}$ is caused by the wait between two tracks. Furthermore, we see a high mean bandwidth of $129.76\,\mathrm{kbit\,s^{-1}}$, and a standard deviation of $482.63\,\mathrm{kbit\,s^{-1}}$. This is caused by the difference in traffic activity between times when tracks are either downloaded or not.

For Twitter we see that $90\,\%$ of all transmissions occur with an interarrival time of $1\,\mathrm{s}$. Also, we can observe a high mean interarrival time of $8.91\,\mathrm{s}$ and a high standard deviation of $44.49\,\mathrm{s}$. Additionally, the mean bandwidth is low with only $0.27\,\mathrm{kbit\,s^{-1}}$ and a low standard deviation of $0.04\,\mathrm{kbit\,s^{-1}}$ due to the fact that Twitter text messages are only 140 characters in length and thus only a low volume of traffic needs to be transmitted.

Finally, we consider the Skype application. Similar to the Twitter application, we see that $90\,\%$ of all packets occur with an interarrival time of less than $1\,\mathrm{s}$. However, in contrast to Twitter, we see a low mean interarrival time of $0.55\,\mathrm{s}$ with a standard deviation of $1.95\,\mathrm{s}$. Further, we observe a relatively low mean bandwidth of $1.30\,\mathrm{kbit\,s^{-1}}$ and a standard deviation of $1.8\,\mathrm{kbit\,s^{-1}}$.

To further study the traffic patterns of the applications, we study the autocorrelation of the packet interarrival time with regard to the lag length in Figure 2.4.

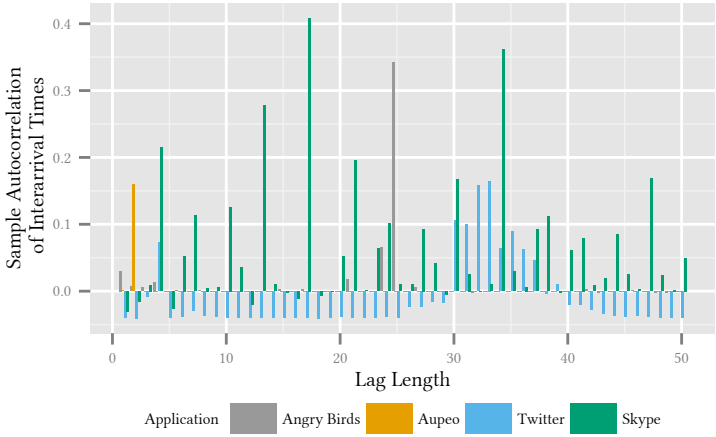


Figure 2.4: Autocorrelation of Interarrival Times for Considered Applications

We note that all studied applications present completely different autocorrelations for the interarrival times. This is one of the reasons that the applications under consideration will display different signalling behaviour in the next section.

Influence of Application Characteristics on Optimisation with Network Timers

Three State Model: Signalling Frequency vs. Power Consumption First, we investigate the signalling frequency generated by the studied applications for the three state network model. Figure 2.5 shows the signalling frequency with regard to the T_{DCH} timer. For all studies of the three state model, the T_{DCH} timeout is set to $T_{\text{DCH}} = 2 \cdot T_{\text{DCH}}$, a realistic value as shown in [23]. We see, that for T_{DCH} timers shorter than 6 s the Skype application in `idle` mode generates the highest signalling message frequency. The Angry Birds application generates

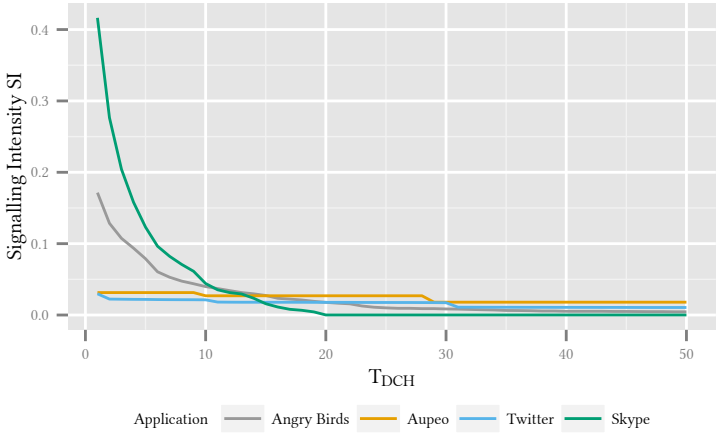


Figure 2.5: Signalling Message Frequency for Varying T_{DCH} Timers for the Three State Model

the second highest frequency of signalling messages, followed by the Aupeo application. The Twitter application generates the smallest signalling load. If the T_{DCH} value is longer than 15 s, this order changes. However, in general the signalling message frequency for higher T_{DCH} timeouts is lower than for shorter T_{DCH} timeouts. Now, the Aupeo application has the highest signalling frequency, followed by the Twitter application. The signalling message frequency for the Angry Birds application takes the third place. The application which generated the highest signalling message frequency generates the lowest frequency for higher timeout values. This behaviour can be explained by the fact that the Skype application sends keep-alive messages with an interval of less than 20 s. If the timer is greater than the interval time of the keep-alive messages, the UE stays always connected and thus generates almost no signalling.

These results show that the traffic patterns of the application have a large influence on the generated signalling load. Signalling is generated for every pause

in sending or receiving larger than the configured timeouts. If such pauses occur frequently, this increases the signalling load as shown on the examples of Skype and Angry Birds. Applications with more time between the sending or receiving of data cause less signalling, as shown by Aupeo and Twitter. Furthermore, we can observe that the signalling load can be reduced by increasing the CELL_DCH timeout, with the minimum being reached as T_{DCH} approaches infinity. From a signalling load perspective, a value of 20 s would probably be sufficient, however if other metrics such as radio resource consumption are considered 10 s would be acceptable for a network operator.

Based on this finding, we see that increasing the T_{DCH} timer decreases the signalling frequency at the RNC. However, the actual signalling frequency depends on the application running at the UE. From a network operator's point of view, the three state model should always be preferred to the two state model because it generates less signalling messages per second, thus decreasing the load at the RNC. This view does however not consider the additional radio resources which are kept in use for a longer time if larger T_{DCH} values are used. Additionally, it should be noted that the choice of the network model is sometimes outside of the domain of the network operator. Proprietary Fast Dormancy algorithms, as the considered two state model, are enabled on the UE by the user.

In Figure 2.6 we consider the power consumption if the network uses the three state model, i.e. if the Fast Dormancy mode of the UE is disabled. The figure shows the mean power consumption of the device with regard to the T_{DCH} timeout. Possible values range between 0 mW if the UE was in `idle` state during the whole measurement and 800 mW if the UE was in CELL_DCH state during the complete measurement. We see, that the least power over all considered T_{DCH} values is consumed by the Twitter application. The second least power consumption is required by Aupeo, followed by Angry Birds. Finally, the most power is consumed by Skype. Here we see, that the maximum value of 800 mW is reached at a T_{DCH} timeout of 20 s. This is because, due to the periodic traffic behaviour of Skype, the device is always in CELL_DCH state. Again, we see that the traffic characteristics of the applications impact the power con-

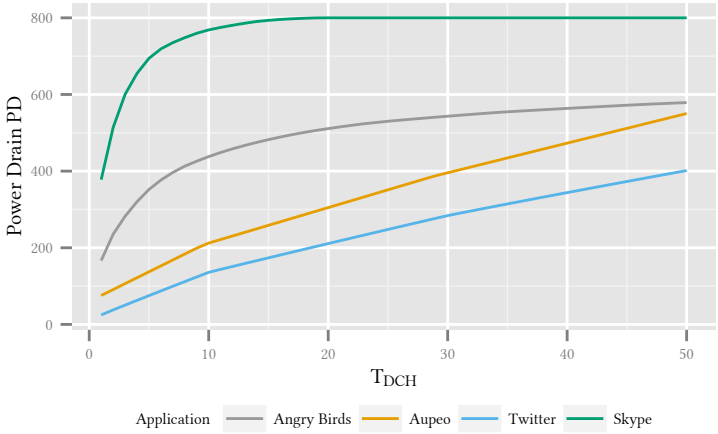


Figure 2.6: Power Drain for Varying T_{DCH} Timers for the Three State Model

sumption. Applications with more network activity are forced to stay in more power consuming states for a longer time. We see that for very small network timers, the power consumption is minimal. However, as seen in the last section small timers increase the signalling load at the RNC. Again, a choice of 10 s for the T_{DCH} timer can be seen as a compromise between signalling load and power consumption.

Finally, we aggregate both metrics in in Figure 2.7. The X-axis of the figure gives the signalling message frequency. On the Y-axis we show the power consumption. Different T_{DCH} values are shown by different colors as specified by the colorbar. First we consider Angry Birds. We observe that as the signalling frequency approaches zero, the power consumption rapidly increases, even if only small gains in signalling frequency reduction can be achieved. The Aupeo application presents a completely different picture. Here, we can see multiple almost horizontal lines of markers. If T_{DCH} is chosen in this range, each increase of T_{DCH} brings a small decrease in signalling frequency for a increase in power

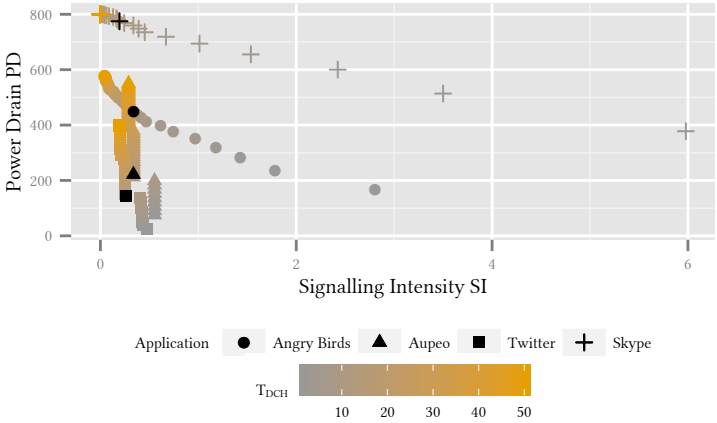


Figure 2.7: Influence of Manipulating T_{DCH} Timer on Signalling Intensity and Power Consumption for the Three State Model. Filled Marker Highlights $T_{DCH} = 11\,\mathrm{s}$

consumption. However, some points of discontinuity exist. If for example the CELL_DCH timer is increased from 10 s to 11 s, a decrease in power consumption of 40 % can be achieved by only suffering from a small increase of power consumption. These points of discontinuity would present themself to be suitable targets of optimisation. Next, we consider the Twitter application. It displays a similar behaviour as the Aupeo application, with multiple points of discontinuity. Note that Twitter exhibits a different point of discontinuity, and the T_{DCH} value of 10 s, which provided good results for Aupeo is not optimal for Twitter. Finally, Skype shows a completely different picture. First, note that due to the large signalling frequency of Skype for small values of T_{DCH}, $T_{\text{DCH}} = 1\,\text{s}$ is not displayed in the figure. Furthermore, as the T_{DCH} timer increases above 20 s the signalling frequency does not decrease any further, and the power consumption remains at the maximum value. We observe, that there is no common optimal value for all applications which would result in an acceptable trade off.

Two State Model: Signalling Frequency vs. Power Consumption Now, we study the consequences of the application traffic in a network using the two state model. The two state model occurs in reality if Fast Dormancy implementations are considered. Here, the UE disconnects from the network if for a certain time no traffic is sent or received in order to reduce power consumption. As for the three state model, Figure 2.8 shows the signalling frequency with regard to the setting of the T_{DCH} timer. We see the same general behaviour as with the three state model, however the signalling frequency generated by each of the applications for the two state model are usually higher. For example, even for relatively high T_{DCH} timeout values of 10 s, the Angry Birds application causes 270 % of the signalling frequency as in a network using the three state model.

Next, we consider the changes in the power consumption of the UE if the user decides to enable Fast Dormancy, i.e. switch to a two state model, in Figure 2.9. As with the signalling frequency, we only see a quantitative differences to the three state model. Again, we compare the differences between two state model and three state model on the example of the Angry Birds application. For

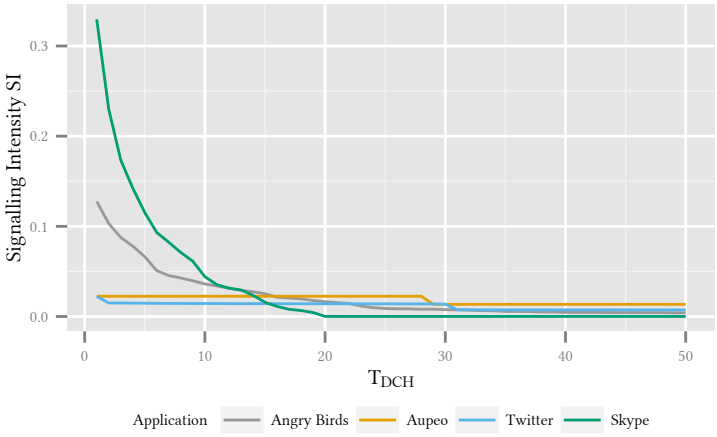


Figure 2.8: Signalling Message Frequency for Varying T_{DCH} Timers for the Two State Model

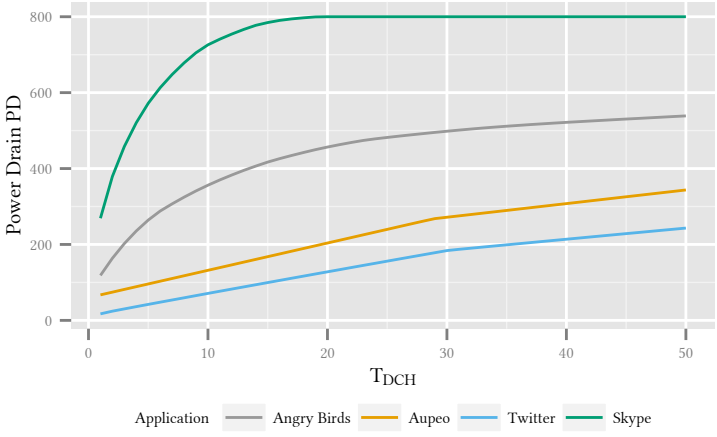
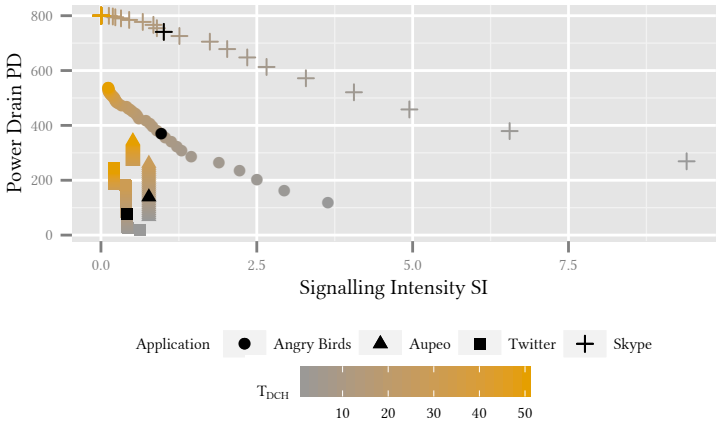


Figure 2.9: Power Drain for Varying T_{DCH} Timers for the Two State Model



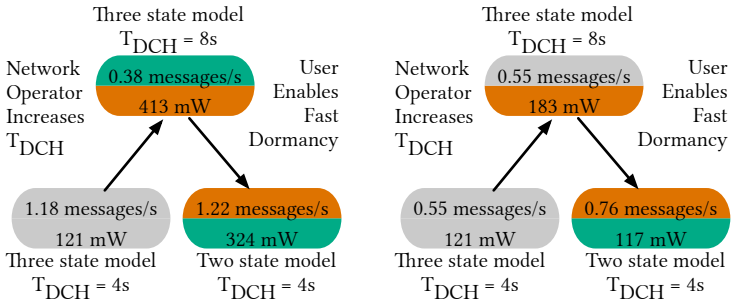
(a) Influence of Manipulating T_{DCH} Timer on Signalling Intensity and Power Consumption for the Two State Model. Filled Marker Highlights $T_{DCH} = 11\,\mathrm{s}$

the same considered T_{DCH} timeout of 10 s, we see a decrease of 81 % in power consumption when compared with the three state model.

Finally, we compare the influence of changes of thr T_{DCH} timeout on both signalling frequency and power consumption for the two state model in Figure 2.10a. As for the three state model, we see that there is no trade-off between power consumption and signalling frequency that would be acceptable for all application. Even for single applications T_{DCH} values such as 10 s which was an acceptable trade off Angry Birds is no longer a good choice in the two state model.

Consequences of Trade-Off: Signalling Frequency vs. Power Consumption

To illustrate the ramifications of the behaviour discussed in the previous section, we compare the influence of the T_{DCH} timer on an application with different traf-



(a) Angry Birds *(b) Aupeo*

Figure 2.11: Influence of Manipulating T_{DCH} Timer on Different Applications

fic characteristics, for example the Aupeo application as shown in Figure 2.11b. The signalling load before the increase of the T_{DCH} timer was 0.55 messages per second, after the change to $T_{\mathrm{DCH}} = 8\,\mathrm{s}$ the load remains unchanged. Thus, the policy change based on one application brings no significant gain to other applications. However, from a user's point of view, the power consumption increased from $121\,\mathrm{mW}$ to $183\,\mathrm{mW}$. Again, we assume the user activates fast dormancy to deal with the increase in power consumption of more than 50%. This results in a decrease of power consumption to $117\,\mathrm{mW}$, and an increase of overall signalling frequency to 0.76 messages per second. By changing the value without considering all applications, the network operator has decreased the QoE for other users, and worsened his overall situation. Thus, due to the large number of applications it seems impossible to optimise the T_{DCH} timeout to reduce the signalling message frequency without negatively impacting the users QoE in unexpected ways.

There exist applications, like Twitter and Aupeo, where optimisation by modifying the T_{DCH} values can provide acceptable results. However, these optimisations are only successful if a single application or network model is considered. For other applications, like Angry Birds or Skype, this optimisation ap-

proach does not seem to be successful. A reduction of signalling load and power consumption is possible, if the application developers are incentivised to optimise their applications in these regards. In [23] the authors suggest methods to achieve this optimisation, for example batch transfer of advertisements for applications like Angry Birds or decreasing the refresh rate in applications like Skype. However, at the moment application developers are neither receiving incentives to optimise applications in this way, nor do hardware vendors provide interfaces to facilitate such optimisation. Such interfaces would allow application developers to schedule their data transmissions in such a way that both signalling and battery drain would be reduced. Additionally, these interfaces would need to allow the application developer to specify if sending the transmission is urgent because the application is being actively used by the user and requires the feedback of the transmission, or if the data is being sent as a regular update while the application is running in the background and can be scheduled for later transmission as suggested by [27, 28].

2.2.3 Impact of Network Configuration and Background Traffic on Web QoE

So far we have discussed only power consumption as a QoE influence factor. For applications like web browsing, one relevant QoE influence factor are page load times. Therefore, we consider a web QoE model which quantifies the impact of page load times on mean opinion scores [29]. We distinguish here between *web QoE* and *QoE*, as no QoE models are currently existing which consider page load times as well as power consumption. In this section, we study the impact of background traffic as well as network timer settings on the page load time of an image and the resulting Mean Opinion Score (MOS). For this study, we only consider the three state network model, but the results can be applied to the 2-state model as well.

We assume a scenario, where a user is running a background application like Twitter or Skype. Then, while the application is in the background, the user be-

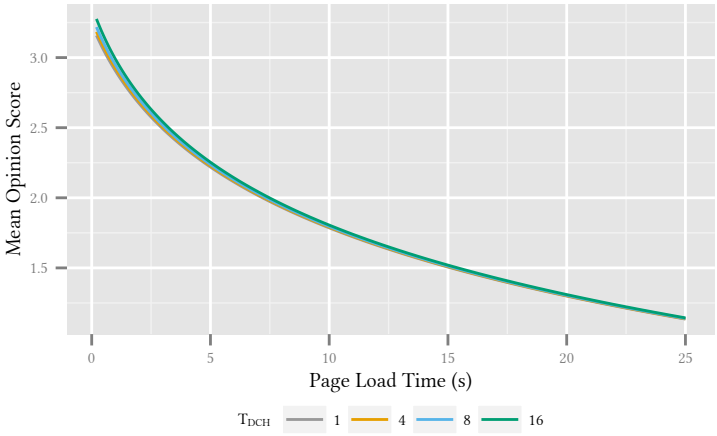
gins to download an image from a website. Due to the background traffic, and depending on the network model and associated timer values, the UE may be currently either in idle, CELL_FACH or CELL_DCH state. We give the probability of a random observer encountering the system in CELL_FACH state by $p_{\text{CELL\_FACH}}$ and the probability of a random observer encountering in idle state by $p_{\texttt{idle}}$. If the device is currently not in CELL_DCH state, it takes some time to connect. This promotion time depends on the current state and is according to [24] 2 s if the UE is in idle state and 1.5 s if the device is in CELL_FACH state. For this study, we assume that the user randomly chooses a time to begin downloading an image. The time until the image is displayed consists of the time to load the page t_p, as well as the time to go online t_o, where t_o is the mean time to go online, given as

$$t_o = p_{\texttt{idle}} \cdot 2.5\,\text{s} + p_{\text{CELL\_FACH}} \cdot 1.5\,\text{s}.$$

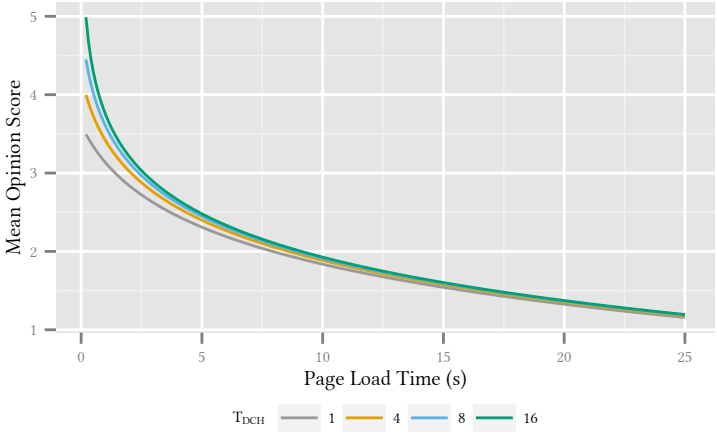
Thus, the total time t that is required to download the image is given by $t = t_o + t_p$.

The authors of [29] give a function to calculate the MOS based on the required page load time as $QoE(t) = a \cdot \ln t + b$, were a and b depend on the type of content being downloaded. For our scenario, picture download, values of $a = -0.8$ and $b = 3.77$ are suggested. It has to be noted that for different web sites, the logarithmic function was still observed, but different values for a and b were obtained as given in [29]. These values depend for example on the type of web page as well as the size of the content. Nevertheless, the results presented in this section are therefore generalizable for web browsing to various pages. This allows us to give an expected MOS for downloading pictures while a background application is influencing the probability of a device already being in CELL_DCH state or still having to be promoted to CELL_DCH state.

Using this methodology we study the influence of background traffic on the QoE for two background applications with different traffic characteristics. In Figure 2.12a we assume that the user is running the Twitter application as a



(a) Background Traffic Generated by Twitter



(b) Background Traffic Generated by Skype

Figure 2.12: Perceived QoE for Loading a Page with Existing Background Traffic

background process. The application is set to update the users status feed every 300 s. In Figure 2.12b the user is running the Skype application as a background application. This application sends keep alive messages every 20 s. For each application, we assume the three state network model with T_{DCH} settings of 1 s, 4 s, 8 s and 16 s. We always set $T_{\mathrm{DCH}} = 2 \cdot T_{\mathrm{DCH}}$. In both figures we show the assumed page load time, as provided by the network, on the X-axis for values from 0.2 s to 25 s. We assume 0.1 s as a lower bound because page load times lower than 0.1 s seconds are not distinguishable [30] by humans. The calculated MOS values are given on the Y-Axis.

The picture downloads with the background traffic generated by the Twitter application result in MOS values beginning at 3.15 for $T_{\mathrm{DCH}} = 1$ s, 3.18 for $T_{\mathrm{DCH}} = 4$ s, 3.21 for $T_{\mathrm{DCH}} = 8$ s, and 3.27 for $T_{\mathrm{DCH}} = 16$ s respectively. With increasing page load time, the MOS again decreases. This behaviour is due to the fact that the Twitter application periodically sends traffic every 300 s. Then, no further activity occurs until the next refresh occurs. In this time, the UE transitions to `idle` state. This traffic characteristic causes a high probability of a user encountering the device in a `idle` state. In contrast, downloading pictures with the Skype application generating background traffic, causes different MOS values. For a page load time of 0.2 s the MOS value with $T_{\mathrm{DCH}} = 1$ s is 3.49 with $T_{\mathrm{DCH}} = 4$ s we get 3.99 for $T_{\mathrm{DCH}} = 8$ s we get a MOS value of 4.44 and finally for $T_{\mathrm{DCH}} = 16$ s we get 4.99 respectively. For increasing page load times, the MOS decreases. This increased MOS values occur because of the high frequency of traffic sent by the Skype application. Here, every 20 s traffic is sent. This means, that even for relatively low values of T_{DCH} the user has a high probability of encountering a state where no promotion delay is required before the actual page load time can begin.

From these studies we can conclude that, when considering QoE on mobile devices, not only the page load time caused by the network but also additional delays caused by the state of the device should be considered. As shown on two examples, this state can be affected by other applications which are running in the background and generate traffic.

2.3 A Performance Model for 3G RRC States

[14]

2.3.1 System Description and Basic Assumptions

We consider a UE which sends and receives a sequence of data packets via a 3G UMTS network. As discussed in Section 2.1.1, the arrival process of the packet transmissions determines the RRC states of the UE. However, the direction of packets, i.e. them originating from the UE or the NodeB, has no impact on the RRC states, as the states depend solely on traffic activity. Due to the high impact of RRC states on traffic patterns, we do not consider packet sizes in this model. In real UMTS networks very small packets might be treated differently for RRC states, but we neglect this both for simplicity reasons and because the impact of packet sizes is highly network operator specific [21]. Furthermore, RRC state transitions are complex procedures depending on implementation details of the UE, the specific UMTS release, and the configurations by the network operator. In order to keep our model simply, but realistic, we reduce the set of standardized RRC states and the state transition triggers in the following ways.

In a first step we consider only a basic scenario with two RRC states: `idle` and CELL_DCH as shown in Figure 2.2b. The UE switches to CELL_DCH to transmit or receive data and after an inactivity period of duration T_{DCH} it switches back to `idle`. The motivation for the two states RRC scenario is twofold. First, it serves for illustration purposes. We derive the model step-by-step in this simple scenario to explain the ideas behind the equations. Then, the ideas can be easily transferred to the more complex three states scenario in Section 2.3.1. Second, the scenario is of practical relevance since proprietary implementations of the fast dormancy concept [36] exist, where the UE decides to switch to `idle` shortly after the transmission of a packet without using any other RRC state. Furthermore, this model is very similar to the one found in LTE systems. In LTE, only a distinction between connected and disconnected states can be found, which maps to the `idle` and CELL_DCH states discussed in this model.

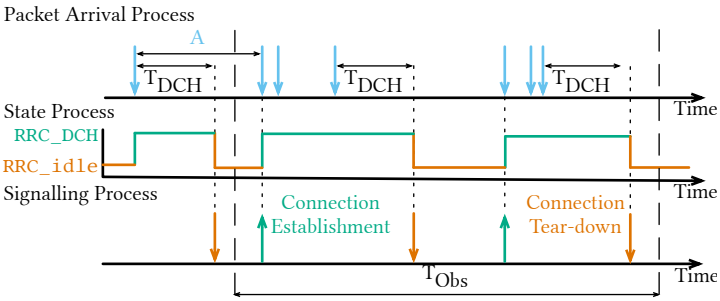


Figure 2.13: Relation of Packet Arival Process A, State Process, and Signaling Process in the Two State Scenario

In our model we aggregate both packets sent and received by the UE in in the packet arrival process, which is assumed to be a renewal process, i.e. a process with identical and independently distributed inter-arrival times, described by the random variable A as shown in Figure 2.2b. Thus, the probability that the time between two consecutive packets is at most t is $P(A \leq t) = A(t)$. We challenge this assumption by studying measurements of packet traces in Section 2.3.2.

The packet arrivals determine the RRC state of the UE and the corresponding transitions. Therefore, the packet arrival process can be seen as a modulating process, c.f. [31, 7], while the state and the signaling process represent modulated, i.e., resulting processes.

The Case of Two RRC States

State Distribution First, we are interested in the state distribution $P(S = s)$, i.e., the fraction of time the UE spends in state $s \in \{\texttt{idle}, \texttt{CELL\_DCH}\}$ for a given inter-packet time A. For this purpose, we define an observation interval T_{Obs}, depicted in Figure 2.13, which is assumed to be larger in orders of magnitude than the average packet inter-arrival time $E[A]$. In addition, we take the

position of an outside observer who observes the state S at a random point in time t^*, uniformly distributed within the observation interval. Then the state distribution $P(S = s)$ is the probability that the observer encounters the UE in state S at the time t^*.

We calculate this distribution as

$$P(S = s) = \int_0^\infty q(\tau) \cdot P(S = s)|A = \tau) d\tau, \tag{2.1}$$

where $q(\tau)$ is the probability density that t^* falls into an interval of length τ and $P(S = s)|A = \tau)$ is the probability that the UE is in state S under the condition that t^* is within an interval of length τ.

First, we derive $q(\tau)$. This probability density has to be proportional to $a(\tau)$ and to τ, where $a(\tau)$ is the probability density function of the random variable A. Therefore, we have that $q(\tau) = a(\tau) \cdot \tau \cdot c_0$ with the proportionality constant c_0. Due to $\int_0^\infty q(\tau) d\tau = 1$, we have $c_0 = 1/E[A]$, which leads to

$$q(\tau) = \frac{a(\tau) \cdot \tau}{E[A]}. \tag{2.2}$$

Next, we derive the conditional probability $P(S = s)|A = \tau)$ that t^* falls within a period with state S under the condition the that the inter-packet time is $A = \tau)$. We use the fact that t^* is uniformly distributed within τ and calculate the probability $P(S = \texttt{idle})$ by considering the relevant cases:

$$P(S = \texttt{idle}|A = \tau) = \begin{cases} 0, & \text{if } T_{\text{Obs}} \leq T_{\text{DCH}} \\ \frac{\tau - T_{\text{DCH}}}{\tau}, & \text{otherwise.} \end{cases} \tag{2.3}$$

Similarly, we obtain $P(S = \text{CELL\_DCH})$ as:

$$P(S = \texttt{idle} | A = \tau) = \begin{cases} 1, & \text{if } \tau \leq T_{\text{DCH}} \\ \frac{T_{\text{DCH}}}{\tau}, & \text{otherwise.} \end{cases} \tag{2.4}$$

Average Frequency of State Transitions Next, we estimate the average frequency of state transitions resulting from a given packet arrival process. For that purpose, we consider again the observation interval T_{Obs} and focus on the state transitions from `idle` to CELL_DCH since every switch from CELL_DCH to `idle` results in a switch vice-versa. The expected number of observed packets during T_{Obs} is $E[n_{\text{P}}] = T_{\text{Obs}}/E[A]$. Furthermore, the probability that time between two consecutive packets exceeds the timer T_{DCH} is

$$P(A > T_{\text{DCH}}) = 1 - P(A \leq T_{\text{DCH}}) = 1 - A(T_{\text{DCH}}). \tag{2.5}$$

The number of state transitions $n_{\texttt{idle}\rightarrow\text{CELL\_DCH}}$ during T_{Obs}) directly corresponds to the number of inter-packet times exceeding T_{DCH} since an active connection is torn down after an inactivity period of T_{DCH}. Thus, the expected number is

$$\begin{aligned} E[n_{\texttt{idle}\rightarrow\text{CELL\_DCH}}] &= E[n_{\text{P}}] \cdot P(A > T_{\text{DCH}}) \\ &= \frac{T_{\text{Obs}}}{E[A]} \cdot (1 - A(T_{\text{DCH}})). \end{aligned} \tag{2.6}$$

Hence, the expected frequency of state transitions is

$$E[n_{\texttt{idle}\rightarrow\text{CELL\_DCH}}] = \frac{1 - A(T_{\text{DCH}})}{E[A]}. \tag{2.7}$$

The same holds also for the state transitions from CELL_DCH to `idle` and hence $E[f_{\text{CELL\_DCH}\rightarrow\texttt{idle}}] = E[f_{\text{CELL\_DCH}\rightarrow\texttt{idle}}]$ holds.

Extension for Three RRC States

In this section, we consider three states: `idle`, CELL_DCH, and CELL_FACH. Again, we assume that the UE switches from `idle` to CELL_DCH whenever it transmits or receives data. After an inactivity of T_{DCH} the UE switches to CELL_FACH, and after an additional inactivity of T_{DCH}, it switches to `idle`, as depicted in Figure 2.2a. This scenario usually occurs when the network controls the RRC state of the UE without proprietary connection tear-down mechanisms implemented on the UE. In today's network some operator transition the UE to a state with a paging channel URA_PCH instead of the `idle`, but the resource consumptions in both states are very similar and we therefore neglect the URA_PCH state for simplicity reasons.

State Distribution The state distribution $P(S = s)$ for the three states $s \in \{\texttt{idle}, \text{CELL\_FACH}, \text{CELL\_DCH}\}$ can be derived in the same way as for the scenario with two states. Therefore, we present only the conditional probabilities, which differ from the two state case, and use Equation 2.1 for the calculation of the distribution. First, we consider $S = \texttt{idle}$:

$$P(S = \texttt{idle} | A = \tau) = \begin{cases} 0, & \text{if } \tau \leq T_{\text{DCH}} + T_{\text{DCH}} \\ \frac{\tau - (T_{\text{DCH}} + T_{\text{DCH}})}{\tau}, & \text{otherwise.} \end{cases} \tag{2.8}$$

For the case of $S = \text{CELL\_FACH}$, we have:

$$P(S = \text{CELL\_FACH} | A = \tau) = \begin{cases} 0, & \text{if } \tau \leq T_{\text{DCH}} \\ \frac{\tau - T_{\text{DCH}}}{\tau}, & \text{if } T_{\text{DCH}} < \tau \leq T_{\text{DCH}} + T_{\text{DCH}} \\ \frac{T_{\text{DCH}}}{\tau} & \text{if } \tau > T_{\text{DCH}} + T_{\text{DCH}} \end{cases} \tag{2.9}$$

The probability for the CELL_DCH state $P(S = \text{CELL\_DCH} | A = \tau)$ does not differ from the two state scenario, i.e. Equation 2.4.

Average Frequency of State Transitions In contrast to the two state scenario, we have to consider a larger number of state transitions. These are the transitions from `idle` to CELL_DCH, from CELL_DCH to CELL_FACH, from CELL_FACH to CELL_DCH, and from CELL_FACH to `idle`. Other transitions do not occur. We first calculate the frequency of state transitions from CELL_DCH to CELL_FACH. This transition happens every time when the inter-packet time A exceeds the timer T_{DCH}. Therefore, the derivation is the same as presented above:

$$E[f_{\text{CELL\_DCH}\to\text{CELL\_FACH}}] = \frac{1 - A(T_{\text{DCH}})}{E[A]}. \tag{2.10}$$

$$E[f_{\text{CELL\_FACH}\to\texttt{idle}}] = \frac{1 - A(T_{\text{DCH}} + T_{\text{DCH}})}{E[A]}. \tag{2.11}$$

Furthermore, all state transitions from CELL_FACH to `idle` correspond to a switch from `idle` to CELL_DCH and therefore $E[f_{\texttt{idle}\to\text{CELL\_DCH}}] = E[f_{\text{CELL\_FACH}\to\texttt{idle}}]$. Finally, we calculate $E[f_{\text{CELL\_FACH}\to\text{CELL\_DCH}}]$. These state transitions occur, if $T_{\text{DCH}} < A \leq T_{\text{DCH}} + T_{\text{DCH}}$. Therefore, we have

$$E[f_{\text{CELL\_FACH}\to\text{CELL\_DCH}}] = \frac{A(T_{\text{DCH}} + T_{\text{DCH}}) - A(T_{\text{DCH}})}{E[A]}. \tag{2.12}$$

Other state transitions do not occur in our scenario, as shown in Figure 2.2a).

Modeling Signaling Intensity and Power Drain of the UE

We assume that every state transition involves signaling traffic. In order to quantify signaling load on an abstract level, we define the Signalling Intensity (SI) of an application, i.e. of a given distribution for A, as the average number of state transitions required for the transmission of a single data packet.

$$SI = \frac{E[f_{ST}] \cdot T_{\text{Obs}}}{E[n_{\text{P}}]} = E[f_{ST}]E[f_{ST}] \cdot E[A] \tag{2.13}$$

where $E[f_{ST}]$ is the sum of all state transitions. Consequently, $SI \in]0, 2]$ for the two state scenario since every packet can at most cause two state transitions, in the three state scenario $SI \in]0, 3]$ holds. This metric is intended to quantify the relation between transmitted data packets and the involved RRC state transitions, which all incur mobile network signaling. The metric can be extended to capture more details, such as the number and type of signaling messages exchanged for a specific state transition. Since we will use this metric for more qualitative analysis of source traffic produced by smart-phone applications, we stick to the definition above allowing for an illustrative understanding of the numerical results.

Next, we model the Power Drain (PD) of the UE due to the UMTS transmission unit. We assume three power levels $)PD_S$, one for every state s and calculate the average power drain PD based on the state distribution, which in turn depends on the packet arrival process A. We obtain

$$PD = \sum_{s \in S} PD_s \cdot P(S = s) \tag{2.14}$$

with $S = \{\texttt{idle}, \text{CELL\_DCH}\}$ or $S = \{\texttt{idle}, \text{CELL\_FACH}, \text{CELL\_DCH}\}$ depending on wether the two state scenario or the three state scenario is considered. This is a user-centric metric and gives insights into how efficient the transmission process uses the battery.

2.3.2 Numerical Examples and Their Implications

Model Validations

In order to assess the applicability of our performance model, we first have to check whether real-world application traces can be modeled as renewal process, which was our main assumption for the model. We use the Lewis-Robinson-Test [39], which is a hypothesis test with null hypothesis H_0 that the tested process is a renewal process. To this end, we use exemplary the measurement, obtained

with the measurement testbed introduced in Section 2.2.1, for two different types of applications: *Twitter* and *K9-Mail*. According to this test, the null hypothesis cannot be rejected for both of our packet traces at a significance level of $95\,\%$. Although this assumption may not be true for all applications, our results show that at least the considered applications can be modeled as a renewal process.

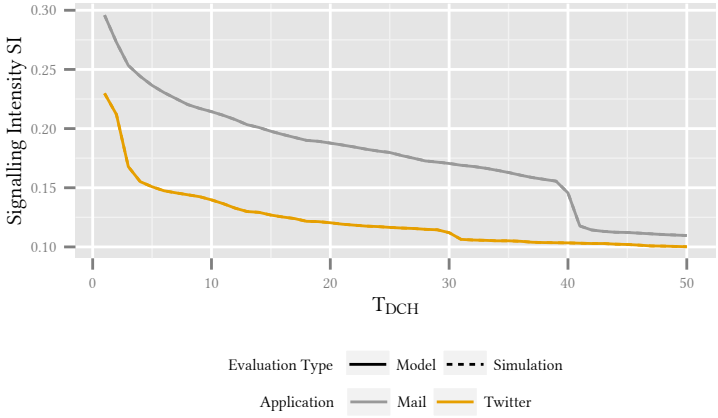
Next, we compare our analytical performance results with RRC protocol simulations using measured application and Transmission Control Protocol (TCP) traces which are described in more detail in Section 2.2.2. In order to produce analytical results that correspond to the real applications, we extract the empirical distributions of the inter-packet time A from the traces for both applications and use these distributions as input for Equation 2.2.

In Figure 2.14 we compare the accuracy of the results obtained by the presented method to the values obtained from simulations for the two measured applications and both considered metrics. We observe that the accuracy for both power drain PD and SI is very high. In Figure 2.14a the results for both the Mail and Twitter application obtained by the model completely align with the signalling intensity obtained by the simulation. The comparison of analytical results for the power drain to the simulation in Figure 2.14b leads to the same conclusions as for the signaling intensity.

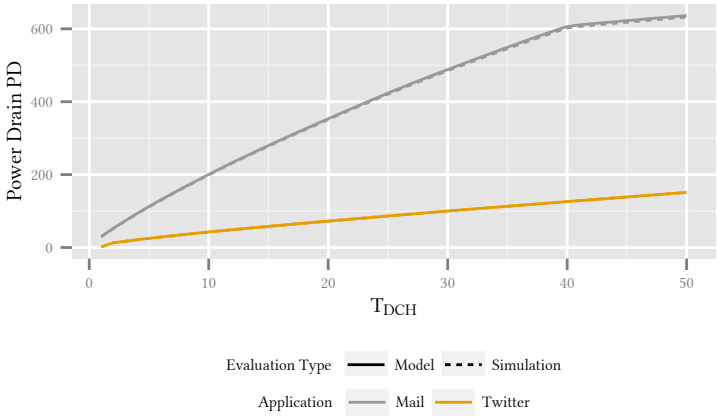
Impact of Traffic Patterns on Signaling Intensity

First, we focus on the signaling intensity SI of traffic patterns and check the impact of the average inter-packet time $E[A]$ and the timer configuration. The signaling intensity SI, i.e., the average number of state transitions required for the transmission of a single packet, is an abstract measure for the signaling load produced by a specific traffic pattern.

Impact of the Average Inter-Packet Time $E[A]$ Some applications, for example those downloading or streaming of videos, send and receive large amounts of data within short time frames. In contrast, other applications such as



(a) Signalling Intensity



(b) Power Drain

Figure 2.14: Comparison of the Performance Model with 3G Simulation for the Three State Scenario

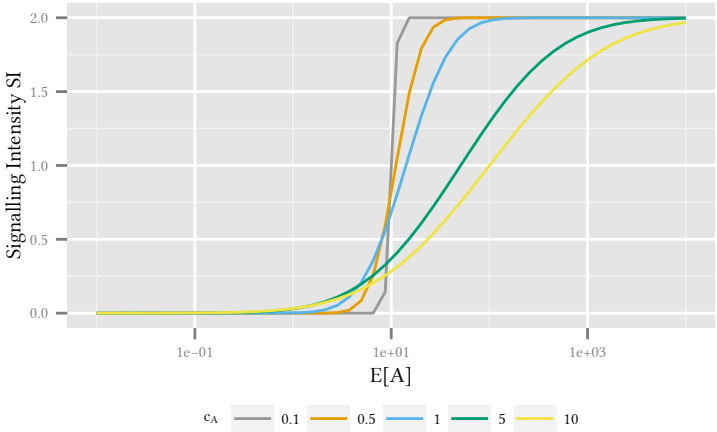


Figure 2.15: Signaling intensity for different traffic patterns considering the two state model with $T_{DCH} = 10\,\mathrm{s}$ for different traffic patterns

social network clients send and receive small amounts of data every few minutes over the time span of some hours or days.

In this section we study the impact of average inter-packet times $E[A]$ and the burstiness of the traffic pattern, i.e., the coefficient of variation

$$c_A = \frac{\sqrt{\mathrm{Var}[A]}}{\mathrm{E}[A]}$$

on the signaling load. For that purpose, we use the simple two state scenario, set the timer $T_{\mathrm{DCH}} = 10\,\mathrm{s}$, consider only the first and the second moment of the inter-packet time A, and assume that A follows a lognormal distribution, where both moments can be varied independently.

In Figure 2.15, we vary the average inter-packet time $E[A]$ in six orders of magnitude and investigate the resulting signaling intensity SI for different co-

efficients of variation c_A. We observe that c_A has no impact on SI for very small inter-packet times $E[A] < 1 \times 10^{-1}$ s. Here, the UE stays in state CELL_DCH for the complete time since no inter-packet times $A > T_{\text{DCH}}$ occur. In addition, the impact of c_A is small for very large values of $E[A] > 1 \times 10^{3}$ s. In this case, the UE switches to state CELL_DCH and back to state `idle` for the transmission of every packet. Therefore, the signaling intensity SI approaches the value 2. For values in between these two extremes, the coefficient of variation c_A has a considerable impact on the signaling intensity SI. More periodic traffic, i.e. smaller values of c_A, results in an increase of $SI$from 0 to 2 very sharp at the value $E[A] = T_{\text{DCH}}$, while this increase is more smooth for larger values of c_A. This is due to the fact, that for nearly periodic traffic it is crucial whether the timer value T_{DCH} is smaller or larger than $E[A]$. For larger values of c_A this dependency is weaker.

Impact of the Coefficient of Variation of the Inter-Packet Time c_A Next, we focus on the impact of the timer value T_{DCH} with respect to the burstiness of the traffic. We use the same setting as before, but fix the average inter-packet time $E[A] = 4$ s. While there are differences in $E[A]$ among users in real world settings, measurement studies have revealed that across all users $95\,\%$ of the packets are received or transmitted within 4.5 s of the previous packet [32]. Therefore, the order of magnitude of $E[A] = 4$ s is of practical relevance.

The signaling intensity SI is shown in Figure 2.16 with respect to the timer value T_{DCH} and the burstiness c_A of the traffic pattern. Obviously, larger timers lead to less frequent state transitions and therefore to less signaling load. We observe in addition that the impact of the timer is crucial for nearly periodic traffic. If the average inter-packet time for nearly periodic traffic is larger than the timer, then every packet transmission involves a state transitions from `idle` to CELL_DCH and a transition back. In contrast, no transitions are required if the average inter-packet time is shorter than the timer. With increasing values of c_A the impact of the timer is reduced. This means that for bursty traffic patterns the timer value is of less importance with respect to the generated signaling load.

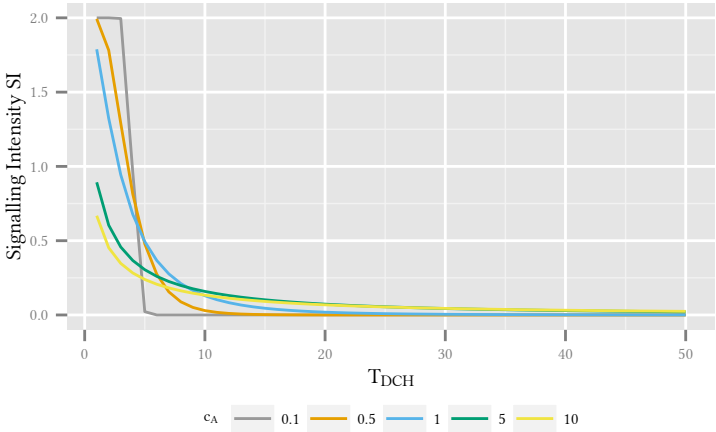


Figure 2.16: Signaling intensity SI for the two state model w.r.t. different timeout values T_{DCH} and coefficient of variations c_A.

Impact of Traffic Patterns on Power Drain of the UE

In this section we study the impact of the traffic patterns on the power drain PD of the UE. This metric quantifies how resource-efficient specific traffic patterns and timer configurations are for the battery of the UE.

For the power drain in the different RRC states, we use the radio network power measured in a commercial UMTS network [21]: $PD_{\text{CELL\_DCH}} = 800\,\text{mW}$, $PD_{\text{CELL\_FACH}} = 460\,\text{mW}$, and $PD_{\texttt{idle}} = 0\,\text{mW}$. We investigate the impact of the average inter-packet time, the impact of the timer configuration and validate our model with simulations. In Section 2.3.2 we have seen that no state transitions occur for very small and very large average inter-packet times $E[A]$. This was due to the fact, that for very small values the UE is continuously in state `idle` and for large values it switches to state CELL_DCH for every packet. Thus, traffic patterns with very small and very large inter-packet times $E[A]$ have also no impact on the power drain of the UE regardless of the burstyness represented

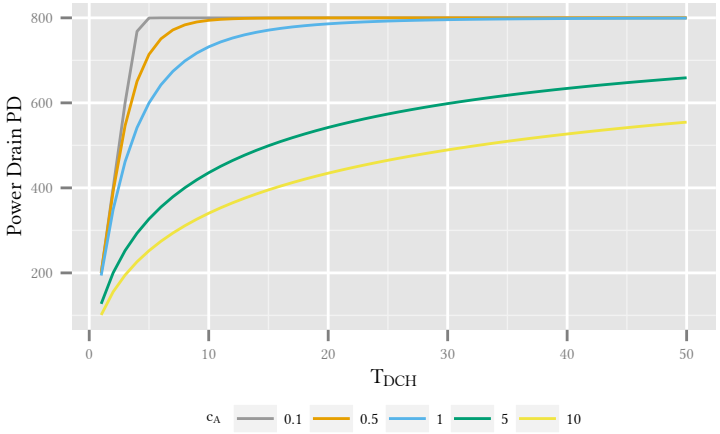


Figure 2.17: Power Drain PD for the two state model w.r.t. different timeout values T_{DCH} and coefficient of variations c_A.

by the coefficient of variation c_A.

To study the impact of the timer configuration T_{DCH}, we use the same setting as for the signaling load: lognormal distribution of inter-packet time A, $E[A] = 4\,\text{s}$ in the two state scenario. The numerical values shown in Figure 2.17 indicate that longer timeouts lead to a higher power drain PD. This is reasonable since the UE stays longer in the power intensive CELL_DCH state in these cases. However, we observe that the burstiness of the traffic pattern has also a considerable impact on the power drain PD. For example, for $T_{\text{DCH}} = 15\,\text{s}$, the power drain is only $400\,\text{mW}$ for very bursty traffic with $c_A = 10$, while it is almost $400\,\text{mW}$ for less bursty traffic with a $c_A = 1$. The reason is that bursty traffic patterns send a lot of traffic during short periods when the UE is in state CELL_DCH anyway. During the following off-periods that UE can save energy in `idle` state. Hence, we conclude that longer timeouts and smaller coefficients of variation $c_A = 1$, i.e. more periodic and less bursty traffic, result in a higher

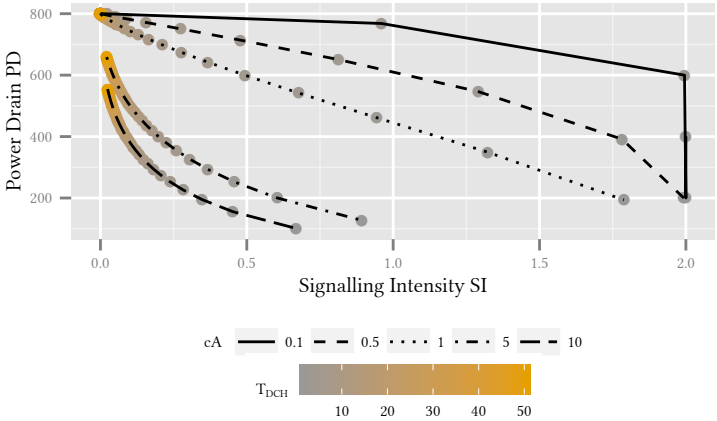


Figure 2.18: Trade-off between PD and SI for the two state model

power drain of the UE.

Trade-off: Energy Consumption vs. Signalling Load

In Figure 2.18, we show the effect of network parameter optimization using the timer T_{DCH} on traffic patterns with varying coefficient of variation.

We see that optimizations may decrease signaling by large amounts while only having very little impact on power consumption for one specific kind of traffic. The same timer setting could increase the power consumption for another kind of traffic while only offering little benefit with regard to the generated signaling intensity.

2.4 Lessons Learned

3 Application

3.1 Background and Related Work

3.1.1 Video Streaming Mechanisms

3.1.2 Video Quality of Experience

3.2 Cloud File Synchronization Services

[10]

3.2.1 System Model

Use Case: Photo Uploading

Cloud Storage Model and Performance Metrics

Scheduling Algorithms

3.2.2 Measurement

Bandwidth and Preparation Times

Image File Sizes

3.2.3 Numerical Evaluation

Waiting Time

Relative Disconnection Time

Connection Count

Mechanism Comparison

Trade-off Analysis

3.3 Dimensioning Video Buffer for Specific User Profiles and Behavior

[9]

3.3.1 System Model

$M/M/1$ Queue with pq-Policy

Mean Value Analysis of Steady State

Mean Value Analysis of Finite Videos and User Aborts

3.3.2 YouTube QoE Model

Stalling QoE Model

Initial Delay QoE Model

Combined QoE Model

3.3.3 QoE Study for Typical User Scenarios

Watch later Scenario

Default Video Streaming Scenario

Video Browsing Scenario

3.4 Trade-Offs for Multiple Stakeholders in LTE

[13]

3.4.1 System Model

Model Assumptions

Video Traffic Model

LTE Network Model

Evaluation Metrics for Smartphone Energy Consumption, Wasted Traffic, and Connection Count

3.4.2 Numerical Evaluation

Energy Consumption

Wasted Traffic

Connection Count

3.4.3 Trade-Offs

Transmission Mechanism Selection

Influence of Buffer Threshold Selection

3.5 Lessons Learned

4 Cloud

4.1 Related Work

4.1.1 Energy Efficiency in Data Centers

4.1.2 Network Function Virtualization

4.1.3 Crowdsourcing

4.2 Data Centers

[15]

4.2.1 Problem Formulation

4.2.2 Modeling

Default Data Center

Energy Efficient Data Center

4.2.3 Closed Form Solution

4.2.4 Performance Evaluation

4.3 Virtualized Network Functions

[33]

4.3.1 Model

Traditional GGSN

GGSN using Network Function Virtualization

4.3.2 Measurement Data

Dataset Description

Statistical Evaluation

4.3.3 Performance Evaluation

Traditional GGSN

Virtual GGSN

Impact of Startup and Shutdown Times

4.4 Crowdsourcing Platforms

[**Schwartz2015**]

4.4.1 Model

Analytical Consideration

Simulation

Validation

4.4.2 Measurements

Deriving Realistic Model Parameters

Comparison of Detailed and Analytical Model

4.4.3 Performance Evaluation

Impact of Campaign Interarrival Distributions

Trade-off Considerations for Platform Operators

4.5 Lessons Learned

5 Conclusion

Glossary

T_{DCH} Activity timer. UE is demoted from CELL_DCH state if no packets are sent or received for T_{DCH} seconds. 5, 6, 9, 10, 15, 17–25, 29, 30, 32–35, 39–43

T_{DCH} Activity timer. UE is demoted from CELL_FACH state if no packets are sent or received for T_{FACH} seconds. 5, 6, 9–11, 17, 29, 34, 35

HTTPS HTTP Over TLS. 14

LTE Long Term Evolution. Fourth generation mobile communication standard developed by the 3GPP. 30

NodeB Base station in an UMTS mobile network. 3, 4, 30

VoIP Voice over IP. 14

Acronyms

CELL_DCH Dedicated channel. 4–12, 19, 22, 27, 30, 31, 33–36, 40–42, 55

CELL_FACH Forward access channel. 4–12, 27, 34–36, 41, 55

idle idle mode. 4–7, 9–12, 17, 19, 27, 29–36, 40–42

3G Third Generation. 3, 9, 30

3GPP 3rd Generation Partnership Project. 7, 9, 55

CDF Culmulative Density Function. 15

CN Core Network. 3

IP Internet Protocol. 9, 10

M2M Machine to Machine. 3

MOS Mean Opinion Score. 26, 27, 29

PD Power Drain. 36, 37, 41–43

PSTN Public Switched Telephone Network. 3

QoE Quality of Experience. i, 8, 14, 25–29

QoS Quality of Service. 4

RAN Radio Access Network. 3

RNC Radio Network Controller. 3, 5, 11, 12, 19, 20

RRC Radio Resource Control. i, 3–12, 30, 31, 34, 36, 37, 41

SI Signalling Intensity. 35–37, 39–41, 43

TCP Transmission Control Protocol. 37

UE User Equipment. 3–8, 10–12, 14, 18, 19, 22, 27, 29–32, 34–36, 40–43, 55

UMTS Universal Mobile Telecommunications System. i, 3, 7, 30, 36, 55

Bibliography and References

Bibliography of the Author

Book Chapters

[1] T. Zseby, T. Zinner, K. Tutschku, Y. Shavitt, P. Tran-Gia, C. Schwartz, A. Rafetseder, C. Henke, and C. Schmoll, "Multipath routing slice experiments in federated testbeds", in *Future Internet Assembly (FIA) Book Future Internet: Achievements and Promising Technology*, J. Domingue, A. Galis, A. Gavras, T. Zahariadis, D. Lambert, F. Cleary, P. Daras, S. Krco, H. Müller, and M.-S. Li, Eds., Springer, May 2011.

Journal Papers

[2] N. P. Ngoc, T. N. Huu, T. V. Quang, V. T. Hoang, H. T. Thu, P. Tran-Gia, and C. Schwartz, "A new power profiling method and power scaling mechanism for energy-aware netFPGA gigabit router", *Computer Networks*, vol. Special Issue: Green Communications, 2014.

[3] H.-T. Nguyen, N. P. Ngoc, T.-H. Truong, T. T. Ngoc, D. N. Minh, V. G. Nguyen, T.-H. Nguyen, T. N. Quynh, D. Hock, and C. Schwartz, "Modeling and experimenting combined smart sleep and power scaling algorithms in energy-aware data center networks.", *Simulation Modelling Practice and Theory*, vol. 39, Dec. 2013.

[4] C. Schwartz, T. Hossfeld, F. Lehrieder, and P. Tran-Gia, "Angry apps: the impact of network timer selection on power consumption, signalling load, and web qoe", *Journal of Computer Networks and Communications*, vol. 2013, Sep. 2013.

[5] D. Hock, M. Hartmann, C. Schwartz, and M. Menth, "Resilyzer: ein werkzeug zur analyse der ausfallsicherheit in paketvermittelten kommunikationsnetzen", *PIK - Praxis der Informationsverarbeitung und Kommunikation*, vol. 34, Aug. 2011.

[6] S. Ickin, K. Wac, M. Fiedler, L. Janowski, J. Hong, and A. Dey, "Factors influencing quality of experience of commonly used mobile applications", *IEEE Communications Magazine*, vol. 50, no. 4, pp. 48–56, Apr. 2012.

[7] P. Tran-Gia, "A class of renewal interrupted poisson processes and applications to queueing systems", *Zeitschrift für Operations Research*, vol. 32, no. 3-4, pp. 231–250, 1988.

Conference Papers

[8] S. Gebert, C. Schwartz, T. Zinner, and P. Tran-Gia, "Continuously delivering your network (short paper)", in *IEEE/IFIP International Symposium on Integrated Network Management (IM)*, Ottawa, Canada, May 2015.

[9] T. Hoßfeld, C. Moldovan, and C. Schwartz, "To each according to his needs: Dimensioning video buffer for specific user profiles and behavior", in *IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan)*, May 2015.

[10] C. Schwartz, M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Performance model for waiting times in cloud file synchronization services", in *26th International Teletraffic Congress (ITC)*, Karlskrona, Sweden, Sep. 2014.

[11] M. Hirth, S. Scheuring, T. Hoßfeld, C. Schwartz, and P. Tran-Gia, "Predicting result quality in crowdsourcing using application layer monitoring", in *5th International Conference on Communications and Electronics (ICCE 2014)*, Da Nang, Vietnam, Jul. 2014.

[12] V. Burger, M. Hirth, C. Schwartz, and T. Hoßfeld, "Increasing the coverage of vantage points in distributed active network measurements by crowdsourcing", in *Measurement, Modelling and Evaluation of Computing Systems (MMB 2014)*, Bamberg, Germany, Mar. 2014.

[13] C. Schwartz, M. Scheib, T. Hoßfeld, P. Tran-Gia, and J. M. Gimenez-Guzman, "Trade-offs for video-providers in LTE networks: smartphone energy consumption vs wasted traffic", in *22nd International Teletraffic Congress Specialist Seminar on Energy Efficient and Green Networking*, Christchurch, New Zealand, Nov. 2013.

[14] C. Schwartz, F. Lehrieder, F. Wamser, T. Hoßfeld, and P. Tran-Gia, "Smartphone energy consumption vs. 3g signaling load: the influence of application traffic patterns", in *Tyrrhenian International Workshop 2013 on Digital Communications: Green ICT*, Genova, Italy, Sep. 2013.

[15] C. Schwartz, R. Pries, and P. Tran-Gia, "A queuing analysis of an energy-saving mechanism in data centers", in *International Conference on Information Networking (ICOIN)*, Bali, Indonesia, Feb. 2012.

[16] D. Hock, M. Hartmann, M. Menth, and C. Schwartz, "Optimizing unique shortest paths for resilient routing and fast reroute in IP-based networks", in *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, Osaka, Japan, Apr. 2010.

[17] D. Hock, M. Hartmann, C. Schwartz, and M. Menth, "Effectiveness of link cost optimization for IP rerouting and IP fast reroute", in *15th International GI/ITG Conference on Measurement, Modelling and Evaluation of Computing Systems and Dependability and Fault Tolerance (MMB & DFT 2010)*, Essen, Germany, Mar. 2010.

[18] D. Hock, M. Menth, M. Hartmann, C. Schwartz, and D. Stezenbach, "Resilyzer: a tool for resilience analysis in packet-switched communication networks", in *15th International GI/ITG Conference on Measurement, Modelling and Evaluation of Computing Systems and Dependability and Fault Tolerance (MMB & DFT 2010)*, Essen, Germany, Mar. 2010.

[19] C. Schwartz, J. Eisl, A. Halimi, A. Rafetseder, and K. Tutschku, "Interconnected content distribution in lte networks", in *GLOBECOM Workshops, 2010 IEEE*, IEEE, 2010, pp. 2028–2033.

[20] M. Hartmann, D. Hock, M. Menth, and C. Schwartz, "Objective functions for optimization of resilient and non-resilient IP routing", in *7th International Workshop on the Design of Reliable Communication Networks (DRCN)*, Washington, D.C., USA, Oct. 2009.

[21] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Characterizing radio resource allocation for 3g networks", in *Proceedings of the 10th Annual Conference on Internet Measurement (IMC)*, 2010.

[22] P. Perala, A. Barbuzzi, G. Boggia, and K. Pentikousis, "Theory and practice of RRC state transitions in UMTS networks", in *Proceedings of the Global Communication Conference (GLOBECOM) Workshops*, 2009.

[23] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Profiling resource usage for mobile applications: a cross-layer approach", in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2011.

[24] ——, "Top: tail optimization protocol for cellular radio resource allocation", in *Proceedings of the 18th IEEE International Conference on Network Protocols (ICNP)*, 2010.

[25] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: a measurement study and implications for network applications", in *Proceedings of the 9th Annual Conference on Internet Measurement (IMC)*, 2009.

[26] C. Lee, J. Yeh, and J. Chen, “Impact of inactivity timer on energy consumption in WCDMA and CDMA2000”, in *Proceedings of the Wireless Telecommunications Symposium (WTS)*, 2004.

[27] M. Calder and M. Marina, “Batch scheduling of recurrent applications for energy savings on mobile phones”, in *Proceedings of the Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, Boston, USA, Jun. 2010, pp. 1–3.

[28] E. Vergara and S. Nadjm-Tehrani, “Energy-aware cross-layer burst buffering for wireless communication”, in *Proceedings of the Conference on Future Energy Systems: Where Energy, Computing and Communication Meet*, Madrid, Spain, May 2012, p. 24.

[29] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, “Time is bandwidth? narrowing the gap between subjective time perception and quality of experience”, in *Proceedings of the International Conference on Communications (ICC)*, Ottawa, Canada, Jun. 2012, pp. 1325–1330.

[30] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, “Waiting times in quality of experience for web based services”, in *Proceedings of the Workshop on Quality of Multimedia Experience (QoMEX)*, Yarra Valley, Australia, Jul. 2012, pp. 86–96.

[31] P. Tran-Gia, “A renewal approximation for the generalized switched poisson process”, in *Proceedings of the International Workshop on Applied Mathematics and Perfomance/Reliability Models of Computer/Communication Systems*, 1983.

[32] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, “A first look at traffic on smartphones”, in *Proceedings of the 10th Annual Conference on Internet Measurement (IMC)*, 2010.

[33] F. Metzger, C. Schwartz, and T. Hoßfeld, “GTP-based load model and virtualization gain for a mobile network’s GGSN”, in *5th International Conference on Communications and Electronics*, Da Nang, Vietnam, Jul. 2014.

Software Demonstrations

[34] D. Hock, M. Hartmann, C. Schwartz, and M. Menth, *Resilyzer: ein werkzeug zur analyse der ausfallsicherheit in paketvermittelten kommunikationsnetzen*, Kiel, Germany, Mar. 2011. [Online]. Available: `http://www3.informatik.uni-wuerzburg.de/resilyzer`.

General References

[35] *Ts 25.331, Radio Resource Control (RRC); Protocol Specification*, 3GPP, 2012. [Online]. Available: `http://www.3gpp.org/ftp/Specs/html-info/25331.htm`.

[21] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Characterizing radio resource allocation for 3g networks", in *Proceedings of the 10th Annual Conference on Internet Measurement (IMC)*, 2010.

[22] P. Perala, A. Barbuzzi, G. Boggia, and K. Pentikousis, "Theory and practice of RRC state transitions in UMTS networks", in *Proceedings of the Global Communication Conference (GLOBECOM) Workshops*, 2009.

[36] NokiaSiemensNetworks, *White paper: understanding smartphone behavior in the network*, 2011.

[37] GSM Association and others, *Network efficiency task force fast dormancy best practices*, White Paper, May 2010.

[38] *Tr 22.801, study on non-mtc mobile data applications impacts (release 11)*, 2011.

[23] F. Qian, Z. Wang, A. Gerber, Z. Mao, S. Sen, and O. Spatscheck, "Profiling resource usage for mobile applications: a cross-layer approach", in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2011.

[24] ——, “Top: tail optimization protocol for cellular radio resource allocation”, in *Proceedings of the 18th IEEE International Conference on Network Protocols (ICNP)*, 2010.

[25] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, “Energy consumption in mobile phones: a measurement study and implications for network applications”, in *Proceedings of the 9th Annual Conference on Internet Measurement (IMC)*, 2009.

[26] C. Lee, J. Yeh, and J. Chen, “Impact of inactivity timer on energy consumption in WCDMA and CDMA2000”, in *Proceedings of the Wireless Telecommunications Symposium (WTS)*, 2004.

[6] S. Ickin, K. Wac, M. Fiedler, L. Janowski, J. Hong, and A. Dey, “Factors influencing quality of experience of commonly used mobile applications”, *IEEE Communications Magazine*, vol. 50, no. 4, pp. 48–56, Apr. 2012.

[27] M. Calder and M. Marina, “Batch scheduling of recurrent applications for energy savings on mobile phones”, in *Proceedings of the Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, Boston, USA, Jun. 2010, pp. 1–3.

[28] E. Vergara and S. Nadjm-Tehrani, “Energy-aware cross-layer burst buffering for wireless communication”, in *Proceedings of the Conference on Future Energy Systems: Where Energy, Computing and Communication Meet*, Madrid, Spain, May 2012, p. 24.

[29] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, “Time is bandwidth? narrowing the gap between subjective time perception and quality of experience”, in *Proceedings of the International Conference on Communications (ICC)*, Ottawa, Canada, Jun. 2012, pp. 1325–1330.

[30] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, “Waiting times in quality of experience for web based services”, in *Proceedings of the Workshop on Quality of Multimedia Experience (QoMEX)*, Yarra Valley, Australia, Jul. 2012, pp. 86–96.

[31] P. Tran-Gia, "A renewal approximation for the generalized switched poisson process", in *Proceedings of the International Workshop on Applied Mathematics and Perfomance/Reliability Models of Computer/Communication Systems*, 1983.

[7] ——, "A class of renewal interrupted poisson processes and applications to queueing systems", *Zeitschrift für Operations Research*, vol. 32, no. 3-4, pp. 231–250, 1988.

[39] H. Ascher and H. Feingold, *Repairable Systems Reliability: Modeling, Inference, Misconceptions and Their Causes*. Marcel Dekker, Inc., 1984.

[32] H. Falaki, D. Lymberopoulos, R. Mahajan, S. Kandula, and D. Estrin, "A first look at traffic on smartphones", in *Proceedings of the 10th Annual Conference on Internet Measurement (IMC)*, 2010.

ISSN xxxx-xxxx