Lehrstuhl Informatik VIII
Technische Universität München

Fault Analysis and Fault Tolerance of a Base Station System for Mobile Communications

Hubertus Kehl

Lehrstuhl Informatik VIII
Technische Universität München

Fault Analysis and Fault Tolerance of a Base Station System for Mobile Communications

Hubertus Kehl

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. J. Schlichter
Prüfer der Disseration:	1. Univ.-Prof. Dr. E. Jessen 2. Univ. Prof. Dr. U. Baumgarten

Die Disseration wurde am 27.12.1999 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 22.03.2000 angenommen.

Abstract

This thesis describes a new approach for fault analysis and the evaluation of fault tolerance (FT) schemes of telecommunication systems. Mobile base stations in a mobile communication network are considered as an example for such systems. A general view of the state of the art in FT with respect to FT for Mobile Radio Systems is given. Techniques for fault analysis and FT evaluation are summarized.

It will be shown that standard metrics for availability and reliability are in general not suitable for telecommunication systems, in particular not for mobile base stations. The major point against the use of standard metrics is that they don't take the telephone service itself into account. In particular, a new model is proposed that leads to three different metrics which measure reliability and availability from either the system or the user point of view. In order to evaluate these metrics a combination of analytical and simulation methods based on statistics of real measured fault rates is applied. The only necessary condition to adapt the method to other telecommunication systems is to have such statistics in form of logfiles about possible faults. Long surveillance periods and data from many systems increase the statistical significance. Standard statistic formulas can be used to estimate confidence intervals for these metrics.

On the one hand, the results of these metrics are used to discover the current weak spots of unreliability and unavailability and to improve them accordingly. Since the contribution of single components, boards, functional classes or classes of failures to unavailability and unreliability (as defined by the metrics) can be calculated, the determination of weak spots is simple. Improvement may be incorporated by two means: by the elimination of the faults or by introducing new fault tolerance strategies. Whereas the former alternative is very important it also has its limits, the latter method is more general, but leads to the problem of evaluating and selecting FT-strategies.

On the other hand, the approach also allows to evaluate new candidate redundancy schemes for fault tolerance improvement. Already used and new possible FT-schemes for the example system "Base Transceiver Site Equipment (BTSE)" have been categorized and described. Benefits of several new candidate fault tolerance schemes are calculated and predicted accurately, whereas up to now the benefits could only be guessed or estimated. This thesis clarifies the performance of the schemes and gives the basis for selecting them according to the benefit-effort ratio, this constitutes a vital method for system designers.

A base station is typically not comparable to a system in avionics or nuclear power plant environments, where ultra-high reliability and availability requirements must be fulfilled. In fact this wouldn’t even be possible due to the restrictions of the air interface. So the importance of FT in the mobile communication environment as well as various other problems like the general unreliability of the air-interface, the possible behaviour in the future and the general importance of the BTSE in the network will be discussed.

Acknowledgements

I very much appreciate my supervisor's Prof. Dr.-Ing. E. Jessen's encouragement during the project - also his ongoing support, the numerous scientific discussions guiding me into the right direction, further his mentoring on the organization, planning and execution of the project; last but not least I would like to thank him for spending so much time reading the thesis from the early stages of the project to the final release.

Next I want to thank the Siemens AG, especially Dr. J. Sokat and Dr. K. Fuchs for making this possible, sponsoring the project and for having the possibility to use all the computer equipment, the library, etc.
Special thanks go to Dr. K. Fuchs as an excellent coach. He provided me with his profound knowledge about Siemens' BTS, he offered his advice and spent much time for me.

I would also like to thank the wonderful team I was allowed to work with at Siemens for supporting me: Mr. S. Dillig, Mr. A. Köhn, Mr. Chr. Thomas, Mr. V. Franz. Especially Mr. F. Treml led helpful mathematical discussions, and Mr. K. Hong was engaged in programming the GUI of the alarm evaluation tool. Simultaneously a lot of thanks also go to all the other staff at Siemens who helpfully provided me with data material, information and knowledge.

I am very grateful to Prof. Dr. U. Baumgarten for his final remarks and advice.

Finally I want to thank my family, my friends and last but not least my partner Angela Schulz for her sophisticated and intensive encouragement.

Table of Contents

1 Introduction

Note: This introduction uses some terms out of the area of fault-tolerant computing. For an exact definition and introduction to these terms refer to chapter 2.

As today’s modern telecommunication systems play an important role in everyone’s life malfunctions of such systems can lead to significant economic impacts and even loss of lives, for example if emergency calls could not be placed or if the telecommunication system is used for controlling of certain machines, trains, industry plants etc.

Not surprisingly availability and reliability of such systems have become a key factor in the market.

Techniques for improvement of these factors have gained high importance and are of increasing interest (see Introduction of [SS82]). This thesis will describe such a technique for a class of systems for which the SIEMENS Base Transceiver Station Equipment (BTSE) is representative. The method will be useful for this class of systems that can be identified with the following characteristics:

- distributed real-time systems,
- consisting of mixed HW/SW components,
- undergoing changes in traffic conditions,
- the system has to operate in harsh environments and/or has to use unreliable transmission channels (here: air interface),
- typically such systems use redundancy or fault-tolerance schemes and may be able to handle traffic even when failures occur, so they show up a broad range of degraded system states from minor malfunction to totally inoperational (graceful degradation),
- the system is part of a permanent evolution, where new features will be added and redesigns of the system will happen in sequential development stages, where
- maintenance personnel is not present at the system site and maintenance is even very expensive, because of long driving distances to the system site, and downtimes are considerable,
- and where data about failures can be gathered from already installed systems,
- but redundancy in the network may be used to enhance the availability of the network to the end user (here: multiple cell coverage).

In the following the method will be briefly described:

At the first step a failure model (FM) for the complex BTSE system will be established.

As each model is always an abstraction of a real system and a perfect model, which represents the system in every aspect and in detail, is too expensive and not necessary (see [BFS83]), the system is modeled only so detailed that the intended analysis can be exerted. It will be shown that the standard definitions of availability and reliability are inappropriate for such systems, e.g. it is insufficient to model the system just as working and non-working and to use only Mean Time Between Failures (MTBF) values for components for a calculation. This is indeed not the

whole picture especially for software faults and specific (software-) recovery routines that react on detected errors. There are many types of faults that will result in different erroneous states. Therefore in our model a wrong behaviour of subsystem is only considered as a failure of that subsystem if it finally affects service.
New appropriate metrics have been defined to solve these problems.

A tool was developed that is able to calculate these metrics out of gathered data from real operational systems. Even the contributions of single components, boards, functional classes, classes of failures, probable causes, etc. to unavailability and unreliability (as defined by the metrics) can be calculated with this tool.

One of the major aims of the failure model is to make the weak spots evident - the major points for improvement. Improvement can be incorporated by two different strategies:
1. Eliminating the causes of the faults, find the software bugs, redesign some hardware
and/or
2. Introducing a new fault tolerance (FT) strategy that will be able to prevent a class of system failures, which may be induced by even new or not yet known faults.
Normally both strategies are used, but the first one has its limits. Even if much effort is spent no truly fault free system can be built. This forces the use of the second strategy, if the availability and reliability requirements are high.

With the failure model we are able to develop different FT schemes that cover the weak spots. Some of these schemes are already known, but up to now their performance could only roughly be estimated.
The models will give an estimation of what can be gained for a maximum, if such a scheme is incorporated (hopefully without introducing many new faults).
So the models help for the decision/selection of new fault tolerance techniques, which will be added as features in the evolution of such a system.
These schemes are now objectively evaluated by the use of metrics and those with the best benefit/effort ratio can finally be selected. The models form a solid basis for analysis and improvement.
System designers will be given an appropriate method of evaluating new fault tolerance schemes and ideas. The reliability and availability figures of the BTSE will finally be improved. These figures are key factors for some operators in the decision to use a SIEMENS BTSE, especially in security applications, e.g. GSM-Railway.

Other benefits of the approach are:

- It helps to understand the impact of faults in such a complex distributed system as the BTSE. This knowledge is important for diagnosis, maintenance and system test and also development.
- The models also permit to forecast the behaviour for future BTSE systems. Future systems need to handle more traffic; they may consist of similar (but more of the same) components. So with today's component failure rates an availability prediction of future systems is possible. These predictions can be used for evaluating the use of the FT schemes in the future.

- The weak spots may give hints about new FT schemes, which may be able reduce the existing problems.

As FT may exist already in the network by the use of multiple cell coverage the ultra-high availability of a single BTSE may not be as important. One metric was defined to estimate the network availability even if failures in a single BTSE happen. A discussion of these network aspects is important to estimate the necessity of FT improvement at BTSE level.

To conclude, the approach consists of:
1. Setting up the model
2. Applying the model to gathered data (calculating the weak spots)
3. Developing FT schemes for the weak spots
4. Evaluating the FT schemes with the model
5. Selecting FT schemes
and
6. Discussing the necessity of FT improvement

In the following paragraphs an overview on the structure and the topics of this thesis is given:

Chapter 2.1 introduces into the area of FT the terminology, models and various methods. In Chapter 2.2 a short explanation of the GSM network is given. Chapter 2.3 covers recent publications on fault tolerance in mobile communication systems, (as far as publications on this topic are made). In chapter 2.4 some basics out of the area of telephone traffic theory are described. A good review on the models of this thesis is given in chapter 3.2: Figure 3.1 shows how the various evaluations, metrics and discussion topics and results are interconnected and what results serve as input for further investigations. Refer to this picture for a general guide to this thesis.

In chapter 3 the failure model is presented. After the general introduction and discussion of the new model, the description of the BTSE hardware, software and behaviour is given. With this basics the equations for the new metrics PTT (chapter 3.4), PRP (chapter 3.5) and PURE (chapter 3.6) are derived. Chapter 4 describes the tool that was programmed to calculate these metrics. In the chapter 5 results of the calculations are presented. In chapter 5.1 the input data is described. Chapter 5.2 covers the validation of statistical significance. In chapter 5.3-5.5 the final results for the different metrics are provided.

In chapter 5.6 the current weak spots are listed. Chapter 6.1 lists possible and already used FT-schemes for mobile base station systems. In chapter 6.2 candidate redundancy schemes are evaluated by the use of the models. Chapter 6.3 shows how FT-schemes should be finally selected. A Discussion of various FT-aspects can be found in chapter 7 and the final conclusion is given in chapter 8. Aspects that should be researched in future work are described in chapter 8.3. A list of terms and abbreviations used throughout this dissertation is given in chapter 9. References on bibliography are collected in chapter 10.

2 State of the Art

2.1 State of the Art in FT

2.1.1 Faults, Errors and Failures

2.1.1.1 Terminology

In the area of FT many different definitions (or ways of usage) for the same terms can be found. As a clear and common vocabulary should be used, the definitions for faults, errors and failures are described in the following.

[Joh89] makes the following classification: failures are caused by errors, which are caused by faults.

This classification is characterized by the kind of universe, where they appear:

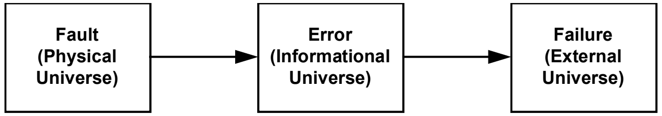

Fault (Physical Universe)
Error (Informational Universe)
Failure (External Universe)

Figure 2.1: Fault, Error and Failure [Joh89]

[Joh89]: The **Physical Universe** contains the semiconductor devices, ... , power supplies and all the other physical entities that make up the system. Beside the physical universe there are also other sources of faults like implementation, specification or operator mistakes (see Figure 2.4). The **informational universe** is where the error occurs. Errors affect units of information. The **external universe** is where the user of a system ultimately sees the effect of faults and errors (failures).

The definition of the following terms that mostly match our needs are derived from [Gib76] and [Lan86]:

Definition 2.1: Fault

A **fault** is a malfunction in a hardware, software or human component of the system, which may introduce errors and may lead to failures.

Definition 2.2: Error

An **error** is an invalid state of the system, such as incorrectly stored or transmitted items of data.

Definition 2.3: Failure

If no tolerance for specific faults exists, a fault may lead to a failure. A **failure** is a fault condition in the system or a functional unit (see Definition 2.4 and Definition 2.5) that has an influence on service. A **partial failure** degrades service, but does not interrupt it completely [Lan86]. A **total failure** interrupts the service until the system/functional unit is recovered automatically or until it is repaired.

Sometimes a fault leads directly to a failure, as in the case of a major hardware breakdown. So Figure 2.1 would need another arrow for these cases: directly from fault to failure, e.g. if the power supply has a fault, no change in system state (error) would be needed to produce a total failure.

2.1.1.2 Characteristics of Faults

Nelson and Carroll [Joh89] give the following overview of fault characteristics. They can be distinguished by cause, nature, duration, extent and value:

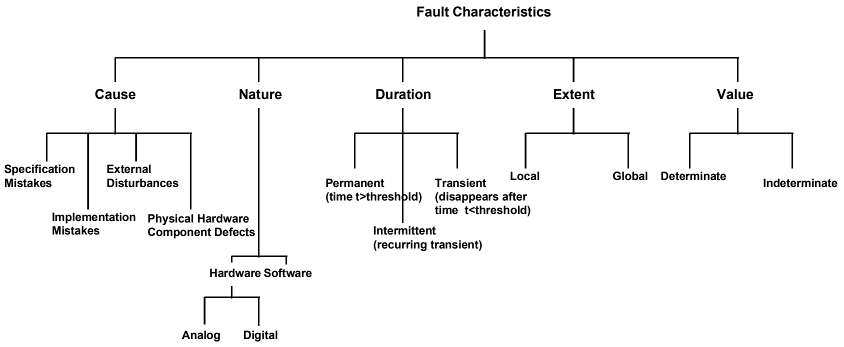


Figure 2.2: Fault Characteristics [derived from Joh89]

[AL81] adds another important separation of faults: on whether they can be foreseen in the design or not. So we have the two categories anticipated and unanticipated faults. If there are faults in the design, their effects will be unanticipated.

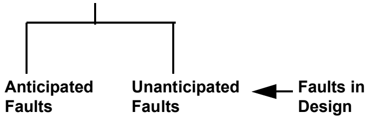


Figure 2.3: Anticipated and Unanticipated Faults

Anticipated faults will either be covered by some fault tolerance technique, or the consequences of the fault will be accepted (see [Ech90] below).

2.1.1.3 Barriers Constructed by Design Techniques

[Joh89] shows up the general barriers that are used in system design to cope with the fault -> error -> failure propagation.

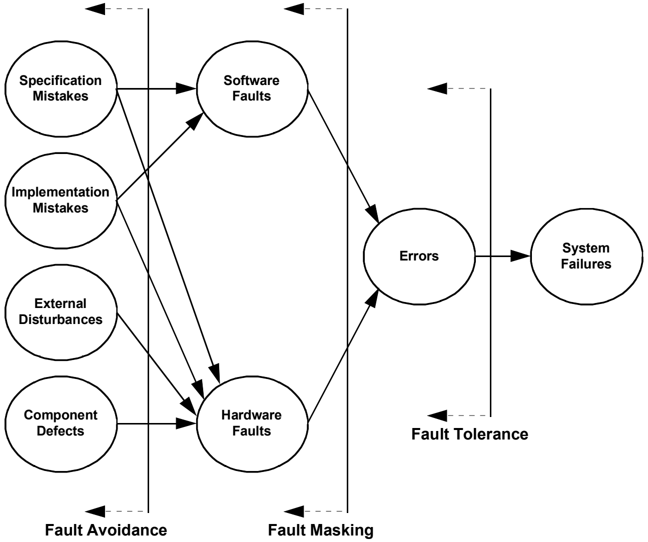


Figure 2.4: Barriers Constructed by Design Techniques [Joh89]

Special design techniques can be used to avoid faults, which is the first barrier and best barrier. It is clear that even if much effort is spent, faults cannot completely be avoided and no absolutely fault-free hard- and software system can be constructed. So the next barrier copes with faults but does not lead to errors because of fault masking. That means techniques like error correcting codes or voting are used to eliminate a manifestation of a fault before a propagation to the informational universe takes place (an error in the system), e.g. error correction in a memory before the data is delivered. Fault tolerance is the last barrier; its ultimate goal is preventing failures even if errors happen.

2.1.2 Phases in Fault Tolerance

[AL81] describes several phases in fault tolerance, which form the basis for fault tolerance techniques. They need not appear in the following sequential order. There also can be an interplay or mix of the phases, this tends to blur their identification in a real system:

Error Detection

- Normally faults cannot be detected directly, beside some cases where fault condition can be observed e.g. loss of power supply, overtemperature, etc.. The manifestation of the fault will generate errors somewhere in the system.
- The detection of an error does not necessarily serve to identify the fault, because the relationship between faults and errors is complex.
- The usual starting point is the detection of an erroneous state.

Damage Confinement and Assessment

- Delay between the manifestation of a fault and the detection of its erroneous consequences leads to invalid information spread again leading to other errors that have not (yet) been detected (error propagation).
- So it is necessary to assess the extent to which the system state has been damaged.

Error Recovery

- means to transform the current erroneous state into a well defined and error free state from which normal system operation can continue.

Fault Treatment and Continued Service

- Even if the system may have returned to an error-free state, techniques may be required to ensure that the same fault does not immediately recur.

2.1.2.1 Error Detection

[AL81] also categorizes the error detection phase in further detail, there are:

Replication checks

- Either an alternative implementation is checked against "normal" component
- or identical redundant components checking each other are used.
- Replication checks are extensive, powerful, general, simple, but associated with high costs.

Timing checks

- Timeouts reveal the presence of faults but not the absence.
- E.g. watchdog timers, supervision of time, still alive messages.

Reversal checks

- In this scheme input is calculated again of output and compared with the real input, or a fixed relationship between input and output exists, that can be used for checking.
- Only possible for a few special applications, where a reverse calculation or a fixed relationship between input and ouput exists.
  Simple, independent of design and implementation.

Coding checks

- All kinds of error correcting codes: e. g. parity, CRC, ...
- They are efficient, economical, but give only a limited form of acceptability check.

Reasonableness checks

- Based on knowledge of internal design and implementation, e.g. range checks, run-time checks of special programming languages, assert statements, ...
- Run-time checks vs. performance.

Structural checks

- Checking large data structures, by type information, redundant pointers (e.g. double linked lists) or counts.

Diagnostic checks:

- Exercising a component with inputs and checking against known outputs, this is checking the behaviour of components, rather than the system behaviour.
- Expensive in time and resources, therefore only as background job or periodically.
- Tests that are more stringent than normal operation (marginal tests)
- Diagnostic checks are good at identifying faults in deteriorated hardware.

2.1.2.2 *States of a System*

[Ech90] describes the states related to FT (see Figure 2.5:). There we can identify the phases "error detection" and "error recovery", which also can fail. We can categorize faults in sets, as already pointed out above. We have the set of faults that are happening in reality (unanticipated + anticipated faults), set of the anticipated fault and the set of faults we want to tolerate.

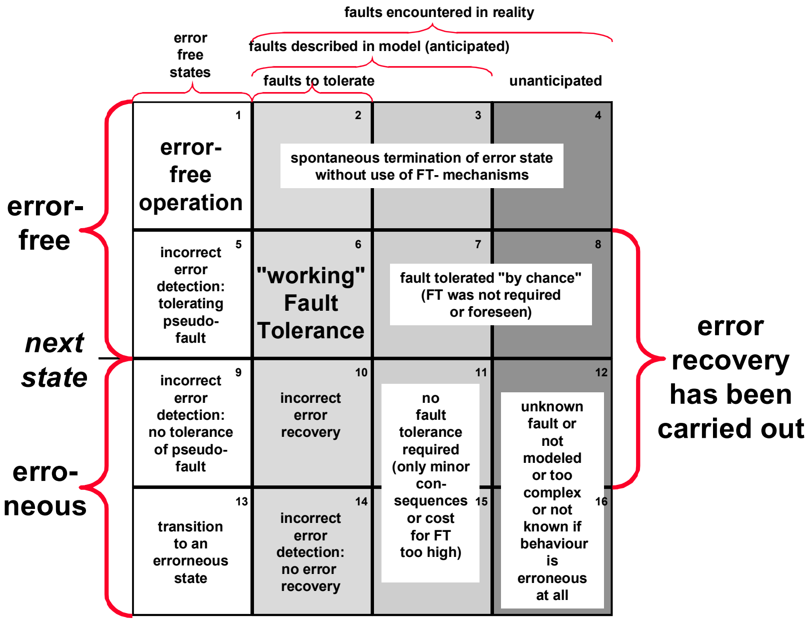

faults encountered in reality
error free states
faults described in model (anticipated)
faults to tolerate
unanticipated
1
error-free operation
2
3
4
spontaneous termination of error state without use of FT- mechanisms
5
incorrect error detection: tolerating pseudo-fault
6
"working" Fault Tolerance
7
8
fault tolerated "by chance" (FT was not required or foreseen)
9
incorrect error detection: no tolerance of pseudo-fault
10
incorrect error recovery
11
no fault tolerance required (only minor con-sequences or cost for FT too high)
12
unknown fault or not modeled or too complex or not known if behaviour is erroneous at all
13
transition to an errorneous state
14
incorrect error detection: no error recovery
15
16
error-free
next state
erro-neous
error recovery has been carried out

Figure 2.5: States of a System [Ech90]

The figure describes the states a system could reach after a fault condition is reached. The wanted behaviour of the system is switching from state 1 to state 6 and back to state 1. In this case the fault was foreseen and something was designed to tolerate it, also the error detection and the error recovery worked right. For all the resulting states of the upper half of the diagram (states 1-8) the system does not leave "errorfree" operation. Whereas in the bottom half (states 9-16) the fault is either not detected, the error recovery was wrong, or an erroneous state was reached because the fault was unknown. Also the possibility exists that no fault tolerance strategy was built to handle a known fault, because this would be to expensive in these cases (states 11,15), i.e. FT is intentionally not required.

2.1.3 System Structure and FT Modeling

2.1.3.1 *Recursive System Definition*

[Randell 77] cited by [AL81] and [Jal94] describes the system as follows (see also Figure 2.6:):

Definition 2.4: System

1. A System is an identifiable mechanism that maintains a pattern of behaviour at an interface between the system and its environment.
2. A System is composed of a set of components or subsystems, which interact under the control of design.
3. Each subsystem (or component) is a system in its own right. Each system at level N >1 is composed of a number of subsystems at level (N-1). The same holds for level N-1 to level 1. The decomposition stops at level 1, which is not possible or desirable to specify in further detail. Components at this last level are considered as atomic components.

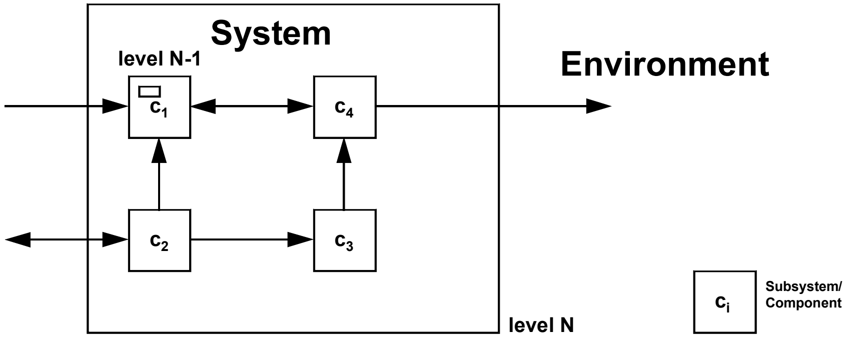


Figure 2.6: Example of a System, following the Recursive Definition of [AL81]

2.1.3.2 *Example: System Levels in the Mobile Radio System*

As an example for this kind of modeling and the different levels that will be encountered, we can look at the mobile radio system, which this thesis is focused on (See Figure 2.7:). The terms used in the figure will be explained later on, they aren't important right now for this example of system levels.

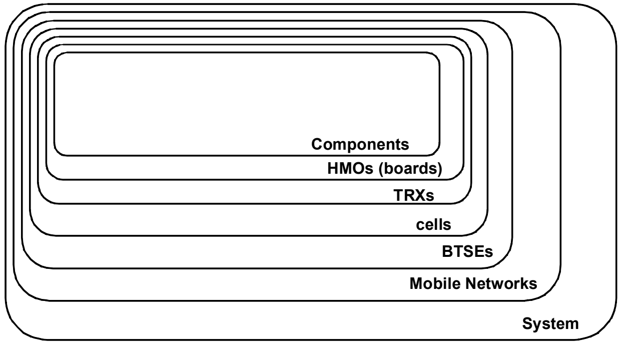


Figure 2.7: System Structure of the Mobile Radio System

2.1.3.3 *System Views*

Definition 2.5: Functional Unit (FU)

[JV86] defines the term functional unit (FU) for a unit that could be identified by its function or behaviour, where function or behaviour should be identifiable by actions.
A system is a FU that is structured into FUs. For single hardware components actions may not be identifiable, so FU is not appropriate there. [JV86] states that also other views than FUs for decomposing a system are possible:

- building units,
- maintenance units,

...
For the different levels of systems the terms elements, groups, subsystem and system itself are appropriate.
The following Table 2.1 from [JV86] shows the combination of these two dimensions:

Level \ View	Functional Unit	Building Unit
Element	Functional Element	Component
Group	Functional Group	Board
Subsystem	Functional Subsystem	Rack/Device
System	System defined by its function	System defined by its construction

Table 2.1: System Levels and Views

Obviously these terms are defined more precisely for the decomposition of systems than the terms of the recursive system definition above from [AL81]. Also a distinction between several views is made. We will come back to this aspect in chapter 3 where the dependency of building units to functional units is of interest for the failure model. A similar approach can also be seen in the structure-function model [Ech90] given in chapter 2.1.3.5 .

2.1.3.4 *Idealized Component*

In [RX95] the system is also modeled in above described way, but a component is modeled in more detail. It interacts with its environment as shown by Figure 2.8.

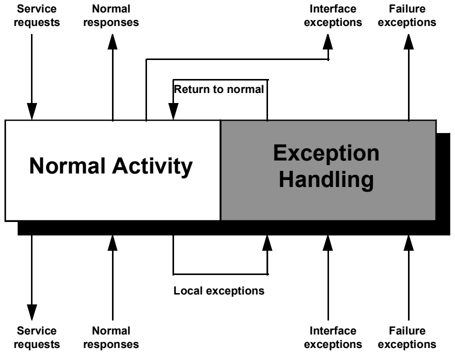


Figure 2.8: Idealized Component [RX95]

2.1.3.5 *Structure-Function Model*

[Ech90] proposes another kind of a system model. For fault tolerance evaluations it is also important to group components to subsystems and show function relations between the components. [Ech90] depicts components by circles and function relations between those components by arrows. Systems or subsystems are indicated by a set of components surrounded by a gray box.
The function relation f as depicted in Figure 2.9 can be interpreted differently:

- C1 is part of C2
- C1 provides data or resources to C2
- C1 is called by C2
- C1 is service provider for C2

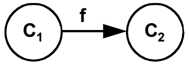


Figure 2.9: Function Relation

In Figure 2.10 an example of a structure-function model is shown.
There we can identify:
Components:

- P1 , P2 Processors

- C1 , C2 Interaction Software Components
- CM Communication Medium
- CS1 , CS2 Communication Software

Subsystems:

- Hardware HW = {CM, P1 , P2 }
- Software SW = {C1 , C2 , CS1 , CS2}
- User SW = {C1 , C2}
- Communication System={CS1 , CS2 , CM}
- Execution Unit 1={ C1 , CS1 , P1 }
- Execution Unit 2={ C2 , CS2 , P2 }

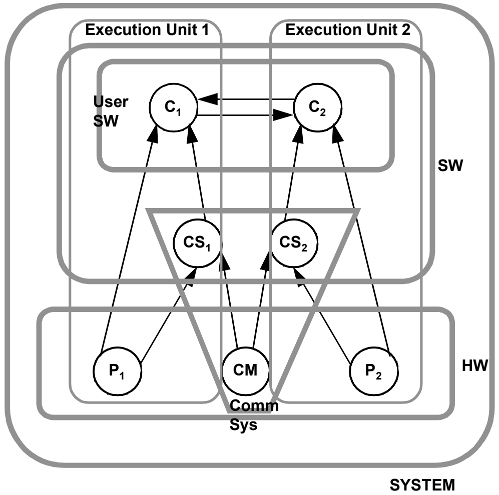


Figure 2.10: Example of a Structure-Function Model [Ech90]

For example some of the function relations can be interpreted in the following way: Software Subsystem C_2 uses Hardware P_2 and uses the communication software CS_2. The communication software itself uses Hardware P_2. Software Subsystem C_1 interacts with Subsystem C_2.

2.1.4 FT Methods, Design and Evaluation

FT methods concern the provision of redundancy. [Joh89] and [Ech90] distinguish between time, information, functional and structural redundancy. Redundancy also can be categorized by static, hybrid and dynamic redundancy:

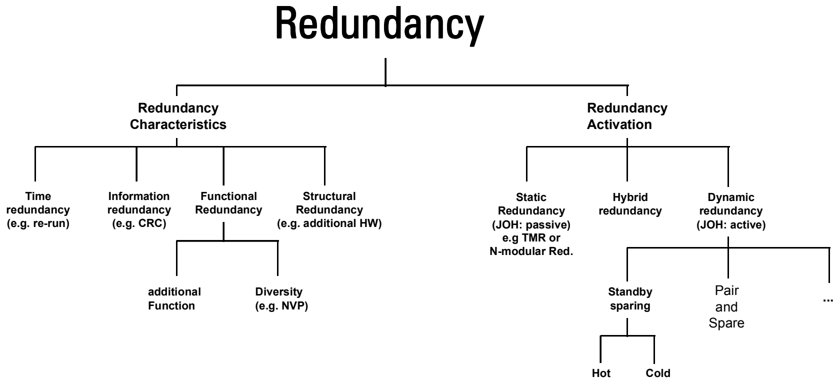

Redundancy
Redundancy Characteristics
Redundancy Activation
Time redundancy (e.g. re-run)
Information redundancy (e.g. CRC)
Functional Redundancy
Structural Redundancy (e.g. additional HW)
additional Function
Diversity (e.g. NVP)
Static Redundancy (JOH: passive) e.g TMR or N-modular Red.
Hybrid redundancy
Dynamic redundancy (JOH: active)
Standby sparing
Pair and Spare
...
Hot
Cold

Figure 2.11: Redundancy [derived from Joh89 and Ech90]

2.1.4.1 Design of FT Systems

[AL81] states the following design questions that concern the provision of redundancy

- How can the phases by applied?
- Where is FT required?
- To what extent, how much is necessary?

The factors for provision of redundancy are

- reliability requirements of the particular application
- limitation of costs
- terms of power/weight space constraints
- time overhead for FT
- cost of system failure

[AL81] remarks also that

- FT design is an iterative process, of identifying the possible faults and evaluating alternative methods.
- The aim of FT design is to minimize redundancy while maximizing reliability provided subject to constraints on the cost of the system.

For the discussion of the level where FT should be applied [AL81] says:

- Utilize (static) redundancy within the system versus enhancing the reliability of the components (by examining their structure and improving those components by redundancy).
- Redundancy at low levels within a system can be very economic and effective, but only against particular classes of faults.
- In contrast, redundancy at system level to cope with the unreliability of a component will be more expensive, but can be much more general and contend with a range of faults.

2.1.4.2 Evaluation of FT Techniques

For the Evaluation of FT Techniques the following measures appear consistently in many books:

While observing a system we can measure the following times: the time it takes until a new or just repaired system fails (Mean Time To Failures, **MTTF**). The mean time it takes to repair it (Mean Time to Repair, **MTTR).** And finally we can observe the Mean Time between Failures (**MTBF**):
The MTBF constitutes out of MTTF plus MTTR, because while repairing no failure would happen and the MTTF is measured with a new or repaired system.

Definition 2.6: MTBF

$$MTBF = MTTF + MTTR$$

- Availability

Definition 2.7: Availability

Availability is a measure of the degree to which a system is able to fulfill its intended function.
In common standard literature this degree is defined on a time basis. So availability is defined as:

$$Availability = \frac{MTTF}{MTTF + MTTR}$$, see [AL81]

In chapter 3.4 a new definition of availability is presented that will be more suitable for telecommunication systems, because in addition to the time also the traffic the system can handle is taken into account.

- Reliability

Definition 2.8: Reliability

Reliability is defined as the ability of a system to perform its intended function for a specified time under stated conditions.

In common fault tolerance literature the definition of reliability is further substantially restricted. This following definition of reliability will not be useful for evaluating mobile radio systems, for this reason a new reliability metric will be presented in chapter 3.5. So the following definition serves only for the discussion and introduction to the state of the art:
Reliability is characterized by a function R(t) (see Figure 2.12), which expresses the probability the system will conform to its specification throughout the period of duration t. R(t) cannot be known, but predicted.
Most widely used single numeric value is Mean Time Between Failures (MTBF, the average interval between failures) as an indicator for reliability [AL81].

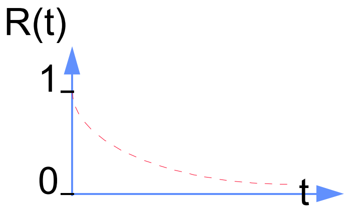


Figure 2.12: Reliability over Time: Function R(t) [AL81]

Definition 2.9: Mission Time

The time the reliability of a system needs to fall below a certain probability level is called mission time (See Figure 2.13:).
An example reliability comparison of a triple modular redundancy (TMR) system consisting of 3 identical components with a voter and a single system is shown in Figure 2.13: (adapted from [Joh89]). At a certain time the system consisting of 3 components will be less reliable than the single system, then the unreliability of the 3 components will be outperformed by a single system.

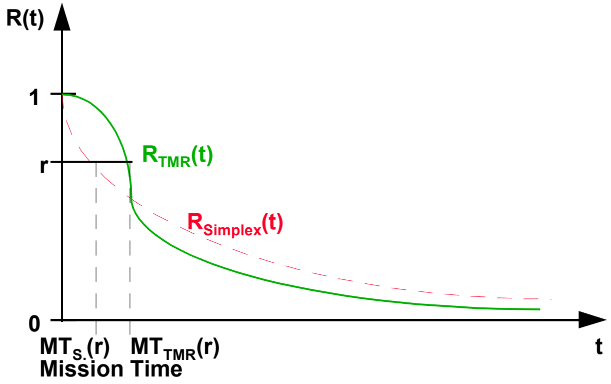


Figure 2.13: Reliability Comparison of TMR and a Single System [Joh89]

[Joh89] states that system comparisons can be made
- by Mission Time (MT) or
- by MTBF

[SS82] proposes some comparative measures for two systems
- Reliability Difference R2(t)-R1(t)
- Reliability Gain R2(t)/R1(t)
- Mission Time Improvement MT2(r)-MT1(r)

[JV86] states that if availability of a functional unit (FU) is defined by the strong criterion that the FU has to be able to fulfill all of its specified functions, then most of the computer systems would be unavailable. A better but weaker criterion is serviceability. The FU is serviceable, if the availability for the current tasks or jobs given to it is satisfiying.

2.1.4.3 Markov Reward Models for Performance and Dependability Analysis

[TMHV92] show how Markov Reward Models can be used for performance and dependability analysis. This will be quite similar to our computational model. The differences and analogies are described in chapter 3.4.4. They consider a homogenous finite-state continuous time Markov chain (see [Bre69], [GT96] and [TMHV92] for the derivations) with N states and a given generator matrix Q=[qij], where qij represents the transition rate from state i to state j. Let $P_i(t)$ be the unconditional probability that the Markov chain is in state i at time t, then P(t) is the transient state vector probability vector. The differential equation $\dot{P}(t) = P(t) \cdot Q,\ \mathrm{P}(0) = \mathrm{p}_0$ and the constraint for steady states that $\dot{P}(t) = 0$ lead to the steady state probabilities π , with the following characteristics:

$$\pi \cdot \mathrm{Q} = 0 \quad \text{and} \quad \sum_{\mathrm{i}} \pi_{\mathrm{i}} = 1$$

The steady probability π_i gives exactly the probability that the chain is in state i for $t \to \infty$.

If we define a vector $L(t) = \int_0^t P(u)\mathrm{d}u$, the $L_i(t)$ gives the expected total time spent in state i during the interval [0,t). Now the reward model is set up as follows: for each state i a reward rate r_i is associated. A reward $r_i \cdot t$ is accumulated for the sojourn time t of the process in state i.
Figure 2.14 shows an example of a Markov reward model:

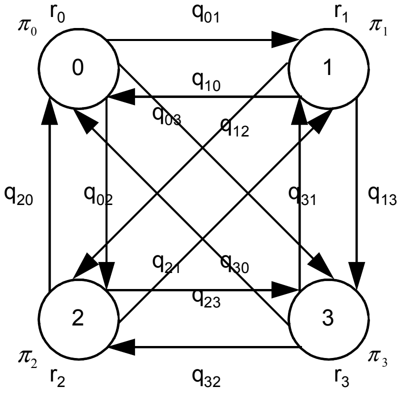


Figure 2.14: Markov Reward Model with four States

[TMHV92] describe several reward based measures, some of them are:

- the steady state expected reward rate: $E[X] = \sum_i r_i \cdot \pi_i$
- the expected accumulated reward in the interval [0,t): $E[Y(t)] = \sum_i r_i \cdot \int_0^t P_i(u) \cdot du = \sum_i r_i \cdot L_i(t) \cdot$

The defined reward rate depends on the kind of wanted performance or dependability measure. With an appropriate definition of the reward rate, predictions of bandwidth, throughput, availability, reliability, etc. are possible. For reliability and availability measures [TMHV92] show that it is useful to define absorbing states, where, once reached, the chain will remain in such an absorbing state.

2.2 Introduction to the GSM Network

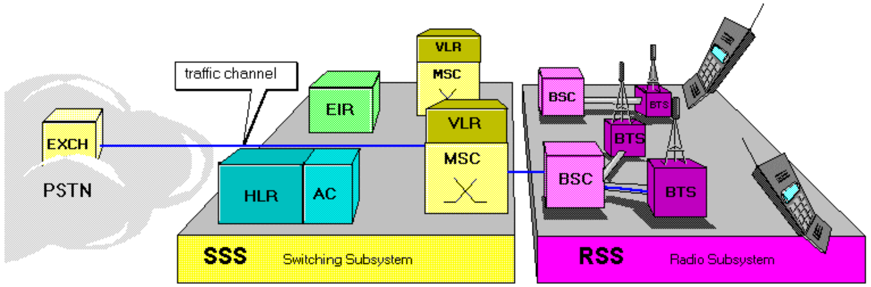


Figure 2.15: The GSM Network and the Network Elements

This chapter is an excerpt from [MOU94]:
The basic infrastructure of GSM is similar to all other cellular radio networks.
The system is realised as a network of contiguous radio cells, together providing a complete coverage of the service area. Each cell has a Base Transceiver Station (BTS) operating on a dedicated set of radio channels that are different from the channels used in adjacent cells.

BTSs are logically grouped together and controlled by a Base Station Controller (BSC) for functions such as handover and power control. A group of Base Station Controllers is served by a Mobile services Switching Center (MSC), which routes calls to and from the Public Switched Telephone Network (PSTN), Integrated Services Digital Network (ISDN), Public Data Networks and possibly various private networks.

2.2.1 Mobile Services Switching Center (MSC)

The MSC is the heart of the cellular radio system. It is responsible for routing or switching calls from the originator to their destination. It can be thought of as "managing" the call, being responsible for set-up, routing, control and termination of the call, for management of inter-MSC handover and supplementary services, and for collecting charging and accounting information. It also acts as the interface between the GSM network and the public telephony and data networks. The MSC also may be connected to other MSCs on the same network and to other GSM networks.

2.2.2 Base Station Controller (BSC)

The primary function of the BSC is call maintenance. As a subscriber moves around, he is likely to move out of one cell and into the area of another. The process by which the call is maintained without a noticeable break is known as handover, and on GSM this break is kept shorter than on most cellular systems.
On GSM, during a call, the mobile station (MS) "listens out" to all surrounding base stations and gives a continuous report of their signal quality to the BSC. This allows the BSC to make a very good decision on when to initiate a handover, and to which new cell.
GSM features power control of mobile stations and optionally the base stations. This reduces the signal levels radiated, minimising interference to other GSM users, and maximising battery life.

2.2.3 Base Transceiver Station (BTS)

The primary function of the BTS is to provide the base station radio transmission and reception. The BTS may contain one or more transceivers, to provide the required capacity. A cell site may be omni-directional or split typically into three directional cells. The hardware of one site is called Base Transceiver Station Equipment (BTSE).

2.2.4 The GSM Network Elements and Interfaces

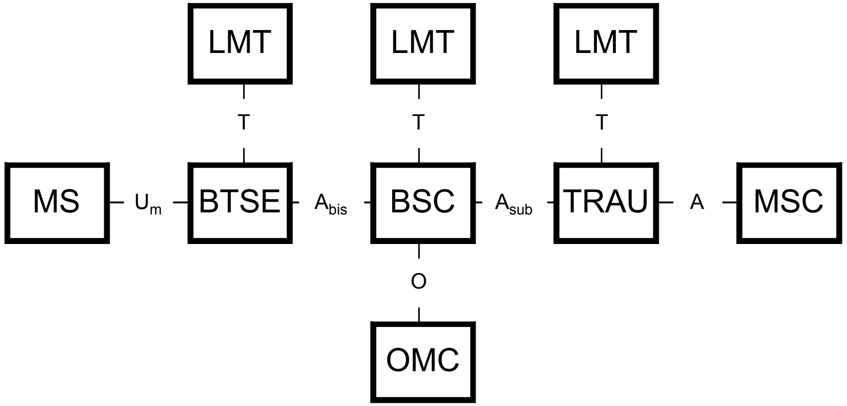


Figure 2.16: Network Elements and Interfaces between them

The local maintenance terminal (LMT) can optionally be attached to several network elements (NEs). It is not needed for normal operation purposes. The terms between the network elements are the names of the interfaces. U_m is the air interface between the Mobile Station (MS) and the Base Station. A_{bis} is the interface between the BTSE and the BSC. The A_{bis} is sometimes realized by cable through leased lines in the PSTN or it is done by radio relay links, this depends on the costs and the site location.
The A_{sub} and A interfaces are the interfaces for the corresponding NE further up in the hierarchy.
The Transcoding and Rate Adaption Unit (TRAU) is responsible for adapting the higher data and speech rates from the fixed network (64kbit/s) to smaller bitrates used for the radio network (such as 13 - Full Rate or 6.5 kbit/s - Half Rate Speech) and vice versa.

2.3 Fault Tolerance in Mobile Radio Systems

Publications on this matter are rare; reasons for this lack can be seen in the fact that the mobile communications are relatively new, and this topic often touches proprietary information of the telecom companies and is not covered so much by research. In this chapter the few publications that can be found will be summarized.

2.3.1 FT Techniques

[CKLS97] state that wireless networking is inherently unreliable. Various forms of interference on the wireless link result in changing bandwidth and low effective bandwidths due to high error rates. The only form of faults this paper covers are link faults. Hard and software faults of mobiles or base stations aren’t modeled. Fault tolerance is achieved by either redundant links and/or lowering the bandwidth of the application by changing the coding schemes (for example speech or video with less quality). That is why a link that cannot provide its full bandwidth can still be used.

[PKV96] compare several recovery schemes for wireless link failures and mobile stations failures. Mobile base station failures are not covered.

[TC96] propose a distributed diagnosis algorithm, where base stations initiate tests on all their neighbor base stations, so they have knowledge about the working state in their vicinity and with this knowledge they are able to allow/forbid the migration of mobile stations to cells in the neighborhood.
The algorithm is presented theoretically, there is nothing stated about how to incorporate this into existing mobile networks, like GSM.

In mobile networks the locations of all mobile stations have to be tracked efficiently and stored in a database. [RRD95] describe a fault-tolerant protocol for maintaining these location directories, even in case of base station failures, mobile station failures and communication corruptions. Also a state/transition diagram for the mobile hosts is given. No information about the probabilities of such failures is presented.

[LB97] show up an analysis of architecture and performance of the (fixed) link between the base station and the switching center. Delay and capacity various protocols and architectures are compared. Nothing can be found in the paper about the evaluation of link failures.

[BKCG98] present another kind of fault tolerant wireless data transmission by combining different overlaying wireless networks (infrared, WaveLAN, mobile radio, satellite) for data-transfers to one Mobile Host. Their approach is not only able to handle vertical handoffs between two cells of the same network, but also vertical handoffs between different networks. This scheme allows highly available and reliable data services. For performance and other reasons an agent for retrieving data from the fixed data network was used instead of end-to-end communication.

In [ABBG96], the authors describe an ARPA-supported project for rapidly deployable wireless network. The described network contains no fixed infrastructure like that for example base-stations, but uses multihop radios (MSs), which can establish links to each other and hand on traffic. The network is dynamically created in the field (typically for military applications). This dynamical reconfiguration of clusters (a kind of cells/group of MS) makes it very reliable although the MSs move around and failures may happen.
Dynamically changing cell or frequency plans aren't used in GSM. Here optimized network plans are made in long term cycles to assure the maximum performance of the network.

[LRS96] describe the goals and challenges of another DARPA project the Global Mobile Information Systems program (GloMo). The major requirements of wireless communication systems for military applications are given. The communication is expected to be highly reliable, robust, survivable and rapidly deployable, also higher performance requirements than for commercial systems in terms of QoS and bandwidth should be reached. To allow the rapid deployment only multihop networks with untethered nodes like described above can be used.

2.3.2 Real Implementations of Wireless Network Elements

Some of the rare publications from the manufacturers can be found in the following:

[WSH90] outline the architecture of the NEC GSM base station.
[HE95] give a detailed description of the Philips BSC and BTS architectures and functionalities. About FT aspects is only stated that each BTS can recover independently from others in the network. Also if permanent failures of BTS components happen a suitable reconfiguration of the remaining working components is triggered automatically.
[Gar97] describes the design of mobile testing equipment for maintenance personnel, for installing and maintaining PCS base stations even for use in unfriendly environments, as rooftops etc.

Authors from the Korea Telecom describe in [LHKC97] the design and architecture of their PCS wireless network (an upbanded GSM at 1.9 GHz). Schematics and design guidelines of their switching platform, their base station controller and their base station are given. The base station controller features modularity for capacity expansion and each component is doubled for high reliability. Another design principle due to cost aspects was to keep the base station simple and move the complex functions to the base station controller. About fault tolerance of the base station it is stated that the base station was designed for high reliability. In case of a "base station subsystem failure" the system is still able to operate with a decreased capacity or performance. This was achieved by maintaining independence of the subsystems and duplication critical subsystems such as the timing function subsystem. More details about these fault tolerance schemes are not given.

[CMAO97] describe the design objectives, architecture, and upcoming standards for IMT-2000 systems (the ITU term for third generation systems). A comparison to

second generation systems is given, especially what will be improved in IMT-2000 and what advance in technology for this will be needed.
Some of the objectives for IMT-2000 include minimizing complexity, cost and size and enhancing flexibility and performance. To meet the cost and reliability objectives [CMAO97] state that the technologies selected should have a level of complexity consistent with the state of technology, desired service objectives and radio environment.

2.3.3 Performance Measurement and Traffic Engineering

There is quite a great number of publications on this topic. Although the objective of most of them is to either measure and improve the radio channel or trying to enhance the capacity by sophisticated network planning. Nevertheless, the measures are useful for an system wide analysis fault tolerance. Therefore the following chapter 2.4 will introduce to the necessary traffic engineering terms.

[Goe92] shows up various performance measures for speech quality such as delay, noise, echo and quantisation distortion. Existing and future mobile communication systems are compared under these aspects. End user expectations are discussed.

[Gri96] gives an overview the ITU E.750 series of recommendations. The major aspects of these series are the definition of Grade of Services (GOS) parameters for traffic engineering, traffic models, dimensioning methods, and traffic measurements for third generation systems. The GOS parameters also include the Probability for End-to-End Blocking and the Probability for Unsuccessful Land Cellular Handover.

[Jab96] gives a review on the literature available in the area of performance modeling of wireless networks. As performance measures for such networks several probabilities are named:

- the probability for call blocking,
- the probability for handoff failure,
- the probability for forced termination or call dropout during a call
- the probability of call delivery

The handover traffic and newly arriving traffic are modeled as Poisson processes in a Markov chain model. Fault tolerance is mentioned by using overlapping cells and directed retry algorithms (see).
Equipment problems are not modeled and considered in this paper.

2.4 Telephone Traffic Theory

In this chapter the basic theory of telecommunication services is introduced. The formulas and definitions (see 2.4.2) form the basis of our models and will be used extensively in the following chapters.
Though most of them have their roots in the work of the famous telephone traffic pioneer A. K. Erlang (1878-1929) these formulas are still valid and very useful today [see TF81].

2.4.1 Telephony Modeled as Queueing System

Many problems in the telephony world are solved by queueing theory. The telephony service is described by a M/M/c/c queueing system [All90]. That means that interarrival time (time between two calls) of calls follows a negative exponential distribution (M) [JV86]. The same applies for the service time (call length) (M). The number of servers (channels) is c and there is no queueing of calls; so the capacity of the queuing system is also c (channels). Thus for the telephony model random (exponential) interarrival and service times (independent Poisson processes) with c identical channels are assumed.
With these assumptions the following formulas of A. K. Erlang can be derived [All90].

2.4.2 Erlang Formulas

Definition 2.10: Traffic Intensity A

The traffic intensity is a measure of the density of traffic. It is a non-dimensional quantity given in the unit Erlang and represents the mean number of calls existing simultaneously. It can be calculated by

$A = \lambda \cdot W_s$ [Erlang]

Equation 2.11: Traffic Intensity

where λ is the mean arrival rate (in our case of telephone calls) and W_s is the expected (mean) customer service time (in our case the average call length).

E.g. If 105 calls per hour are arriving with an average call length of 4 minutes, the traffic intensity A is 7 Erlang.

Definition 2.12: Blocking Probability B

Let us consider for example a given telephone system, a mobile base station or a computer system, that has a fixed number of possible lines, traffic channels or servers. With a given traffic intensity and the assumptions of chapter 2.4.1 we want to know the probability B that all lines, etc. are busy and the next user will be blocked. This can be calculated with the Erlang B formula.

Equation 2.13: Erlang B Formula (Blocking Probability)

$$B[c,A]=\frac{\frac{A^c}{c!}}{1+A+\frac{A^2}{2}+\ldots\frac{A^c}{c!}}=\frac{\frac{A^c}{c!}}{\sum_{i=0}^{c}\frac{A^i}{i!}} \quad \text{for a number of servers c}$$

Definition 2.14: Permissible Traffic

In some cases we want to know the „reverse“ Erlang B, if the blocking probability and the number of servers/lines/channels is given and we want to determine the permissible Traffic (the maximum allowed traffic intensity to hold the given Blocking). There is no equation for the reverse Erlang B $A[c,B]$. This problem can only be solved numerically using a Newton Approximation with the Erlang B formula.

Definition 2.15: Served Traffic

Due to the blocking some of the calls don’t get through, this traffic cannot be handled by the system and it gets lost. The real traffic intensity the system encounters must therefore be reduced by percentage of lost traffic. We get:

Equation 2.16: Served Traffic y

$$y=A\cdot(1-B)$$

Erlang has also given a more general formula that not only calculates the probability for the Blocking case (all servers in use). Furthermore, the following formula gives the probability that exact n of N servers are in use. (The Erlang B can be derived for n=N).

Equation 2.17

$$P_n=p(n,N,A)=\frac{\frac{A^n}{n!}}{\sum_{i=0}^{N}\frac{A^i}{i!}} \qquad (\text{with } 0\le n\le N)$$

For example a BTSE (2/0/0) (see chapter 3.3.1.3.1 for a description of this notation) with 14 traffic channels at a considered blocking probability of 2 % (permissible traffic A= 8.2 Erl) results in the following distribution:

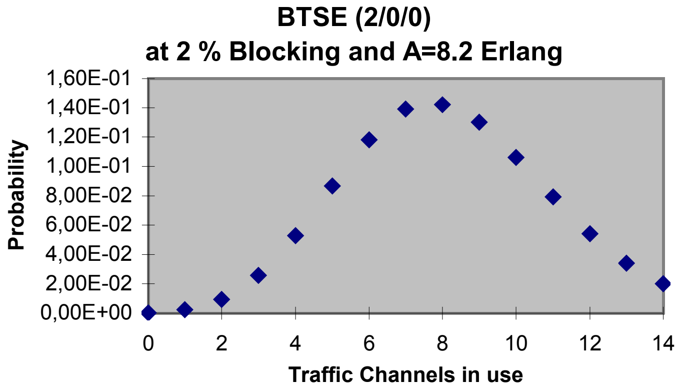


Figure 2.17: Example for the general Erlang Formula

As visible in the Figure 2.17 the Blocking Case, with 14 channels in use, happens with a probability of 2%. Also it could be recognized that the expected value of traffic channels in use is 8.2.

3 Failure Model (FM)

3.1 Motivation of the New Model

Common availability and reliability metrics (see: chapter 2) are insufficient and not suitable for modelling a call processing system like the BTS. In the following it will be explained: what are these insufficiencies and in what way would our metrics would be superior?

- The grade of degradation is not expressed by an MTBF. When reaching a failure condition many different degraded system states are possible in this distributed call processing system. The reaction in a distributed system could be more accurately described as a function loss or as an influence on a functional area rather than with a single MTBF value that describes how often failures (total or partial) take place.
- Availability normally is defined on system availability. In a call processing system it is more practicable and useful to define availability by the permissible traffic or by the blocking probability.
- The BTSE uses a threshold mechanism for software faults, i.e., if the same fault repeats multiple times another defense action (even total disabling of boards) can be chosen. If the resulting software failures would only be modeled by one single MTBF, the recovery/repair triggered by the threshold mechanism would be averaged for all different defense actions. In contrast to hardware faults the duration of recovery cannot easily be modeled by one MTTR.
- If we are interested in the causes of failures, in the contributions of parts and components and in the distributions of several failure types a more detailed model is necessary. If a more detailed model on based on MTBF is used, the figures for each component would not be comparable, since the importance of the component is not taken into account. Improvement of fault tolerance in a system will only be successful if we have an accurate model to describe the availability and reliability. The FM enables us to compare different fault tolerance schemes and evaluate the best one or those that can be realised with the least effort. The FM also shows up weak spots in hardware and software quality and it models these weaknesses according to the overall system behaviour. Only with a detailed modelling of the system behaviour we are able to calculate the influence of software failures and take the importance of several components for fulfilling the system functions into account.
- The common reliability metric would be defined on system failures, but for a call processing system the system failures are not as interesting as the "user level reliability of call handling" or more exactly the probability that calls are dropped due to failures.

3.2 Introduction to the Model

This chapter (3) describes a failure model (FM) for the BTS-system. This should be seen as an example for modeling telecommunication systems. Where something special for the BTSE is modeled and the formulas or models are not valid in general it will be remarked in the following.
With the FM a solid basis for analysis and improvement is given. System designers will be given an appropriate method to evaluate new fault tolerance schemes and ideas.
The use of gathered data, which describes the real behaviour of a running system for a period of time, is essential for the method that will be described. After the setup of the model different such data sets (alarmlogs) from different OMCs (Operation and Maintenance Centers) are used as input data for the models, to estimate the current reliability and availability of the BTSE in form of the newly defined metrics. With this estimation we are able to identify the current weak spots and the potentials of improving soft- and hardware fault tolerance.

If we had one universal metric to evaluate fault-tolerance schemes we could select the best one by this single criterion, but unfortunately availability and reliability represent two not reducible dimensions of the problem. Only a combination in form of loss of money for the operators or another definition of „damage“ for the operators would be possible.
As a starting point for this objective mathematical ratings from several operators about failure scenarios should be available. They should give information about „how much damage“ a given failure scenario would be in comparison to other scenarios.
„Damage“ could for example be expressed as the total loss of money for an operator caused by failures. To calculate this a lot of unpredictable things must be taken into account. For example if users are annoyed, because their calls have been prematurely released very often, they will change the operator. The operator begins to lose market shares and money.
The rating wouldn’t be easy, just think about the question, what is worse "losing an established call" or "not being able to phone at all". One answer will be "it depends on the situation". Such ratings do not exist so far. So for evaluating fault-tolerance schemes, we have to calculate those metrics independently and look at the advantages the new schemes may have in several different dimensions.

Ratings which operators give may be different from ratings that users will give about reliability aspects, and manufacturers surely have a different meaning about these aspects also, the user could be more concerned about his lost call, whereas the operator is more concerned about paying the service personnel and equipment to fix the fault and the manufacturer is concerned about locating the bug in a software release or about the costs to realize a highly available/reliable system.

A more neutral way to look at problems in communication systems is to regard it as quality parameters of telephony. There are some vital definitions of quality in telephone systems. We can express „damage“ of failures of telephone systems as variations of these parameters.

Such measures are:

- the premature release probability (probability that a call is cut, before the user wants this),
- the blocking probability (defined as the probability that a user cannot establish a new call) or from a different point of view the traffic intensity a system can handle at a given blocking probability,
- the network availability,
- speech quality and other quality of service measures and
- measures for short interruptions of calls.

These measures mostly match the view the user has on the system, because he just wants to make a telephone call and is not interested in the technical details of how this is accomplished. Operator or manufacturer views are also dependent on these vital telephony definitions.
So for the FM of the BTSE it was decided to calculate the (in our opinion) top three important metrics:
1. The definition of availability on the permissible traffic of a single system (permissible traffic time - PTT, refer to chapter 3.4 for the definition), this takes the scalability of systems, their graceful degradation and the importance of the components on the system functions into account. A directly associated measure: would be the blocking probability for a single BTSE (see chapter 3.4.3 for how this is related) and
2. the premature call release probability PRP (a user level reliability measure),
3. the probability that a user cannot re-establish after his call has been released (this expresses the availability of the network).

The other measures mentioned above, as speech quality and short interruptions of the call, are not investigated, because nearly all hardware or software faults that are known in the BTSE (with the exception of problems with the antenna equipment) have no influence on the quality of calls.
Furthermore, in nearly all cases they release calls completely or they influence the ability to make a telephone call at all. Short interruptions or bad quality in receiving speech are an important topic in mobile communications, but not the most important one when investigating those systems from the FT point of view.

In Figure 3.1 we give an overview to these metrics, how they are derived and what results can be evaluated.

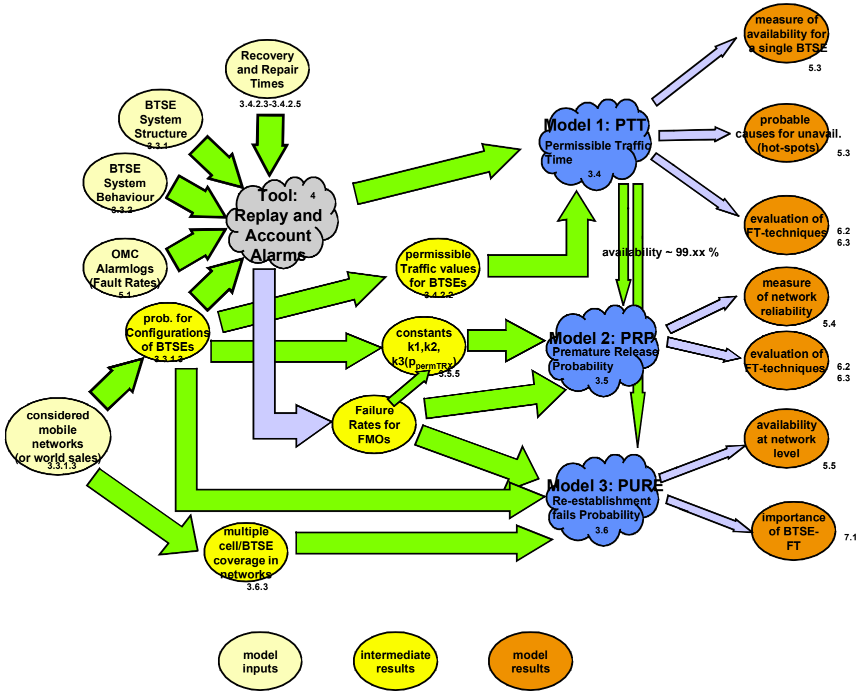

Recovery and Repair Times
3.4.2.3-3.4.2.5
BTSE System Structure
3.3.1
BTSE System Behaviour
3.3.2
OMC Alarmlogs (Fault Rates)
5.1
Tool: Replay and Account Alarms
4
prob. for Configurations of BTSEs
3.3.1.3
considered mobile networks (or world sales)
3.3.1.3
permissible Traffic values for BTSEs
3.4.2.2
constants k1,k2, k3(p$_{permTRX}$)
5.5.5
Failure Rates for FMOs
multiple cell/BTSE coverage in networks
3.6.3
Model 1: PTT
Permissible Traffic Time
3.4
availability ~ 99.xx %
Model 2: PRP
Premature Release Probability
3.5
Model 3: PURE
Re-establishment fails Probability
3.6
measure of availability for a single BTSE
5.3
probable causes for unavail. (hot-spots)
5.3
evaluation of FT-techniques
6.2
6.3
measure of network reliability
5.4
evaluation of FT-techniques
6.2
6.3
availability at network level
5.5
importance of BTSE-FT
7.1
model inputs
intermediate results
model results

Figure 3.1: Overview to the Inputs, Models and Results

So, before we begin to set up the FM once again let us summarize the main objective of the FM:

- Improvement of fault-tolerance in a system only makes sense if we have an accurate model to describe the availability and reliability. The FM should enable us to compare different fault tolerance schemes and evaluate the best one or those that can be realised with the least effort.

3.3 Model Basics

This chapter outlines the major points of the structure, configurations and behaviour of the Base Station. This will be necessary to set up the appropriate models. Some definitions and notations used for the models are introduced here. The chapter will give the basics, necessary to understand the models and metrics itself.

3.3.1 BTSE Hardware

The following diagram from [F1S94] shows the general hardware structure of a typical BTSE variant. Meanwhile several modifications of the hardware have been made and many different versions exist, especially for the RF-hardware parts. However, the general principle can still be used for this analysis.
See also chapter 0 about new hardware.

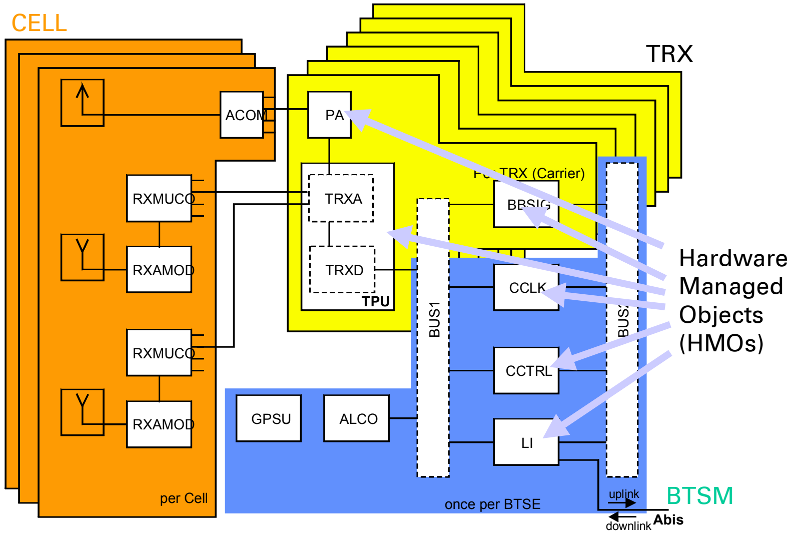


Figure 3.2: The SIEMENS BTSE Hardware and its Hardware Managed Objects

Explanation of the abbreviated building blocks:

A_{bis}:	BTSE-BSC Interface
ALCO	Alarm Controller Board
ACOM	Antenna Combiner (used only for downlink)
BBSIG	Base Band and Signalling
CCTRL	Core Controller

CCLK	Master Clock
GPSU	Generic Power Supply Unit
LI	Link Interface
PA	Power Amplifier (downlink)
RXAMOD	Receiver Antenna Module (uplink)
RXMUCO	Receiver Multicoupler (uplink)
TPU	Transceiver and Processor Unit consisting of an analogue and a digital part (TRXA/TRXD)

3.3.1.1 HW Structure Description

In the following the hardware structure of the BTSE is briefly described by the flow of traffic (speech) frames through the system.
In downlink direction the speech data takes the following flow through the system:
TRAU speech frames arrive from the BSC via the Abis PCM30 link (which is timeslot based and could be established via microwave or cable) at the LI. The link interface gives them onto BUS2 and the corresponding BBSIG takes them. In the BBSIG the frames are encoded, encrypted, interleaved and mapped together to bursts for transmission. These bursts are given via BUS1 to the TPU, a training sequence is added in TRXD. In the TPU the burst is further handed on to the TRXA, which does the analogue signal processing part, like the GMSK modulation. The resulting output signal is amplified to the required power level in the PA. Then the different signals from the TPUs are combined together in the ACOM part; there a transmission band filter also filters out noise.
Finally, the transmission signal gets to the right transmission antenna and is sent to the mobile station through the air.
In the uplink direction of speech from the MS the components are used in the opposite order. Minor differences appear in the receiving antenna equipment: For example two receiving paths by two antennas and doubled RXMUCO and RXAMOD components are used. If one receiving antenna path fails, the cell will be still able to receive, but with a lower signal level, which may lead to the loss of distant MS calls.

3.3.1.2 HMO-FMO Definition

The components of the BTSE in the solid white boxes in Figure 3.2: are the Hardware Managed Objects (HMOs). They are either processor boards or non-processor boards. Processor boards may have hardware and software faults, non-processor boards have only hardware faults.

Definition 3.1: HMO

An HMO is an element out of the set of possible boards of the BTS. Modeling the system with HMOs means that we are looking on the system on a "Board-Level". Note that those HMOs for cell related or carrier related hardware may appear multiple times in one BTSE, since one BTSE may setup several cells and several transceiver.

$HMO \in \{\mathrm{acdc}, \mathrm{acom}, \mathrm{alco}, \mathrm{bbsig}, \mathrm{cclk}, \mathrm{cctrl}, \mathrm{gpsu}, \mathrm{li}, \mathrm{null}, \mathrm{pa}, \mathrm{rxmuco}, \mathrm{rxamod}, \mathrm{tpu}\}$
TRXA and TRXD are not HMOs. Together they built up the HMO tpu. The busses are also not modeled as HMOs.
Additionally the HMO null is used for the case where a fault happened that didn't have any influence on call processing. In case of faults or even in case of failures (see Definition 2.1-Definition 2.3) alarm messages (see Definition 3.5) are produced. If no failure condition is reached the alarm may be just an informational message or warning.

Several components (HMOs) together fulfill a function. Together they form a Functional Managed Object depicted in the following Figure 3.3:.

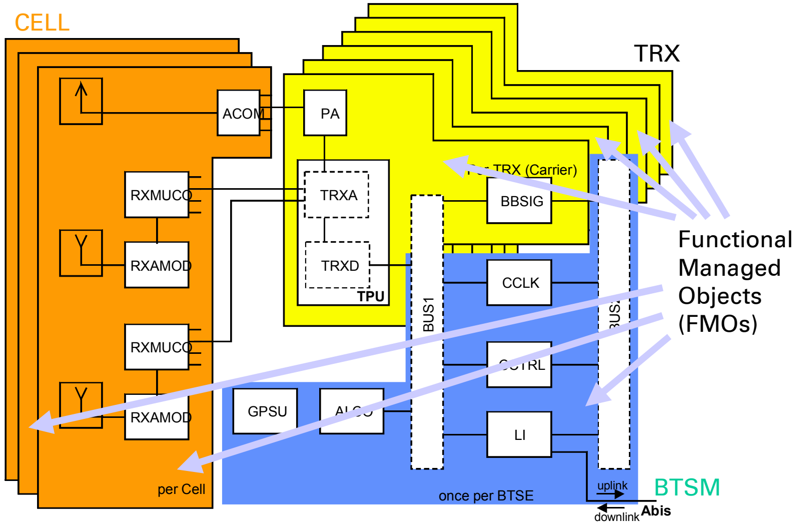


Figure 3.3: The SIEMENS BTSE Hardware and its Functional Managed Objects

The BTSE consists of three different types of FMOs:

- TRX: Transceiver (Transceiver = Carrier related Hardware)
- CELL: cell related hardware (BTS Base Transceiver Station)
- BTSM: Base Transceiver Station Manager (core Hardware per BTSE)

The FMOs also identify the fault-areas, because a failure of an HMO may have an impact on a specific functional managed object. See chapter 3.3.2.4 below about this topic.

Definition 3.2: FMO

A FMO is an element out of the set of possible functional managed objects.
$FMO \in \{NULL, TRX, CELL, BTSM\}$
FMOs are marked in Figure 3.3 as different gray building blocks. TRX and CELL FMOs may appear multiple times, the BTSM FMO only once in a BTSE.

Additionally $NULL$ is used for the case where a fault happened that didn't have any influence on call processing. So a failure condition isn't reached and the alarm may be just an informational message or warning.
The definition of HMOs and FMOs can be applied to any other system. In chapter 3.3.2.4 the dependency of both is displayed. For other systems the dependency may be more complex, but the principle remains the same.

3.3.1.3 BTSE Configurations

In a network different kinds of BTSEs can be found. For different traffic scenarios (cities, rural areas, etc., ...) different combinations of base station components are needed. Such a combination will be called *configuration* in the following. These configurations differ in the number and combination of traffic channels. The statistics of used configurations in a network clearly have a strong influence on the statistics of failures. This chapter will describe which configurations are used today, what properties they have and how statistics about them will be used for the calculation of the metrics.

Variants of telecommunication equipment are typical for telecommunication systems. The estimated traffic the system should handle is the major factor which influences the configurations or extensions that are used. In the case of the BTSE the estimated traffic is a function of the map location which finally has an influence on the estimated traffic of sectors and cells and the configuration used.

3.3.1.3.1 Notation and Description

The common notation for configurations is (x/y/z), where x,y,z are the numbers of TRX per sector. The current maximum of sectors is three, future extensions to for example 6 will not be considered, so the (x,y,z) notation is adequate.
One TRX consists of 8 channels (by TDMA). The TRX channels are used for both traffic channels and signalling channels.
Different configurations implicate different counts of HMOs and different probabilities for faults that will also implicate different failure rates.

3.3.1.3.2 Example (2/2/2) BTSE

The following example (see Figure 3.4) depicts a (2/2/2) BTSE and its use in the hexagonal cell structure.

Two TRXs (which means 16 Channels) are used for one cell. As shown in Figure 3.5:, these cells are used in a sectorized fashion, i.e. 3 different antennas are used to send/receive into/from 3 different directions. So the BTSE is located at the edge of three hexagonal cells, which are covered by it.
Another approach is the omni-directional cell, where the BTSE is located in the middle of one cell and is not transmitting in a special direction. However, there exist more different configurations for different application purposes.

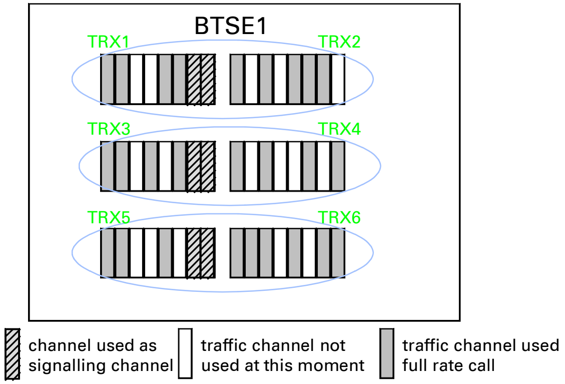

BTSE1
TRX1
TRX2
TRX3
TRX4
TRX5
TRX6
▨ channel used as signalling channel
□ traffic channel not used at this moment
■ traffic channel used full rate call

Figure 3.4: Example BTSE Configuration (2/2/2) and its Structure

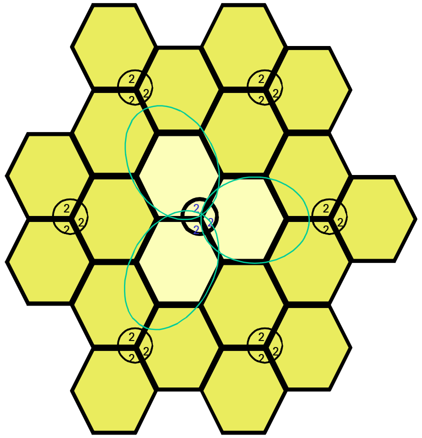


Figure 3.5: Example BTSE Configuration (2/2/2) in the Hexagonal World

3.3.1.3.3 Real BTSE Configurations in the World

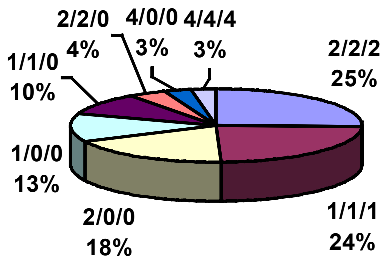
The example above already gave an idea of the fact that one important input for all the models are the different configurations of BTSEs.
A sample of 8738 sold SIEMENS BTSE spread over the whole world was investigated. Typically, two different configuration and equipment related scenarios exist in the lifetime of mobile networks. In the "Start-up"-scenario the operator has just begun his operational service, he normally starts with the smallest BTSEs available to set up some coverage in a country. As more and more users subscribe to his network, he has to extend it by larger BTSEs where traffic demands it. This defines the second scenario: "Expansion".
With the sold BTSE sample all kinds of operators and scenarios are represented. It describes a good representative mix of hardware for fault-tolerance analysis. One other aspect should be noted here: with the tremendous demand in mobile communications and millions and millions more of mobile users, the Base Stations have to handle more traffic, which means a drift to larger and larger configurations. We will come back to this aspect in chapter 7.3, where we can use the FM for an extrapolation of future reliability and availability, with just assuming the same fault-tolerance for the FMOs as nowadays and defining another mix of configurations.

Note that the current SIEMENS BTSE can only be configured for three cells with a maximum of 6 TRX per cell. Some more facts about the world sample are:

- Only 32 different configurations appeared in the sample.
- 6 configurations dominate in 90 % of it.
- The average number of cells per BTSE in the sample was 2.188 Cells; the average number of TRX per BTSE was 3.579. So the average number of TRXs per cell was 1.636.

For a more convenient further calculation the occurring cases of the sample have been reduced to the major ones.

The modified configuration count looks as follows (see the following Table 3.1 and the pie chart)



Configuration	Count	Probability
2/2/2	2224	0.2545
1/1/1	2106	0.2410
2/0/0	1589	0.1819
1/0/0	1099	0.1258
1/1/0	845	0.0967
2/2/0	370	0.0423
4/0/0	254	0.0291
4/4/4	251	0.0287
Sum	8738	1.0000

Table 3.1: Configuration Distribution for further Calculations

The average number of TRXs per BTSE is now 3.563.
The average number of TRXs per cell results in 1.629.
Omni-cell configurations (1/0/0, 2/0/0, 4/0/0) appear with a probability of 33.7%, three-sectorized configurations (1/1/1, 2/2/2, 4/4/4) with a probability of 52.4% and two-sectorized configurations (1/1/0, 2/2/0) with a probability of 13.9%.

Where no data from operators was available about the configurations of their network, these statistics were used to estimate the size of their BTSEs for the calculation of the metrics.
Slightly better results would be gained, if the configurations of the BTSE where the alarms were logged were used, instead of the configurations in the whole world. Nevertheless, the sample is big enough to show a good real distribution of BTSE configurations necessary in networks.

3.3.1.3.4 Properties of BTSE Configurations

As already stated, all channels that are set up by the TRXs may be used as traffic channels or as signalling channels. According to the size of the BTSE a typical number of signalling channels is used, Table 3.2: shows this for all used configurations.

Configuration	Cells	Signalling Channels per cell	Number of TRX	Total Channels	Traffic Channels	Signalling Channels
1/0/0	1	1	1	8	7	1
2/0/0	1	2	2	16	14	2
1/1/0	2	1	2	16	14	2
1/1/1	3	1	3	24	21	3
4/0/0	1	2	4	32	30	2
2/2/0	2	2	4	32	28	4

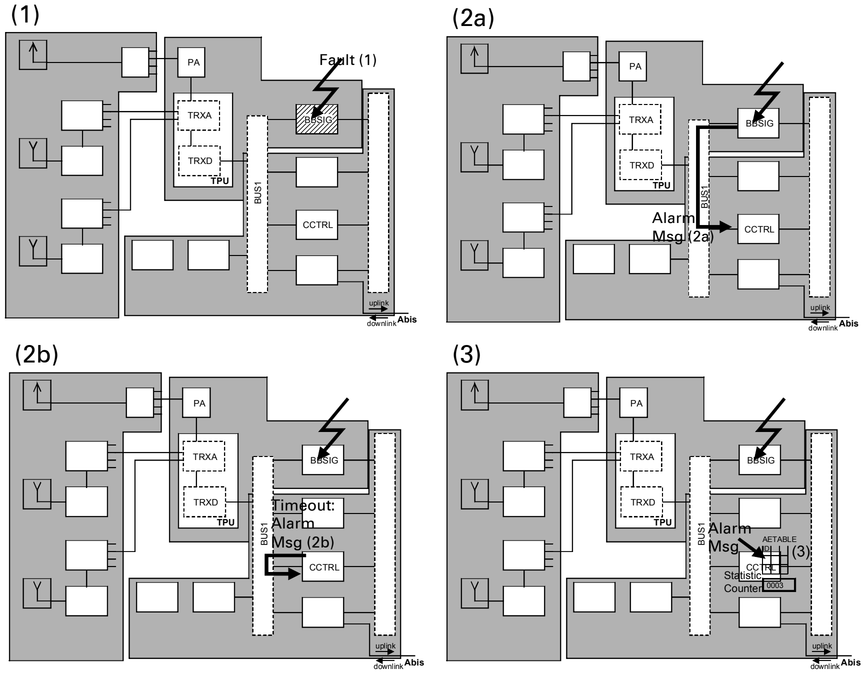
2/2/2	3	2	6	48	42	6
4/4/4	3	2	12	96	90	6

Table 3.2: Properties of BTSE Configurations

3.3.2 BTSE Behaviour and Software

3.3.2.1 Example: Software Fault on a BBSIG

Figure 3.6 shows an example of a software fault that leads to a TRX failure:
(1): A software fault happens on BBSIG
(2): By an alarm message the central controller board gets notice of this (2a). The alarm message may also be generated by a timeout of communication supervision to the BBSIG (2b).
(3) The alarm and error handling function on the central controller looks into a statistics table, which shows how many times this alarm occurred before. So an appropriate defense action can be chosen. (4) In this example the defense action „reset local processor (BBSIG)“ is chosen and sent or triggered to the BBSIG.
(5) The defense action is executed; afterwards a startup (also recovery) of the board has to be made.



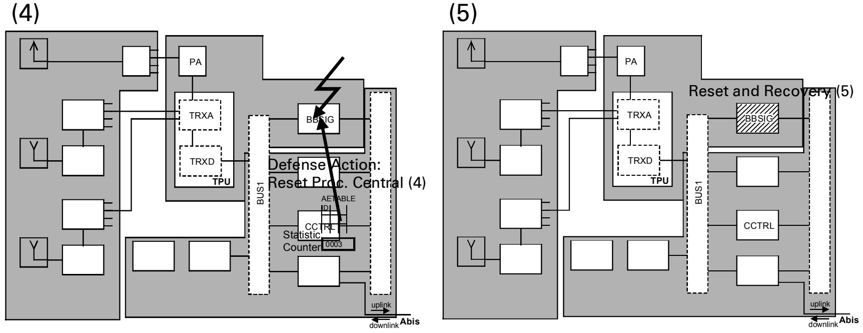


Figure 3.6: Example of a Software Fault on HMO BBSIG

During steps (1)-(5) the corresponding FMO is totally unavailable, (a partial system failure), see the following Figure 3.7.

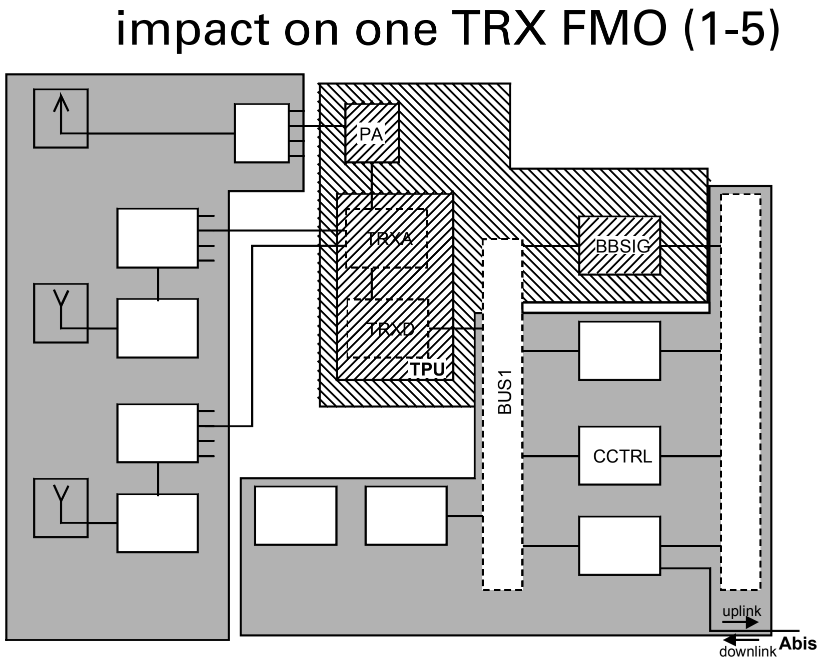


Figure 3.7: Impact of the Software Fault on HMO BBSIG

3.3.2.2 *Defense Actions*

As a result of an alarm (or detected fault) a defense action is chosen. The following table lists the purpose of the existing defense actions:

Defense Action	Type	Purpose
no action	Warning	Only an alarm report is made, nothing is changed in the system after this alarm. For debugging and maintenance purposes only.
protocol only	Warning	Same as „no action“, has to be reported.
reset proc. local	Restart	Reset the local processor board, where the fault came from (self-triggered).
reset proc. central	Restart	Reset of a peripheral board by the central processor board (triggered by central control).
disable non-processor	Disable	Disable non-processor equipment.
disable processor	Disable	Disable processor board.

Table 3.3: Descriptions of Defense Actions

Definition 3.3: Defense Action

The possible defense actions can be simplified to three different cases: either nothing is done, a board is restarted or a board is disabled.

$$defense = \{Nothing, Restart, Disable\}$$

Note that these are very few possibilities, especially for software induced failures more sophisticated types of defense actions could be imagined. We will show later that the recovery of software failures is very fast in comparison to hardware defects, i.e. such defense actions would not significantly improve the availability.

The selection of an appropriate defense action is complex: The sophisticated software solution, within the SIEMENS BTS, which allows a three-stage defense scheme, works as follows: Counters for each alarm type are held on each BTSE. If there are multiple occurrences of a specific alarm on one day, depending on the counter a more aggressive defense action is taken.

For example a minor software problem may be ignored for the first ten times an alarm is generated. Then if the counter exceeds ten it is assumed that the software problem is more serious so the board is restarted. If the problem still exists after ten restarts (20 alarms), it is assumed that the hardware, e.g. a processor, is defective, then the board (HMO) will be disabled.

So the defense action depends on the Alarm ID and the number of Alarms of this kind that already happened. The selection of the appropriate defense action goes via a large alarm table. In this table a maximum of three different defense actions for different thresholds of the counters can be defined.

3.3.2.3 *The Alarm Database*

All the known alarms for faults of the BTSE (and also other NEs) are summarized in alarm sheets and organized with an alarm database. This database is used for displaying faults at the OMC and for the system-wide development of the software: Alarms from all the NE in the system should be unique and the behaviour and properties of each alarm should be clearly specified. Each entry in this database is for one alarm. An entry consists of the following important information:

HEX Error ID	Error-ID:	Short Error String	De-scription	Alarm Type	Probable Cause	Defense Action	Severity Level	Lower Threshold			Upper Threshold		
A000	CCH_AEID_BHBAC_ISR_DROPOUT	Timeslot interrupt dropout	"/fch_bhbac/ detected that ...	Processing error	Software error	No action	Critical	Y	64	reset proc. centr	Y	255	disable processor
A001	CCH_AEID_BUMAP_TABLE_CORRUPTED	Burst mapping table corrupted	"During evaluation of the channel type ... "	Processing error	Software error	No action	Critical	Y	64	reset proc. centr	Y	255	disable processor
A002	CCH_AEID_BHBAC_WRONG_TPUADR	Wrong sender in traffic burst from TPU	"Uplink bursts were received on BUS1 .."	Communication error	Internal protocol error	No action	Warning	Y	5	protocol only	Y	32	reset proc. central
A003	CCH_AEID_MQ_OVERRUN	Message queue for U2 uplink is full	"The queue used to forward ..."	Processing error	Software error	No action	Warning	N			N		
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

Table 3.4: Some Example Entries of the Alarm Database

In total, 714 alarms are known/handled by the database.
This database fundamentally describes the behaviour of the system and/or its components:

- The field HEX-Error-ID describes a unique ID of this alarm (fault).
- The field defense action defines the action that is done during recovery to eliminate the error.
- The more aggressive defense actions happen if the counter for this alarm exceeds the values given in lower threshold or upper threshold. In Table 3.4 we can also look up the corresponding defense actions for these cases. The last and worst defense action is „Disable“ and human interaction is necessary. The maximum is three stages, so two thresholds can be exceeded.
- Stage 0 is the default defense action.
- Stage 1 and 2 are optional. They are defined by a "Y" in the lower or/and upper threshold definitions. There the more aggressive defense actions and the definitions of the thresholds also can be found.

The database contains also fields to specify on which HMOs the alarm can happen and which software subsystem produces this alarm.

3.3.2.4 HMO-FMO Dependency

As already stated, there are three different functional objects (BTSM, CELL and TRX), they are also used in the dependency graph of the configuration and state management (CS) (see [DCS97]). This graph shows the dependency of working states between the HMOs and the states resulting for the FMOs. This graph (see Figure 3.8) can be used for a forward state propagation: i.e. if an HMO fails, which influence has this on the FMO (going up in the graph).
The operators in the graph give the dependency:

- an AND means: every branch below (sons) should work correctly, so that the father (branch up) is working correctly.
- an OR means: one of the branches below must work, so that the father will work properly. This means we have redundancy in the hardware. The only example for this is the OR for the receiving diversity branches.

The interdependencies of the FMOs (a BTSE consists of cells, the cells consist of TRXs) are also calculated by the model, not by the graph.

Note that FMOs are not identical with the real complete functional objects as seen by the user/operator, like an object of the complete BTSE or objects for the different cells. This is calculated in the BSC. So if for example a cell fails, the TRXs of these cells also are not able to fulfill any service.

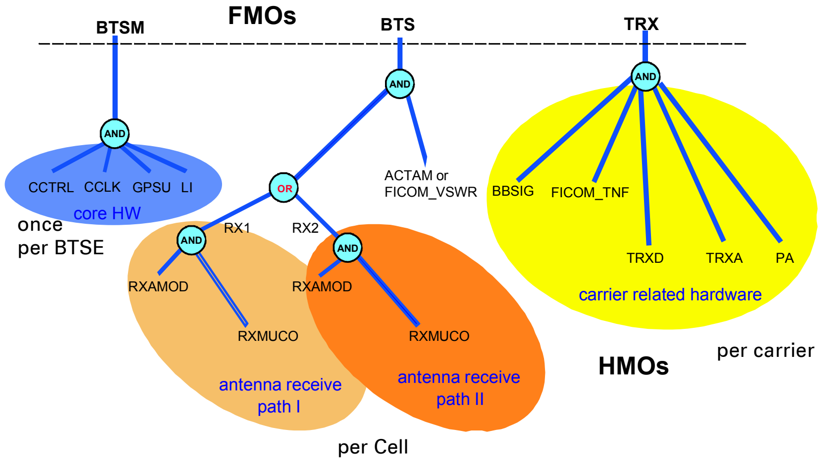


Figure 3.8: Simplified Forward State Propagation Graph for use with the FM

Such dependency graphs may be derived for other systems as well. Although it may be inconvenient if special load sharing redundancy schemes (or schemes where n of m components must work to fulfill the function) are used. Other graphs or formulas must be used here to calculate finally the traffic the system is able to handle when some components fail (see chapter 3.4).

3.3.2.5 Software Subsystems in Charge of FT and Recovery

The following Figure 3.9 shows an interaction diagram, for the software subsystems in charge of fault tolerance and recovery. In this chapter it will be described what was realized for the BTSE, in other telecommunication system such functions can also be identified, their purpose matches the phases of fault tolerance given in chapter 2.1.2.
Several error detectors in hard- and software, even a subsystem (fault protection-FP) that makes periodic tests, try to identify a faulty behaviour of each HMO. If a fault occurs, then after some time the error detector will detect an erroneous state or not responding hardware and inform the local Alarm and Error handling (AE) Subsystem about the HMO fault. This informs the Central AE on the Central Controller Board (CCTRL). For the selection of an appropriate defense action AE uses the data from the (offline) Alarm Database, which is changed only for the software development (see Table 3.4). From this table AE immediately chooses a defense action and sends it to the Configuration and State Management (CS) to calculate a corresponding output action (e.g. disable a HMO). CS communicates then with RC (Recovery) to perform the necessary actions (executes them). From the Central AE also a Failure Event Report i.e. Alarm (Note: that the correct term would be Fault Event Report, because this happens also when there is no impact on service) is sent via the BSC to the OMC. The message contains detailed data about the occurred fault.

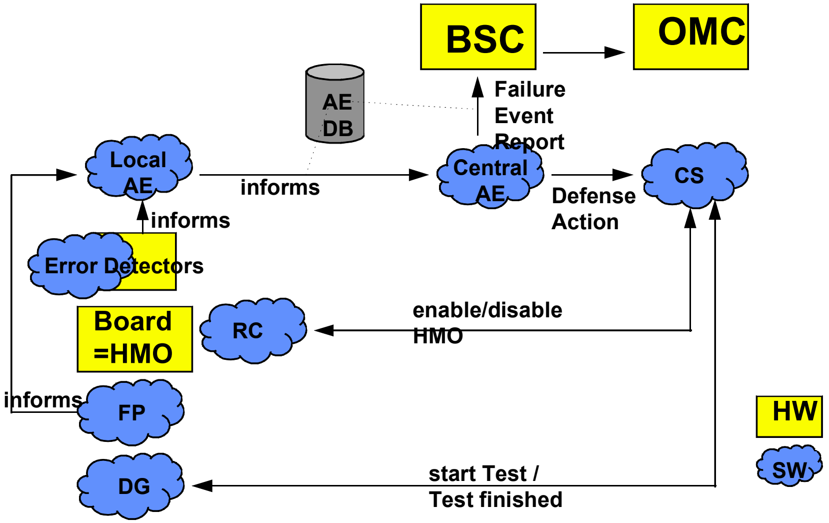


Figure 3.9: Software Subsystems in the BTSE Dealing with Faults and Fault Tolerance

The error detection facilities (see chapter 2.1.2.1) are not assigned to a single software subsystem, they are distributed over the whole BTSE soft- and hardware. Periodic tests can be assigned to FP. Some of the timing checks can be assigned to hardware watchdogs in the system, but the other kinds of error detection methods are spread other the whole operation and maintenance and call processing software, from where AE is informed. Damage confinement and assessment (see chapter 2.1.2) can be assigned to CS and AE, they deduce from the kind and origin of alarm to the extent of the error and to the necessary defense action. The Error Recovery facility can be identified by the subsystem RC. The fault treatment and continued service facilities are expressed by the threshold mechanism of AE, and by the reconfiguration that is guided by CS. Operators can start via an operator command extensive tests within the subsystem DG (diagnosis) to check disabled hardware if it is still usable.

3.3.2.6 The General Failure Model

A general state diagram that describes the system and/or component behaviour is shown in Figure 3.10. An HMO (and in consequence the FMO that needs this HMO) will start in the state „errorfree“. A fault may happen on this HMO and induce a failure. The error detection may immediately detect this. If not, the error state may propagate and more parts of the HMO/system are infected. Then even more serious failure conditions may be reached. Finally the error detection will detect the propagated fault.
The dashed arrow depicts this assumption that all faults will be detected in some way and an HMO will change its state in consequence.

An HMO (in consequence the FMO) gets from a „failed“ state again to the „errorfree“ state by two possibilities. If the HMO was disabled, then maintenance is necessary: it will be repaired and enabled again. If the defense action was Restart (Reset) it will automatically return after a test to the state „errorfree“. See also Figure 3.14: about the state model of a complete system.

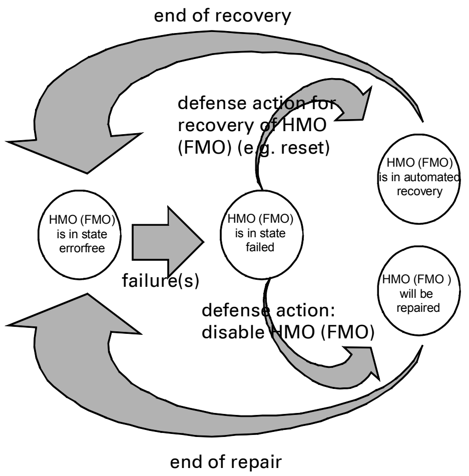

end of recovery
defense action for recovery of HMO (FMO) (e.g. reset)
HMO (FMO) is in automated recovery
HMO (FMO) is in state errorfree
HMO (FMO) is in state failed
failure(s)
HMO (FMO) will be repaired
defense action: disable HMO (FMO)
end of repair

Figure 3.10: State Diagram about Failure Behaviour of Components

In [Ech90] (see Figure 2.5:) there is a similar state diagram, but with two differences:

- Echtle’s state diagram has 16 states stemming from the fact, that there are more possibilities for recovery and error-detection. Errors may remain undetected, recovery may be wrong, so errors are not removed or not removed by the appropriate recovery. In our model nothing remains undetected forever and for the case that not the right recovery action is chosen, we assume that it will be chosen in a consequent step (at least by human interaction).
- In [Ech90] the state diagram finally ends in an unrecoverable error state in which the system rests forever. No repair or human interaction is modeled. However, in real world systems this is an important factor, too, and needs to be considered in a failure model.

3.3.2.7 OMC Alarm Logfiles

As described above, if a fault or some abnormal behaviour beside the normal operation happens, an alarm is generated. This alarm is always sent up in the hierarchy: from the component boards it is sent to the central controller (CCTRL) of the BTSE, there it is further handed on in form of an event report via Abis to the BSC and from there it finally gets to the OMC.
At the OMC a very detailed alarm log is generated about each alarm that has to be signaled to the operator. Some information in this alarm log is very important for the operator to analyze the cause and perhaps take necessary steps to eliminate it (e.g. send out maintenance personnel). The logs contain also more detailed information about the probable fault cause that can be used for a detailed analysis of software faults by the development departments for the elimination of bugs.

The behaviour of the system is monitored at the OMC. Alarm messages of the total network are visualized there and logged there. Alarmlogs of a wanted period can be exported in several details to an ASCII file, which may be used for further human or automatic processing.

Several different physical formats of logs here exist. A software solution was provided to filter out only the relevant data for the following calculations.
The following definitions show exactly which data of the logs was interesting for our analysis.

Definition 3.4: ID

ID is one out of 714 possible 16-bit values identifying the exact kind of alarm that happened.
$ID \in \{ID_1, ID_2, ID_3, \ldots ID_{714}\}$

Definition 3.5: Alarm

An Alarm is a 4-tupel. It consists of the identifiers of the BTSE where it happened, the HMO where it happened, the kind of Alarm that happened, and the Date when it happened.
$Alarm = (BTSE, HMO; ID, Date)$

Definition 3.6: Alarmlog

Alarms are sequentially ordered in an Alarmlog. We identify the order of the Alarm by the exponent.
$Alarmlog = Alarm^1, Alarm^2, Alarm^3, \ldots., Alarm^n$

To describe a component of the 4-tupel Alarm index notation is used, e.g. the BTSE of the third alarm is described by:
$Alarm^3_{BTSE}$.
The number of the last Alarm ($Alarm^n$) will be denoted with n.

Equation 3.7:

For better machine processing the BTSE and the date are mapped to natural numbers.
$BTSE \to \mathbf{N}$

$Date \to \mathbf{N}$

This mapping is carried out in the following way:
- For the conversion of the date: the earliest date appearing in the log is mapped to 1. The day after is mapped to 2 and so on.
- For the conversion of the BTSE: unique BTSE string identifiers are replaced by a unique number starting with 1.

3.4 Permissible Traffic Time (PTT)

3.4.1 Scope Permissible Traffic Time

In this chapter the new definitions of availability and unavailability are introduced. The need for new metrics was motivated in chapter 3.1. In contrast to the common definition of availability (see: Definition 2.7:) the new metrics contain not only the time aspect, but use time and the ability to handle traffic. This will be explained in the following:

The main task of a telecommunication system is to handle traffic. In the case of the BTSE this traffic are mobile phone calls originating from MSs that are in the cells this BTSE is responsible for or handling calls from the network which are destinated to such MSs. If only some TRXs or some of the cells are in an erroneous state, the BTSE still is able to fulfill a degraded service, i. e. it can handle some traffic, how much depends on the configuration (the available TRXs). For these configurations and the permissible Traffic we assume the following (confirmed by network planners):

Assumption 3.8

In the network planning and in the operation of a mobile network the configurations of a SIEMENS BTSE are planned and chosen in a way to hold a blocking probability B of 2%.

Definition 3.9: System Time

The time basis for the following calculations is system time, i.e. the system is observed only when it is expected to be operational. Planned maintenance times or preventive repair times are excluded from the system time.

With Assumption 3.8, Definition 2.14: and the number of available channels even under the constraints of failures we could plot the permissible traffic over time A(t) (see the following Figure 3.11) which represents the system behaviour.

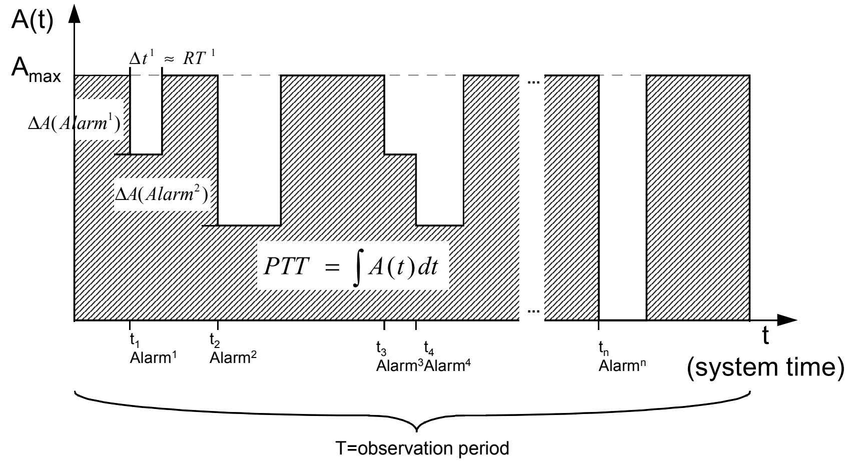


Figure 3.11: System Behaviour modeled by Permissible Traffic over Time under the Constraints of Failures

When the system is switched on the first time or just has been repaired we assume that it is able to handle its maximum of permissible traffic A_{max}. If a failure condition is reached at a certain time t_x less or even no traffic channels are available. The curve A(t) drops down for a certain level ΔA dependent on the kind of failure, which correlates to the kind of alarm (Alarm$_x$ see Definition 3.6) that is visible outside. For a certain time Δt the permissible traffic stays on this degraded level, until the system is repaired or it recovers itself (RT=Recovery/Repair time) (see Assumption 3.18).
Sometimes the Alarm just informs about a fault that does not have an influence on call processing and thus leads not to a failure. In these cases A(t) remains on the same level as before and nothing happens.
We see that the ratio of the area under A(t) to the area under A_{max} is an appropriate measure for the observed availability of the BTSE.
In the following Δt^i, RT^i and ΔA^i will be used for the above-described measures corresponding to a specific Alarm^i.

Definition 3.10: PTT

We need an expression for these areas.
With PTT (Permissible Traffic Time) we describe the area under A(t) in the time interval T.

Equation: 3.11

$$PTT = \int A(t)dt$$

A practical unit for PTT is [Erlang·h]. PTT_{max} is used to describe theoretical maximum (A_{max}·T). Finally PTT_{Loss} means the sum of the missing areas.

Note that the PTT takes degraded service modes and different sizes of BTSEs into account.
Another important characteristic of PTT is that it will increase if t increases. Since we are either interested in the relative contribution of components to PTT_{Loss} or in the relation of PTT_{Loss} to PTT_{max}, this characteristic doesn't matter.
Also useful would be the $\overline{A}$ (the mean permissible traffic), see the note at Definition 3.13 about this aspect.
This metric PTT is useful for all telecommunication systems. The time dependent function A(t) that correlates to the failures of the system is quite natural and a straightforward approach. Different systems may only imply different equations and dependencies to calculate PTT from alarmlog files. With the following equations PTT is derived in detail from the alarmlogs of the BTSE. This should be seen as an example for other systems.

3.4.2 Equations for Permissible Traffic Time

3.4.2.1 Definitions and Dependencies

In Figure 3.11 the case of multiple failures that overlap in their recovery phases is shown (Alarm³ and Alarm⁴). This case happens rarely, we now show when the simplification to „only a single failure at a time“ is possible. To calculate the PTT sum of all failures that happened, the calculation is much easier as long as the failures are independent. For the following derivations of the PTT formulas this simplification is essentially necessary.

This simplification is possible, if the following assumption holds: the sum of all failure condition times is significantly shorter than the total observation time, which can also be deduced from the PTT results:

Assumption 3.12

$$\sum_{i=1}^{n} \Delta t^i << T$$

or

$$PTT_{Loss} << PTT_{max}$$

We don't show the validity of this assumption right now, since this would require some of the results of the analysis, but it is known that the BTSE has a very good availability and duration of failures are short in comparison to T. It is also estimated that PTT_{Loss} is about 1-2 % of PTT_{max}. So the simplification is possible.

Definition 3.13: Availability

Availability is the area under A(t) (PTT) divided by the maximum possible PTT (PTT_{max}).

$$Availability = \frac{PTT}{PTT_{max}}$$

Note: Another possibility, which leads to same results, would be to define the availability on the basis of permissible traffic by using the mean of A(t) ($\overline{A}$):

$$Availability = \frac{\overline{A}}{A_{max}}$$

However, the PTT is more practical to calculate contributions of several types of alarms, because for each occurred alarm it is possible to calculate the PTT_{Loss} it results in.

Definition 3.14: Unavailability

$$\begin{aligned} Unavailability &= 1 - Availability \\ &= \frac{PTT_{Loss}}{PTT_{\max}} \end{aligned}$$

Equations 3.15

$$PTT = PTT_{max} - PTT_{Loss}$$
$$PTT_{max} = A_{max} \cdot T$$

Equation 3.16

$$\begin{aligned} PTT = \int A(t)dt &= \sum_t A(t)\Delta t \\ &= \sum_t (A_{max} - \Delta A(Alarm))\Delta t \\ &= A_{max} \cdot T - \sum_t \Delta A(Alarm)\Delta t \end{aligned}$$

The period of degraded service Δt constitutes out of several components: the error detection time, the time for defense action selection and the recovery or repair time needed after the defense action has been chosen.

Definition 3.17: Recovery/Repair Time

The term RT will be used for the time needed for either automated recovery or for the TTR (Time to Repair) that will be necessary to repair the component manually after $Alarm^i$ happened. Further analysis on these times is done in chapter 3.4.2.3.

Assumption 3.18

The Recovery/Repair time is the major component of Δt: Recovery is in the order of seconds, Repair in the order of hours, whereas error detection and defense action selection happen in ms.

With this assumption we can simplify: $\Delta t \approx RT$.

And we derive

Equation 3.19: (with Equation 3.16 and Assumption 3.18)

$$PTT_{Loss} = \sum_t \Delta A(Alarm)\Delta t \approx \sum_{i=1}^{nrAlarms} \Delta A(Alarm^i) \cdot RT^i$$

3.4.2.2 Lost Permissible Traffic

For derivation of the PTT$_{Loss}$ formula the general idea is shown in Figure 3.12. Four dependencies are given through either tables, graphs or measurements. These dependencies are named f$_1$- f$_4$ in the following. The scheme also shows some of the intermediate results which will be explained in detail below.

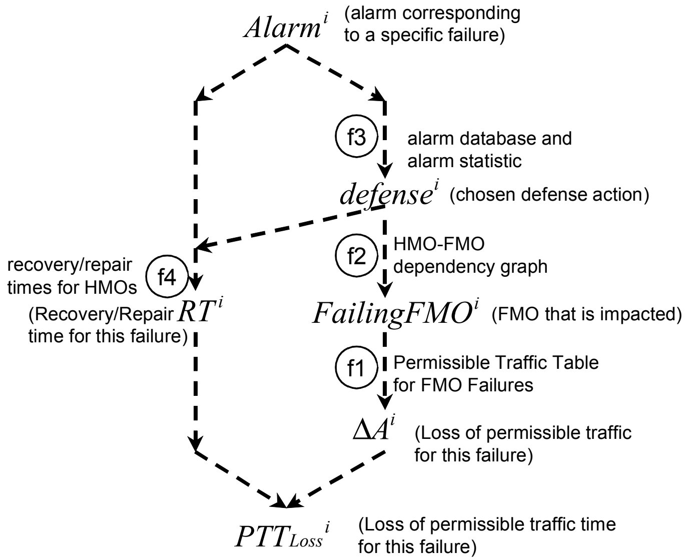

$Alarm^i$
(alarm corresponding to a specific failure)
f3
alarm database and alarm statistic
$defense^i$
(chosen defense action)
f2
HMO-FMO dependency graph
recovery/repair times for HMOs
f4
(Recovery/Repair time for this failure)
RT^i
$FailingFMO^i$
(FMO that is impacted)
f1
Permissible Traffic Table for FMO Failures
ΔA^i
(Loss of permissible traffic for this failure)
$PTT_{Loss}{}^i$
(Loss of permissible traffic time for this failure)

Figure 3.12: Derivation Scheme for the PTT$_{Loss}$ Formula

The lost permissible traffic in a failure condition (notified by an alarm) is described by the following dependency f_1:

Equation 3.20

$$\Delta A(Alarm^i) = f_1(FailingFMO^i, Alarm^i_{BTSE})$$

This dependency f_1 is calculated in the following Table 3.5. It shows all possible configurations and the traffic they are normally able to serve at 2 % blocking, and the traffic they can serve in the failure cases:
- one FMO TRX failed,
- one FMO CELL failed
- or the FMO BTSE failed.

The values have been calculated with the number of traffic channels (see Table 3.2: about the traffic channel/signalling channel aspect) that are available after the failure and the reverse approximation of the Erlang Formula (see Definition 2.14:).

BTSE is of Config-uration	Traffic Chan-nels	A_{normal} (Erl)	TCH-TRX Failure	$A_{TRX\ Failure}$ (Erl)	TCH-BTS Failure	$A_{CELL\ Failure}$ (Erl)	TCH-BTSM Failure	$A_{BTSM\ Failure}$ (Erl)	ΔA		
									f1(TRX,..	f1(BTS,..	f1(BTSM,..)
(1/0/0)	7	2,935	0	0	0	0	0	0	2,935	2,935	2,935
(2/0/0)	14	8,200	7	2,935	0	0	0	0	5,265	8,2	8,2
(1/1/0)	14	8,200	7	2,935	7	2,935	0	0	5,265	5,265	8,2
(1/1/1)	21	14,036	14	8,200	14	8,200	0	0	5,836	5,836	14,036
(4/0/0)	30	21,932	22	14,896	0	0	0	0	7,036	21,932	21,932
(2/2/0)	28	20,150	21	14,036	14	8,200	0	0	6,114	11,95	20,15
(2/2/2)	42	32,836	35	26,435	28	20,150	0	0	6,401	12,686	32,836
(4/4/4*)	90	78,306	82	70,607	60	49,644	46	36,534	7,699	28,662	41,772
Average BTSE (World Sample) $\overline{A}_{XXXFail}$		18,167		12,605		9,183		1,059	5,562	8,984	17,108

Table 3.5: Permissible Traffic after Failures for Different Configurations (f_1)

The following Equation 3.21 describes the way the average value for the last line of Table 3.5 was calculated using the probabilities for different configurations (see Table 3.1)

Equation 3.21

$$\overline{A}_{XXXFailure} = \sum_{i=BTSTypei}^{nrBTSEtypes} p_{BTSEtype} \cdot A_{XXXFailureBTSEType} \qquad XXX \text{ can be } normal/TRX/BTS/BTSM$$

$$BTSEType \text{ can be } (1/0/0), (1,1,0), (2/2/2)....$$
$$\text{according to the used BTSE mix}$$

Equation 3.22

$$FailingFMO^i = f_2(Alarm^i_{HMO}, defense^i)$$

If the $defense^i = Nothing$ then the $FailingFMO^i = NULL$ (for a fault that doesn't lead to a failure). As already described above for the defense actions $Restart$ and $Disable$ the dependency f_2 is given by HMO-FMO dependency graph (Figure 3.8).

As described in chapter 3.3.2.2 and 3.3.2.3 the dependency (named f_3 in the following) for the selection of a defense action depends on the alarm database and the counters held in the BTSE:

Equation 3.23: Selection of a Defense Action

$$defense^i = f_3(Alarm^i_{ID}, statcounter(Alarm^i_{BTSE}, Alarm^i_{ID}, Alarm^i_{Date}))$$

Equation 3.24:

The number of same alarms on the same day and machine (statistic counters) that already happened is calculated by:

$$statcounter(Alarm^i_{BTSE}, Alarm^i_{ID}, Alarm^i_{Date}) = \sum_{k=1}^{i} \begin{cases} 1 & \text{if } (Alarm^k_{BTSE} = Alarm^i_{BTSE} \text{ and} \\ & Alarm^k_{ID} = Alarm^i_{ID} \text{ and} \\ & Alarm^k_{Date} = Alarm^i_{Date}) \\ 0 & else \end{cases}$$

3.4.2.3 Recovery and Repair Time

As given by the following equation the recovery or repair time for a specific alarm depends on the HMO that failed at this alarm, and it also depends on the kind of defense action that is chosen.

Equation 3.25: Determination of Recovery/Repair Times

$$RT^i = f_4(Alarm^i_{HMO}, defense^i)$$

3.4.2.4 *Automated Recovery*

In the case of automated recovery (which happens after the Defense Action Reset) recovery times of the BTSE after failure of some HMOs are measured with real working BTSEs. The recovery time was measured on real working BTSE test equipment. The measurements have been made by manually triggering the reset of the boards. These durations should be nearly identical with the recovery times, because the time for detection and analysis of recovery can be neglected in comparison to the durations below (see Assumption 3.18).

	BS 20-(2/0/0) (sec)	BS 60-(2/2/2) (sec)	avg. BTSE**
BBSIG		17 sec	17 sec
CCLK	180 sec	195 sec	186 sec
CCTRL	180 sec	195 sec	186 sec
GPSU		480 sec	Estimation: 471 sec
LI	90 sec	100 sec	94 sec
TPU		(new/old) 17-55 sec	Estimation: 40 sec

Table 3.6: Measured Recovery Times of HMOs by Configuration

** The calculation of the average BTSE takes the probability for the BTSE configuration into account. As visible in the table the recovery of some HMOs on a 2/0/0 BTSE is faster than on a 2/2/2 BTSE. The recovery time for these HMOs depends on the number of TRXs installed (additional time is needed for recovery of each TRX). In these cases the times for a theoretical average BTSEs have been estimated by a linear approach using the average number of TRXs installed in the world.

3.4.2.5 *Repair Time*

One of the operators who uses SIEMENS equipment provided very good statistics about Repair Times. The time of every repair case of a total network was measured for about 6 months. The measured time was taken from the beginning of a visualized alarm until the failed system was repaired and fully operational again.
The MTTR of all cases was 4.39 h.

The recovery or repair time that results after a specific failure depends on the defense action chosen and on the HMO. Different boards have different recovery times, but in case of a repair, which must follow the defense action $Disable$, always the same MTTR is assumed.

So we can give the dependency f_4 in the following table

	RT	
HMO	**f4(HMO, Restart)**	**f4(HMO, Disable)**
ACDC	- *	4.39 h
ACOM	- *	4.39 h
ALCO	- **	-**
BBSIG	17 sec	4.39 h
CCLK	186 sec	4.39 h
CCTRL	186 sec	4.39 h
GPSU	471 sec	4.39 h
LI	94 sec	4.39 h
NULL	- **	- **
PA		4.39 h
RXMUCO	-*	4.39 h
RXAMOD	-*	4.39 h
TPU	40 sec	4.39 h

Table 3.7: Recovery/Repair Time Dependent on HMO and Defense Action

* Non-processor boards, no Restarts possible
** HMO has no influence on Call Processing

3.4.2.6 *PTT_{Loss} Equations*

We derive the ErlangDownTime of an Alarmlog:

Equation 3.26: (with Equation 3.19: and Definition 3.6)

$$PTT_{LossAlarmlog} = \sum_{i=1}^{nrAlarms} \Delta A(Alarm^{i}) \cdot RT^{i}$$

This result must then be adjusted to one year and one BTSE to compare different alarmlogs.

Equation 3.27: (with Equation 3.7 and Equation 3.26:)

$$PTT_{Loss}(\text{per year and BTSE}) = \frac{365 \cdot PTT_{LossAlarmlog}}{\max(Date) \cdot \max(BTSE)}$$

Now we are able to derive PTT_{Loss}:

Equation 3.28: (with Equation 3.20, Equation 3.22, Equation 3.23, Equation 3.25 and Equation 3.27:)

$$\text{(per year and BTSE)}:$$

$$PTT_{Loss} = \frac{365 \cdot \sum_{i=1}^{nrAlarms} f_1(f_2(Alarm^i_{HMO}, f_3(...)), Alarm^i_{BTSE}) \cdot f_4(Alarm^i_{HMO}, f_3(...))}{\max(Date) \cdot \max(BTSE)}$$

For the Fault Tolerance improvement we are not only interested in this combined metric. We want to know the reason for the downtime, i.e. we are interested in contributions each component (HMO, or FMO) or in the contribution each alarm type does to this metric.

As for each Alarm we know the ID and the HMO to calculate these contributions.

We get the contribution of one HMO by:

$$PTT_{Losshmo} = \frac{\sum_{i=1}^{nrAlarms} \begin{cases} \Delta A(Alarm^i) \cdot RT^i & \text{if } Alarm^i_{HMO}{=}hmo \\ 0 & \text{else} \end{cases}}{PTT_{LossAlarmlog}} \cdot 100\%$$

We get the contribution of one Alarm ID by:

$$PTT_{Lossid} = \frac{\sum_{i=1}^{nrAkarms} \begin{cases} \Delta A(Alarm^i) \cdot RT^i & \text{if } Alarm^i_{ID}{=}id \\ 0 & \text{else} \end{cases}}{PTT_{LossAlarmlog}} \cdot 100\%$$

If we define classes of Alarms, which may be result out of different probable causes, e.g. software errors, equipment errors, etc., similar contributions can be evaluated.

3.4.3 Blocking - the Related Metric

The loss of PTT could also be seen from another point of view: Users recognize this loss only in a rise of the blocking probability. For each A(t) level a corresponding blocking probability level B(t) exists (see Figure 3.13:). The correspondence of both curves is non-linear and follows the Erlang formulas.

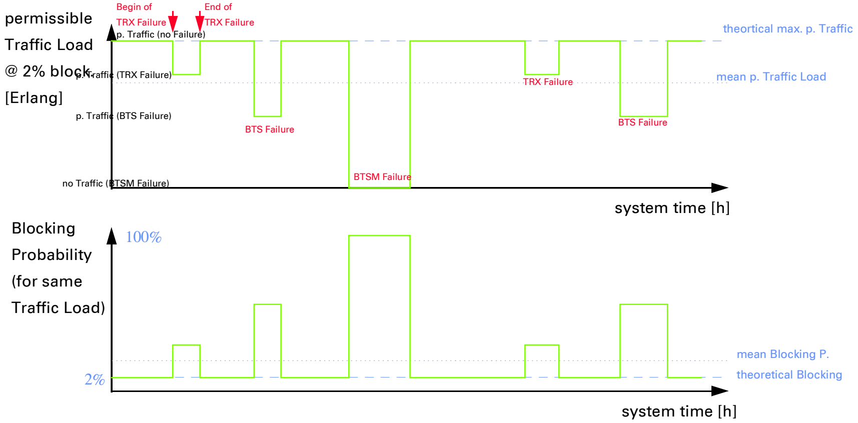


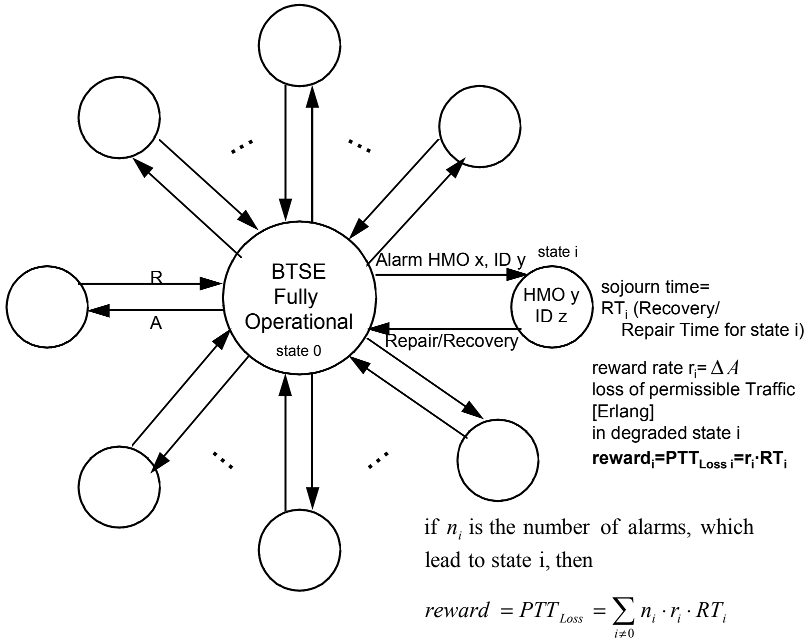
Figure 3.13: PTT and Dependency to the Blocking Probability

This blocking probability would only represent the blocking of a single BTSE. The overlapping of cells in the network changes the blocking as seen by a user significantly. The aspect of blocking in the network is considered with the third metric (PURE) below.

3.4.4 Comparison with a Markov Reward Model

In chapter 2.1.4.3 Markov reward models have been introduced. Since the model for PTT almost matches a Markov reward model in the following it is discussed in what way our model is similar to Markov reward models and how it differs from them. So this chapter is not used for further derivations, it only shows up how our PTT model is interconnected to the State of the Art.

The reward rate we use could be seen as the loss of permissible traffic ΔA in [Erlang]. The following state model (Figure 3.14:) illustrates this:



if n_i is the number of alarms, which lead to state i, then

$$reward = PTT_{Loss} = \sum_{i \neq 0} n_i \cdot r_i \cdot RT_i$$

Figure 3.14: State Model for PTT Calculation

The real used state space is huge, because every BTSE in the network may have several FMOs that consist of several HMOs, where each HMO may encounter up to 714 different alarms.

In contrast to a Markov reward model several simplifications have been made:

- Single failure assumption, only one failure is modeled at the same time. So there exist no state transitions from one failure state to another. Each failure is detected by exactly one alarm.
- The sojourn time in a failure state is deterministic, in the calculations one absolute (mean) value is used for every RT$_i$. In a Markov reward model those times are negative exponentially distributed.
- The transition rates to a failure state and back are identical, because it is assumed that for each failure a recovery or repair will happen.
- The Markov property (state transitions depend only from the knowledge of the last state) of the described system is not true, because whenever a distinct number of the same alarms already happened another defense action may be chosen and the sojourn time in state$_i$ may be longer (Restart <-> Repair).

3.5 Premature Release Probability (PRP)

3.5.1 PRP Scope

Reliability of a telephone system (if we put aside speech quality) can only be observed by the users through a measure of unwanted call releases. For this purpose the ITU has defined a metric named premature release probability (PRP) and it is useful for all kind of telecommunication systems. In fact the calculation of this metric is different for different systems. In this chapter firstly some examples for PRP are given; then a formula is derived to estimate the current PRP of the example system BTSE which can also be done with the use of the alarm logs.

For a call processing system this PRP will be a much better indicator of reliability, because the formula will be dependent on call processing not on a system MTBF. Frequent faults in parts of the system that will have a small influence on call processing, are not as important, as others that will release all calls.

If a FMO fails, where calls reside at the moment of the failure, they will be released. An immediate handover to another TRX, cell or BTSE seems not be possible, because handover is always initiated from the original owner, which cannot do this anymore.

3.5.2 Example 1: TRX Failure, no Signalling Channels Fail

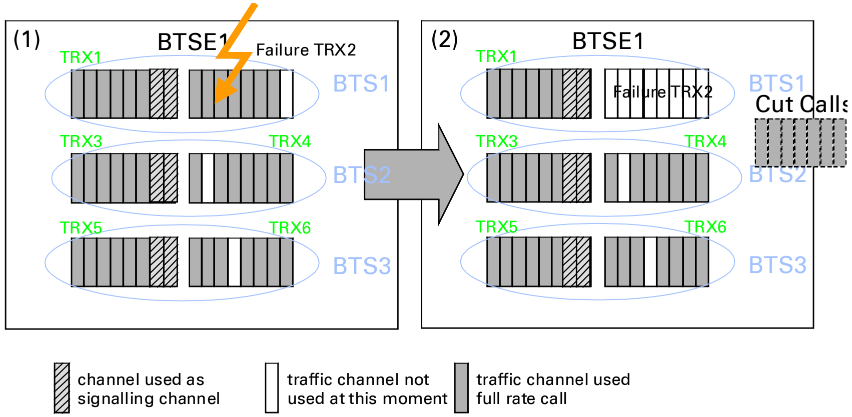


Figure 3.15: Example of FMO TRX-Failure without signalling channels and the impact on premature release

(1) A Failure of a FMO TRX happens.
(2) All calls on this TRX are lost.

3.5.3 Example 2: TRX Failure, with Signalling Channels

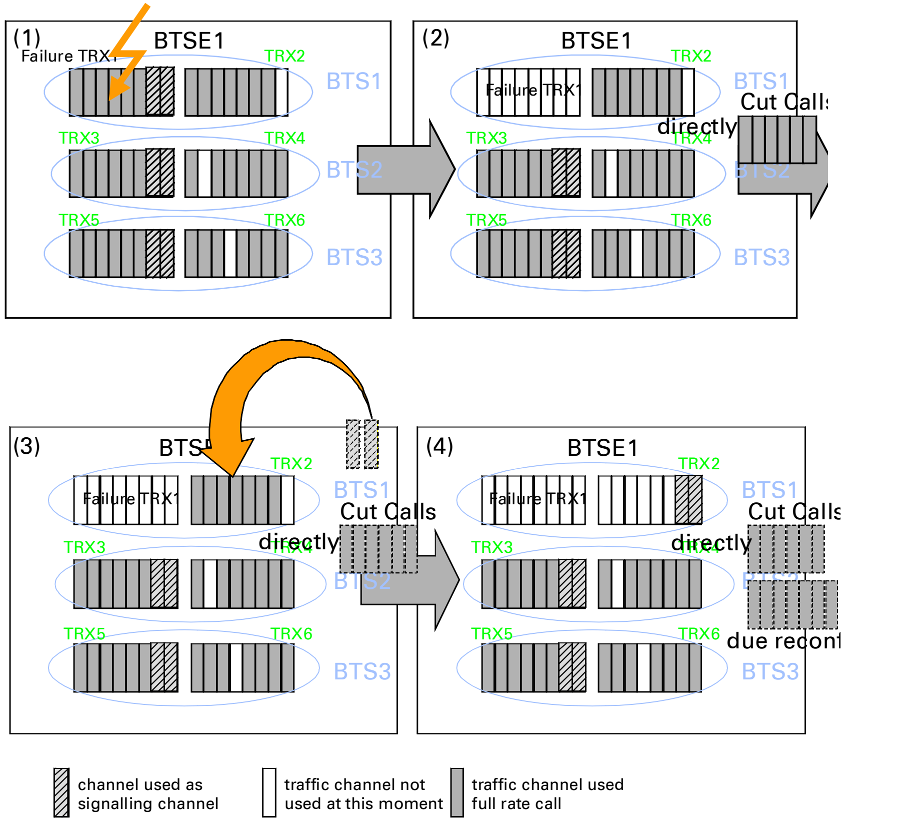


Figure 3.16: Example of FMO TRX-Failure with Signalling Channels and the Impact on Premature Release

(1) A Failure of a TRX, which also holds signalling channels (like the BCCH, or the SDCCH), happens. Without these vital signalling channels the whole cell cannot be kept alive.
(2) By the failure 6 calls are released prematurely. This happens immediately and directly.
(3) The management software realizes the missing of the signalling channels. A reconfiguration has to take place, to set up the cell again. There is a second TRX for the cell available, so the signalling channels can be put on this TRX.
(4) A reconfiguration also implies a restart of the software, so all calls on this TRX are lost, too. So there are seven calls lost due to the reconfiguration.

3.5.4 Example 2: CELL (BTS) Failure

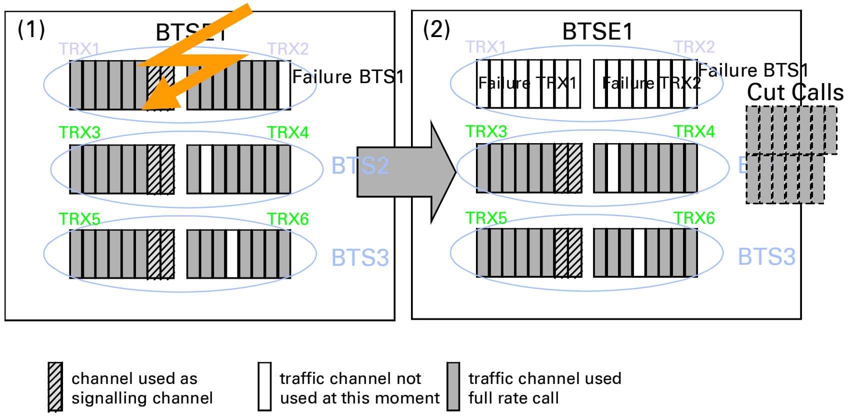

(1)
BTSE1
TRX1
TRX2
Failure BTS1
TRX3
TRX4
BTS2
TRX5
TRX6
BTS3
(2)
BTSE1
Failure TRX1
Failure TRX2
Failure BTS1
Cut Calls
channel used as signalling channel
traffic channel not used at this moment
traffic channel used full rate call

Figure 3.17: Example of FMO CELL (BTS)-Failure and the Impact on Premature Release

(1) a failure condition of a FMO CELL (BTS) is reached.
(2) All calls in this cell are lost.

3.5.5 PRP Equations

For an observed Time period we can define the PRP as the number of observed prematurely released calls through the number of handled calls.

Definition 3.29: PRP

$$PRP = \frac{FailedCalls}{Handledcalls}$$

For the further development of the formula we must make some assumptions:

Assumption 3.30

The average served traffic intensity on the BTSE is calculated like that for the PTT not with real traffic measurement data, but with network planning assumptions leading to the estimated size of the BTSE.
So the failed calls depend on
- The average served traffic intensity y (see Definition 2.15) on BTSE, CELL or TRX.
- The probabilities (or numbers) of failures for the FMOs.
We set up the dependency f_5 for the failed calls as:

Equation 3.31: Failed Calls (with Assumption 3.30)

$$FailedCalls = f_5(\#BTSM_{fail}, \#CELL_{fail}, \#TRX_{fail}, y(BTSE), y(CELL), y(TRX))$$

The handled calls would only depend on the average traffic on the whole BTSE. Since we use the y(BTSE) instead, we need to modify this value at least with the already calculated availability (in times of unavailability no traffic could be handled).

Equation 3.32: Handled Calls

$$HandledCalls = f_6(y(BTSE), Availability)$$

The number of traffic channels depends on the number of TRX dedicated to each cell. 8 full rate speech channels can be established per TRX (minus the necessary signalling channels).

Definition 3.33: c and t_i

Let t_i be defined as the number of TRX of cell i and let c be defined as the number of cells in a particular BTSE.

The following Figure 3.18 illustrates the general case.

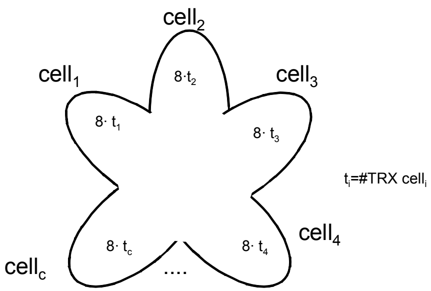


Figure 3.18: Distribution of Traffic Channels on a BTSE

For derivation of the yearly average PRP formula, three simplifications to the major problem are made:

Assumptions 3.34

1. With exception of the following point (2) signalling channels are not considered to derive the formulas. It will be shown that the served traffic intensity will have no influence and eliminate in the final formula. Under this constraint we can neglect errors made by this assumption.
2. The influence of a reconfiguration of the signalling channels cannot be neglected, because reconfigurations occur, if a permanent single failure of a TRX that holds the signalling channels for the cell happens. The reconfiguration will prematurely release all calls of another TRX in the cell (if there is one).
3. Symmetrical built configurations of BTSE are assumed, in practice this is true for about 95 % of the configurations (see chapter 3.3.1.3). With this assumption the traffic intensities y(CELL) and y(TRX) can be given as parts of y(BTSE).

Equation 3.35

$$y(CELL) \approx \frac{1}{c} \cdot y(BTSE)$$

Equation 3.36

$$y(TRX) \approx \frac{1}{c \cdot t} \cdot y(BTSE) \quad (\text{with } t = t_1 ... t_c)$$

Like for the PTT we use the single failure assumption (or more exactly the Assumption 3.12) and we can derive f_5:

Equation 3.37: (with Assumption 3.12, Equation 3.31, Assumptions 3.34)

$$FailedCalls \approx \#BTSM_{fail} \cdot y(BTSE) + \#CELL_{fail} \cdot y(CELL) + (1 + p_{reconfig}) \cdot \#TRX_{fail} \cdot y(TRX$$

Definition 3.38

T_{call} is the mean duration of a mobile call.
In several traffic models for mobile communications a mean duration of 120 seconds (see also chapter 7.2 and [BF97] on this topic)

Assumption 3.39: T_{call}

$$T_{call} = 120\,\text{sec}$$

Equation 3.40: (with Equation 3.32 and Definition 3.38)

$$HandledCalls \approx y(BTSE) \cdot Availability \cdot \frac{T}{T_{call}}$$

Equation 3.41: (with Equation 3.35 and Equation 3.36, Equation 3.37: and Equation 3.40:)

$$PRP = \frac{\#BTSM_{fail} \cdot y(BTSE) + \#CELL_{fail} \cdot \frac{1}{c} \cdot y(BTSE) + \#TRX_{fail} \cdot \frac{(1 + p_{reconfig})}{c \cdot t} \cdot y(BTSE)}{y(BTSE) \cdot Availability \cdot \frac{T}{T_{call}}}$$

3.5.5.1 Reconfiguration of Signalling Channels

As outlined in the Assumptions 3.34 on a permanent failure of a TRX, which also holds the signalling channels (in the case that this is not the only one TRX in this cell), a reconfiguration of the signalling channels occurs. This reconfiguration will prematurely release all calls of the substitute TRX.

If we have more than one TRX in a cell and a symmetrical BTSE, then the probability that a signalling channels is the TRX that fails is $\frac{1}{t}$. Further we need the probability that a TRX failure is a permanent failure ($p_{permTRX}$).

So we derive:

Equation 3.42

$$p_{reconfig} = \begin{cases} p_{permTRX} \cdot \frac{1}{t} & \text{if t} > 1 \\ & else \\ 0 & \end{cases}$$

Finally for a specific or average BTSE the PRP formula is further developed to:

Equation 3.43: PRP

$$PRP = \frac{T_{call} \cdot (\# BTSM_{fail} \cdot k1 + \# CELL_{fail} \cdot k2 + \# TRX_{fail} \cdot k3(p_{permTRX}))}{Availability \cdot T}$$

Where the constants k_1 to k_3 are given in the following table:

Configuration	*t*	c	t_{tot}	*k1*	*k2*	*k3(57.65%)*
1/0/0	1	1	1	1,0000	1,0000	1,0000
2/0/0	2	1	2	1,0000	1,0000	0,6441
1/1/0	1	2	2	1,0000	0,5000	0,5000
1/1/1	1	3	3	1,0000	0,3333	0,3333
4/0/0	4	1	4	1,0000	1,0000	0,2860
2/2/0	2	2	4	1,0000	0,5000	0,3220
2/2/2	2	3	6	1,0000	0,3333	0,2147
4/4/4*	4	3	12	0,5000	0,3333	0,0953
average BTSE **				0,9865	0,5815	0,4514

Table 3.8: Constants for the PRP Formula

* Very large SIEMENS BTSE in reality are made up by two independent BTSE installed at the same location, the values are adjusted accordingly.
** Weighted by their occurrence in the world sample

For a given Alarmlog the occurrences of FMO-Failures are counted as follows:

Equation 3.44: (with Equation 3.22)

$$\# BTSM_{fail} = \sum_{i=1}^{n} \begin{cases} 1 & if\ FailingFMO_i = f_2(Alarm^i, defense^i) = BTSM \\ 0 & else \end{cases}$$

Equation 3.45: (with Equation 3.22)

$$\# CELL_{fail} = \sum_{i=1}^{n} \begin{cases} 1 & if\ f_2(Alarm^i, defense^i) = CELL \\ 0 & else \end{cases}$$

Equation 3.46: (with Equation 3.22)

$$\# TRX_{fail} = \sum_{i=1}^{n} \begin{cases} 1 & if\ f_2(Alarm^i, defense^i) = TRX \\ 0 & else \end{cases}$$

Like for the PTT for the fault tolerance evaluations we are interested in the contributions each HMO, ID, class, etc. has to the PRP. We get these contributions, if we strictly count the failures that are caused by this HMO, ID, class, etc. and calculate the percentage to the whole PTT.
For example:

Equation 3.47: (with Equation 3.44:)

$$\# BTSM_{failhmo} = \sum_{i=1}^{n} \begin{cases} 1 & f_2(Alarm^i_{HMO}, defense^i) = BTSM \text{ and } HMO = hmo \\ 0 & else \end{cases}$$

... so the contribution is:

Equation 3.48: (with Equation 3.43, Equation 3.47: and ...)

$$PRP_{hmo} = \frac{T_{call} \cdot (\# BTSM_{failhmo} \cdot k1 + \# CELL_{failhmo} \cdot k2 + \# TRX_{failhmo} \cdot k3(p_{permTRX}))}{Availability \cdot T \cdot PRP_{Alarmlog}} \cdot 100\%$$

Similar formulas for the contributions of IDs or the classes etc. can be set up.

3.6 Probability of Unsuccessful Re-establishment (PURE)

3.6.1 Scope

PTT shows up only availability aspects of a single BTSE. However, for the user it is more interesting to know the availability aspect of the network, e.g. if the primary BTSE, where a user is connected to fails, will he still be able to re-establish a new call to another BTSE or not. Mobile radio networks bring an inherent kind of redundancy with them through the heavy use of overlapping cells structures in the network plan. The following metric tries to take this redundancy into account:
PURE will be the metric for the probability that if a call is released, it could not be re-established on the same BTSE, nor in the network. So the PURE is a special measure of availability from the user point of view. The results of the PURE metric will be needed to evaluate the influence of overlapping cells on the availability of the network in the case of failures. Note that there is no influence on reliability, since a premature release of calls will not be avoided by the use of alternate cells in the network.
PURE is difficult to determine globally. It depends highly on the kind of network scenario we consider. In urban scenarios many alternate BTSEs in case of BTSM failure may be reached. In a typical rural scenario no alternate or just one alternate may exist. But also the combined movement of traffic to alternates may cause problems. This can only be evaluated by simulation. In this chapter assumptions for the simulation are given. The simulation results for PURE are presented in chapter 5.5.

Definition 3.49: PURE

PURE is a conditional probability: under the condition that a call is prematurely released it represents the probability that this call cannot be re-established.

The PURE metric cannot be adapted to all other telecommunications systems, although it may be useful for systems based on cell-oriented network structures.

3.6.2 Example for the Unsuccessful Re-establishments

In Figure 3.19: we see a regular hexagonal cell structure with (2/2/2) BTSEs. The real cell sizes at a certain higher receiving level in dBm are marked for BTSE and some neighbor cells.

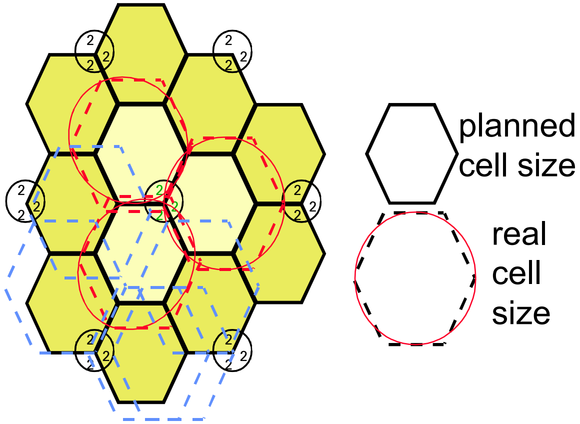


Figure 3.19: Example of (2/2/2) Base Station and its Neighbors with Planned and Real Cell Sizes

Figure 3.20: shows a cell failure scenario, where all the calls of mobile stations that are established to this cell will be prematurely released. The users manually try to establish their call again and will be connected to the cell that provides the best broadcast channel (assuming the failing cell does not have a BCCH any more). Other cells cannot be chosen by the mobile stations, unless a new technique called „directed retry“ is used in the network. This technique allows to negotiate the cells, until one is found that has a free traffic channel.

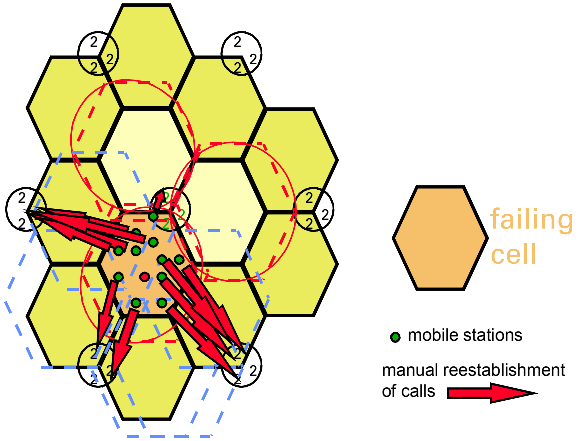


Figure 3.20: Assignment of Mobile Stations after a Cell Failure, to the „Strongest“ Neighbor Cell.

In Figure 3.21: depicts the assignment of calls to alternate cells as shown by the example. The alternate cells already are stressed by some traffic, so they may not always be able to take over more calls.

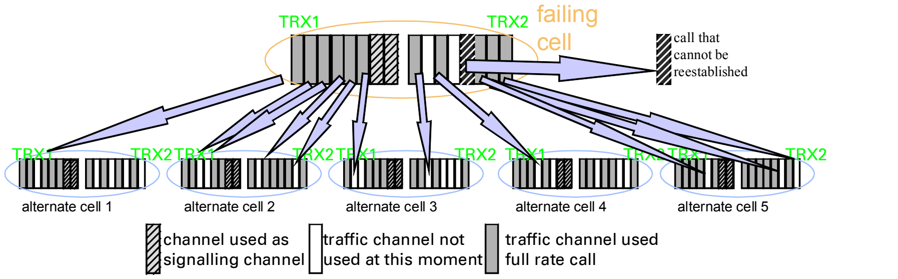

TRX1
TRX2
failing cell
call that cannot be reestablished
TRX1
TRX2
TRX1
TRX2
TRX1
TRX2
TRX1
TRX2
TRX1
TRX2
alternate cell 1
alternate cell 2
alternate cell 3
alternate cell 4
alternate cell 5
channel used as signalling channel
traffic channel not used at this moment
traffic channel used full rate call

Figure 3.21: Assignment of Released Calls to Alternate Cells and TRX at the Re-establishment.

3.6.3 Multiple Coverage

To determine to how many alternate cells traffic should be assigned, we need to know statistics about how many cells or respectively BTSEs are visible by the mobile stations. The derivation of such statistics for a mixed rural/urban scenario is presented by the author in [Keh98].

In Figure 3.22 the multiple cell coverage of a 50 km x 50 km area is shown. The x- and y-axis are specify the coordinates on the map (900 x 900). The height of the values on the z-axis gives the number of base stations that can be received (the predicted receiving level must better than -95 dBm) on that point of the map. The statistically not significant border area has been removed and probabilities for multiple coverage are determined.

In Figure 3.23 the histogram that exactly N base stations (=cells) are receivable at a certain point is given for all points of the map. The map and the number of receivable cells is shown in Figure 3.22 Note that for plotting the 3D graphic an averaging of the z-values on a 15 by 15 grid was necessary. The resulting z-values in Figure 3.22 are therefore a bit lower than the values in Figure 3.23. The highest values (up to 70 cells) could be observed in the middle of the map. This has two reasons:

- in the middle of this scenario lie two large European cities, whereas the border is more rural.
- Though for the improvement of the statistic about 10km around the border of the map was removed, there still is a possibility that distant BTS from the outside of the investigated map area might have also an influence on the investigated remaining map.

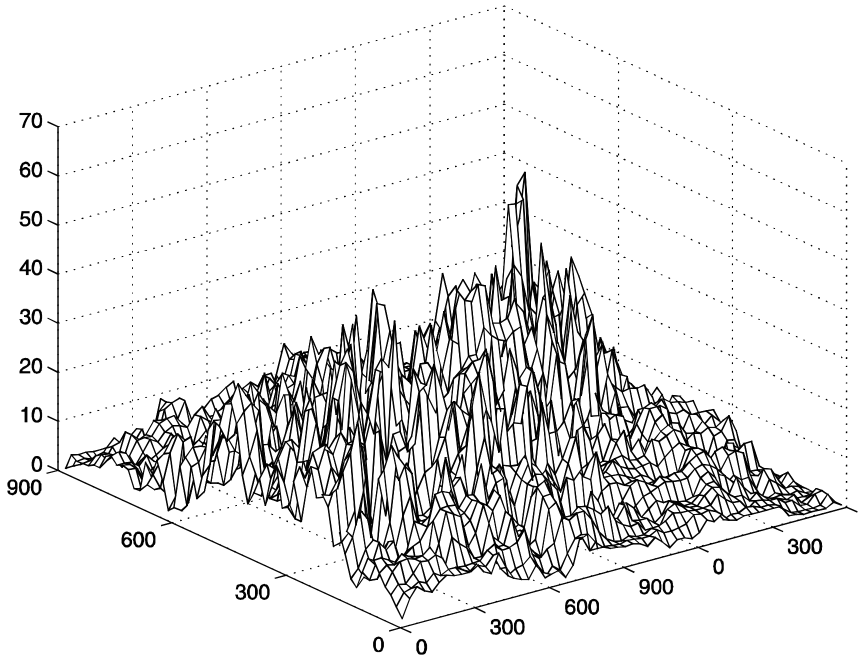


Figure 3.22: Multiple Cell Coverage Prediction for a Mixed Rural/Urban Scenario

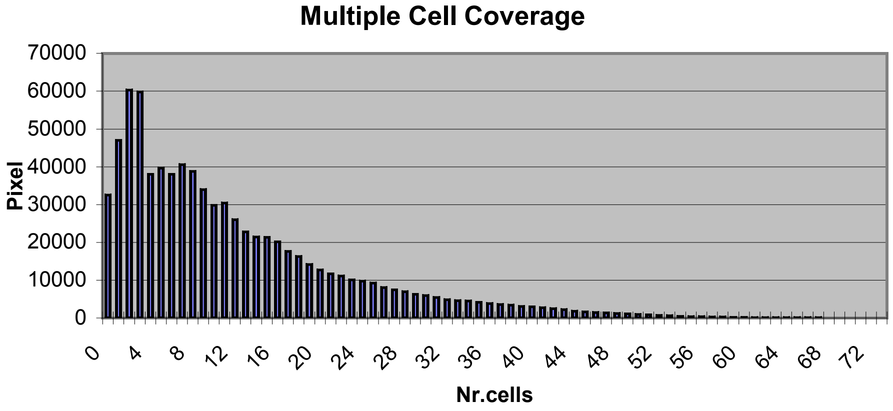


Figure 3.23: Histogram for the Multiple Cell Coverage for this Scenario

In a similar way also the Multiple BTSE Coverage is determined (see Figure 3.24:) If we compare the histograms in Figure 3.23: and Figure 3.24: it could be observed that the numbers of the Multiple BTSE (site) coverage are much lower than numbers of the multiple cell coverage (about a factor of 2). There a two reasons for this:

- A site consists of up to three cells, these cells overlap (at -95 dBm) each other on large areas.
- There are much more cells than sites which could overlap each other.

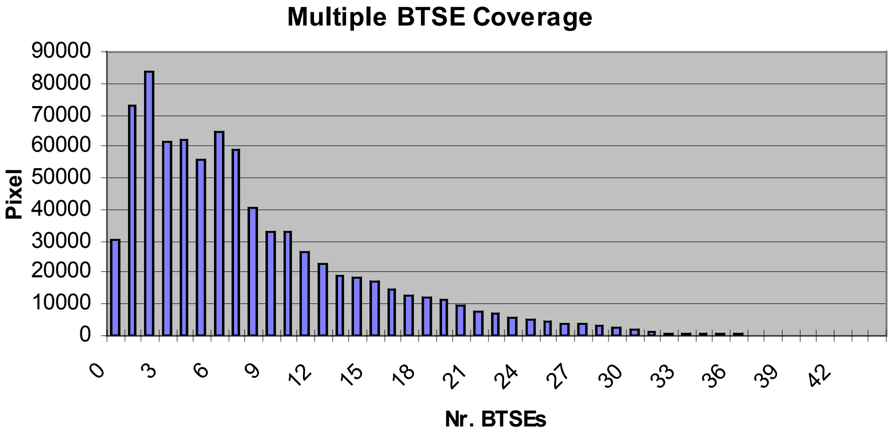


Figure 3.24: Histogram for the Multiple BTSE Coverage of this Scenario

3.6.4 Simulation for PURE

3.6.4.1 Assumptions

- Every user tries to re-establish his call again.
- The probability is determined by many (>100000) random experiments, these are enough to approximate the first few digits of the PURE.
- Traffic at the moment of failure is randomly selected for each experiment under the constraint of an average Erlang Blocking probability of 2%. (refer to the example and formula below).
- The „assignment“ to alternate cells follows the determined multiple cell/BTSE coverage probabilities as given above. That means
  - a) the scenario above is considered representative and
  - b) the assignment of several MSs to neighbor cells can be assumed identical to how many cells one MSs “sees”, if we make many random experiments.

3.6.4.2 Description of the Simulation

1. The type of Failure is randomly determined (BTSM; CELL, TRX), the distribution of this is given by the determined FMO failure rates (see the following table).

pBTSMFailure	pCELLFailure	pTRXFailure
0,9392	0,0489	0,0119

Table 3.9: Percentage of FMO-Failures for Premature Release

2. How many BTSEs have to be modeled is determined by the Multiple Cell or BTSE Coverage.
3. In each experiment the sizes of the affected BTSEs is determined again, by selecting them according to the world sample.
4. Traffic at the moment of failure on the failing BTSE and on the modeled neighbor BTSEs is randomly selected for each experiment under the constraint of an average Erlang Blocking probability of 2%.
5. For each random experiment it is only necessary to judge the results for one call: This call will move with a bunch of other calls to distinct neighbor BTSE. There the number of free channels has to be determined (see 3. and 4.) and the probability for the considered call can be calculated. In GSM systems without the feature “Directed Retry” only one possible alternate site (the one with the strongest BCCH) can be chosen. So if this alternate site has no free traffic channel the call is denied.

6. The PURE results in

Equation 3.50

$$PURE = \frac{\text{considered calls lost}}{\text{random experiments}}$$

7. With the number of random experiments PURE converges to a distinct number.

For a random "booking" of a BTSE the general Erlang formula Equation 2.17 is used.
For results of these simulations refer to chapter 5.5.

4 Alarm Evaluation Tool

The immense amount of data in even small alarm log files and the constant need to evaluate new areas of different operators made clear that an automatic way was needed to evaluate such log files. Based on the models presented a tool was programmed for evaluating PTT and PRP metrics and contributions of HMOs, FMOs, etc. out of given alarm logs. The tool now represents an easy and efficient way to explore gathered data in several ways and identify the major problems. This tools consists of about 50.000 lines of C++ and perl code, a manual evaluation of the data would have been impossible.

This chapter will now shortly describe the real implementation of this tool and in which steps the evaluation is done. Figure 4.1 shows the main steps that are done by the alarm evaluation tool.

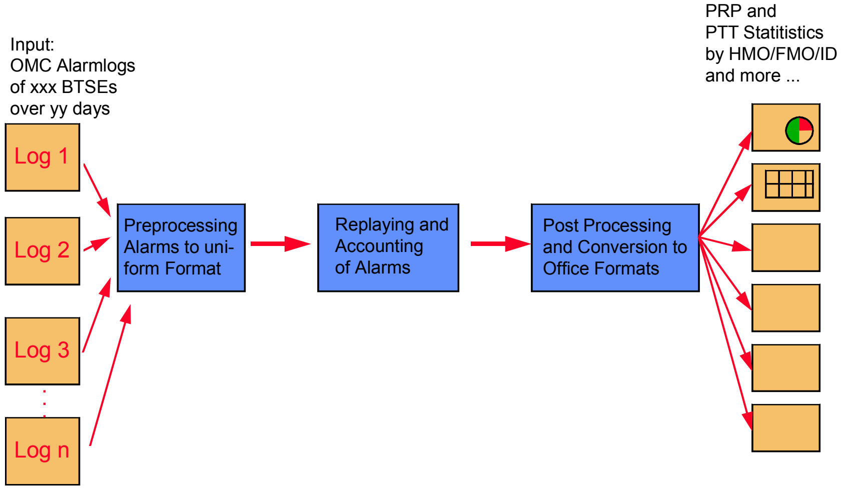

Input:
OMC Alarmlogs
of xxx BTSEs
over yy days
Log 1
Log 2
Log 3
Log n
Preprocessing Alarms to uniform Format
Replaying and Accounting of Alarms
Post Processing and Conversion to Office Formats
PRP and
PTT Statitistics
by HMO/FMO/ID
and more ...

Figure 4.1: The Alarm Evaluation Tool

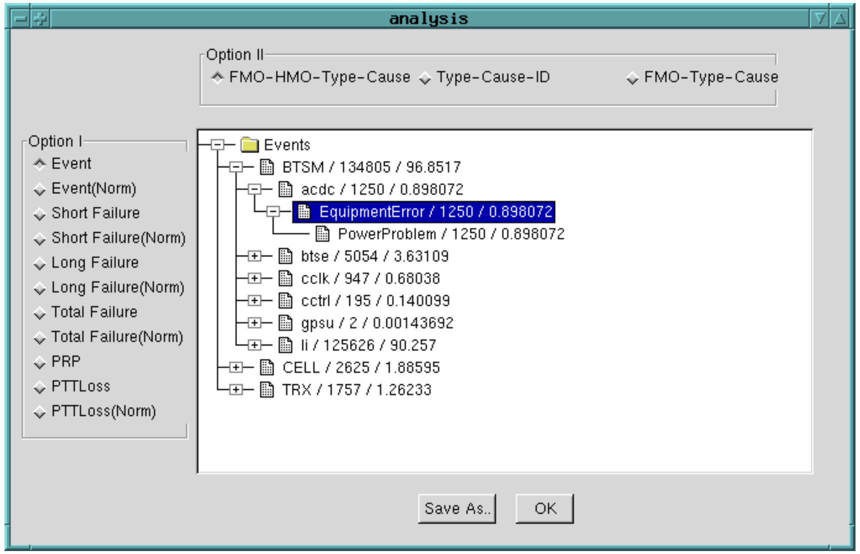
Beside the visible inputs (the Alarm Logs) the tool uses several parameters that may be changed:

- configurations or sizes of BTSEs,
- MTTR and recovery times,
- Behaviour at each different alarm,
- system structure,
- redundancy schemes.

These parameters can be changed interactively with a graphical user interface. Sets of parameters can be organized in projects.

In the first step the Alarm Logs were processed and converted to a uniform Format.
The main step „Replaying and Accounting of Alarms“ consists of the following:

- Alarms are read in one by one and evaluated.
- Therefore states (714 counters of each HMO) of the necessary number of BTSEs are held in memory. The number of BTSEs and number of days (T) is taken from a header of the alarm logfile.
- According to the above-derived formulas and the states, the PRP and PTT contribution of this alarm is calculated and accounted on the HMO, FMO and ID.
- Several reasonableness checks are done while accounting:
  - The HMO given in the alarm log is checked for validity, if the reported HMO does not match the HMOs that are possible for this alarm a right HMO is chosen randomly. This may happen sometimes because of incorrect alarm logging.
  - When the defense action is chosen, it is checked if the particular HMO is already disabled.
- When the MTTR is over, disabled boards will be enabled again (states are changed)



Option II
◆ FMO-HMO-Type-Cause ◇ Type-Cause-ID ◇ FMO-Type-Cause

Option I
◆ Event
◇ Event(Norm)
◇ Short Failure
◇ Short Failure(Norm)
◇ Long Failure
◇ Long Failure(Norm)
◇ Total Failure
◇ Total Failure(Norm)
◇ PRP
◇ PTTLoss
◇ PTTLoss(Norm)

```
Events
  BTSM / 134805 / 96.8517
    acdc / 1250 / 0.898072
      EquipmentError / 1250 / 0.898072
        PowerProblem / 1250 / 0.898072
    btse / 5054 / 3.63109
    cclk / 947 / 0.68038
    cctrl / 195 / 0.140099
    gpsu / 2 / 0.00143692
    li / 125626 / 90.257
  CELL / 2625 / 1.88595
  TRX / 1757 / 1.26233
```

Figure 4.2: Screenshot of the Tool's Result Browser

In a last step the calculated outputs (the PRP and PTT statistics) are displayed hierarchically in a tree. The results for this metrics may be viewed in several ways selectable by the structure of the tree. (see Figure 4.2). Finally the results maybe written to a file and post-processed for an easy adaptation to office tools.

5 Results

In this chapter the results of PTT, PRP and PURE evaluations are described. Several different alarm logs from different operators have been used.

5.1 The Analyzed Alarmlogs

13 different alarm logs have been analyzed. Each of the logs was recorded over different time periods and in different networks. All logs are taken from networks installed in European countries.
Statistically some are more representative, because of larger periods or larger network sizes.

Under the assumption that the BTSE configurations used in the networks are not too different from each other, mean values may be given for the results of all the samples. Since the log files differ in statistical relevance it was necessary to calculate weights to derive means.

These weights are calculated by:

$$weight_j = \frac{duration_j \cdot nrBTSE_j}{\sum_{i=1}^{n} duration_i \cdot nrBTSE_i}$$

a mean value then is calculated by:

$$mean = \sum_{i=1}^{n} weight_i \cdot value_i$$

Log	1	2	3	4	5	6	7	8	9	10	11	12	13
no. BTSEs	271	327	405	206	170	130	43	10	18	5	420	80	135
duration	22	51	51	56	1	1	8	26	6	18	8	7	28
weight	9.37%	26.21%	32.46%	18.13%	0.27%	0.20%	0.54%	0.41%	0.17%	0.14%	5.28%	0.88%	5.94%
date	Aug 97	Oct 97	Oct 97	Sep 97	Aug 98	Aug 98	Oct 98	Oct 98	Oct 98	Nov 98	Nov 98	Nov 98	Nov 98

Table 5.1: Analyzed Alarmlogs

5.2 Validation of Statistical Significance

5.2.1 Classifying Failures and Evaluating Dependency

We used the above described Evaluation Tool just to do a simple evaluation of the input data (alarmlogs): Count how many short- or long-term failure conditions are reached for each HMO: The two classes long term or short term are defined by a certain limit t=1h. Every failure that has a duration shorter than 1 hour is counted as short-term failure (mostly software or transient hardware failures).

Every failure resulting in a downtime above that limit would be counted as long-term failure (mostly hardware failures that need repair). These statistics for each of the above-described input samples (alarmlogs) are shown in Table 5.2 and Table 5.3:.

Note that the classification is not identical to a hard- and software failure classification, since the threshold mechanisms for software failures could transform a short-term failure into a long-term failure when the threshold is reached and the HMO is disabled.

HMO\Log	1	2	3	4	5	6	7	8	9	10	11	12	13
Acdc	10	8	23	0	0	0	0	0	0	0	0	0	0
Actam	29	143	55	14	1	2	1	1	2	0	23	1	12
Bbsig	38	49	25	40	1	0	2	0	0	0	631	0	2
Cclk	3	0	1	0	0	0	0	0	0	0	0	0	0
Cctrl	0	0	0	0	0	0	0	0	0	0	0	0	0
Gpsu	1	1	5	2	0	1	0	0	0	0	22	0	0
Li	0	0	0	0	0	0	0	0	0	0	0	0	0
Pa	15	24	25	11	1	0	1	0	0	0	103	0	30
Rxamd	0	5	0	162	0	0	0	0	0	0	0	30	0
Rxmuc	0	4	0	159	0	0	1	0	0	0	43	27	1
Tpu	52	63	51	74	2	1	11	0	0	0	162	0	49
Txamd	0	1	0	0	0	0	0	0	0	0	0	0	0

Table 5.2: Number of Long-Term Failures in the Different Alarm Logs

HMO\Log	1	2	3	4	5	6	7	8	9	10	11	12	13
Acdc	0	0	0	0	0	0	0	0	0	0	0	0	0
Actam	0	0	0	0	0	0	0	0	0	0	0	0	0
Bbsig	82	79	59	58	1	0	3	0	0	0	37	0	0
Cclk	0	1	1	0	0	0	0	0	0	0	0	0	0
Cctrl	0	0	20	1	0	0	0	0	0	0	0	0	0
Gpsu	0	0	0	0	0	0	0	0	0	0	0	0	0
Li	1632	3691	3119	2	1877	441	214	82	11	10	21099	4369	2666
Pa	13	91	30	20	5	0	0	0	0	0	45	0	42
Rxamd	0	0	0	1	0	0	0	0	0	0	0	0	0
Rxmuc	11	4	0	8	0	0	0	0	0	0	208	0	31
Tpu	42	47	41	2	0	0	0	0	0	1	53	2	20
Txamd	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5.3: Number of Short-Term Failures in the Different Alarm Logs

Logs that observe a longer period of time or a larger network (more BTSEs) should reveal higher failure numbers in the long term or short-term category. Since the BTSEs can be considered as independent machines in the network, where the failure conditions and repair is also assumed to be statistically independent, we can assume a linear relationship: double the amount of BTSEs will produce twice as many failures. The same applies for the time dependency of the alarm logs: double the duration of the log should cause about twice as many failures.

If the duration of the logs is long enough or if enough BTSEs are observed, the points of each sample should not vary much from the regression line through the origin. This regression line should go through origin, since if we have observed 0 BTSEs or the log time is 0, we couldn't have observed any failure.

Figure 5.1 shows the scatter diagram we obtain by plotting Table 6.2 over the duration multiplied with the number of observed BTSEs (derived from Table 5.1:). For each HMO the regression lines can be calculated, as an example the regression lines for long-term tpu failures and short-term bbsig failures are also displayed in Figure 5.1 and Figure 5.2. Figure 5.2 shows the same diagram for the Short Term Failures. Nearly all points of log 11 deviate a lot from the regression lines. We suspect that either the given network size of log 11 is incorrect or that there existed major problems in the start-up phase of this network. The data set of this log has been removed from the figures and isn't considered in the following analysis.

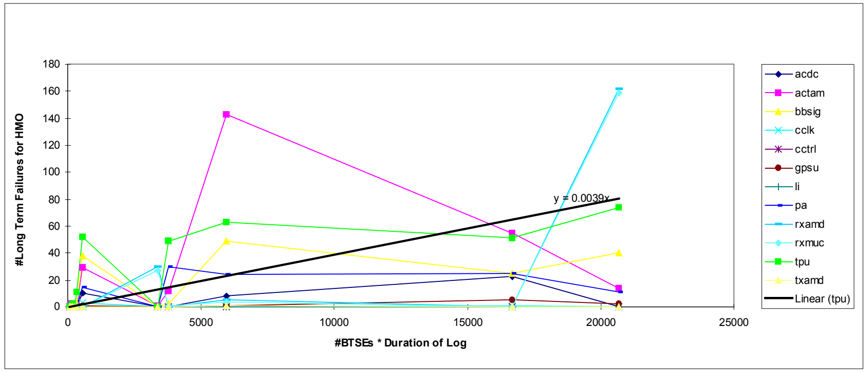


Figure 5.1: Long Term Failures over Log Duration Multiplied with Number of BTSEs

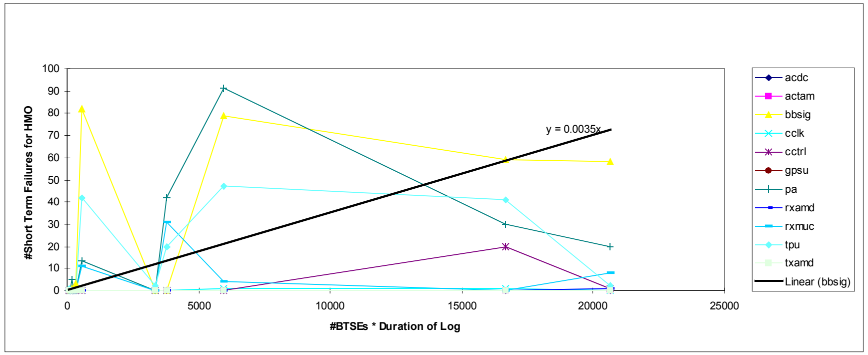


Figure 5.2: Short Term Failures over Log Duration Multiplied with Number of BTSEs

As already visible in the Figures the points are scattered a lot in the diagram and they deviate much from the regression lines. This means that the observed logs are too short and the statistical significance is low or the observed networks (configurations and used hardware variants) are too different to combine them in one statistic. Unfortunately, longer logs are hard to obtain from the operators of mobile networks, either this data is considered as confidential or the operators won't spend the effort to gather the complete data of a network for about 2-3 months or longer. There also exists a data storage problem since the raw alarm log would exceed several hundreds of Mbytes. The presented method is, nevertheless, still valid and we just want to express a warning about the confidence that could be stated in the results presented below.

5.2.2 Linear Regression on the Classified Data

The formulas of simple linear regression can be found in almost every standard book about statistics (see [All90], [GT96], [Tri92], [Rao65]). The special case of regression through the origin is covered by [All90] and in detail by [MoPe92]. Statistical computer software like Wolfram Research's Mathematica or even Microsoft's Excel support the calculation of this special case.

[MoPe92] give the following formulas for regression through the origin:
The least squares estimator of the slope is

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} y_i \cdot x_i}{\sum_{i=1}^{n} x_i^2}$$

Equation 5.1: Estimation of the Slope

In our case x is the number of BTSEs multiplied with the duration of the log, and y is the number of failures of the component.

The regression line is then calculated by:

$$\hat{y} = \hat{\beta}_1 \cdot x$$

Equation 5.2: Regression Line Through the Origin

so we can estimate a wanted response at x=x$_0$ by

$$\hat{y}_0 = \hat{\beta}_1 \cdot x_0$$

Equation 5.3: Estimation of Response at x$_0$

The error mean square (or residual mean square) MS$_E$ for the origin model is:

$$MS_E = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n-1} = \frac{\sum_{i=1}^{n} y_i^2 - \hat{\beta}_1 \sum_{i=1}^{n} y_i x_i}{n-1}$$

Equation 5.4: Residual Mean Square for Origin Model

The 100(1-α) confidence interval on E(y|x$_0$) at x=x$_0$ is given by:

$$\hat{y}_0 - t_{\alpha/2,n-1}\sqrt{\frac{x_o \cdot MS_E}{\sum_{i=1}^{n} x_i^2}} \le E(y \mid x_0) \le \hat{y}_0 + t_{\alpha/2,n-1}\sqrt{\frac{x_o \cdot MS_E}{\sum_{i=1}^{n} x_i^2}}$$

Equation 5.5: 100(1-α) Confidence Interval on the Mean Response at x=x$_0$

A generally used confidence interval is 95 percent. The value $t_{n-1,2.5}$ of the student t distribution can be looked up in statistical tables for example in [All90] or [MoPe92]. For our calculations the number of samples (logs) n is 12. So $t_{11,2.5}$ = 2.20098627.

In the following Equation 5.1 was used to estimate the slope for the short term and long term failure rate β_{1HMO} for each different HMO. This allows the estimation of corresponding MTBFs:

$$MTBF_{HMOShort/Long} = \frac{1}{\hat{\beta}_{1HMOShort/Long}}$$

Equation 5.6: Estimation of Long Term or Short Term MTBFs

Additionally in Table 5.4 the estimated failure counts for one year ($\hat{y}_{365}$) are calculated with Equation 5.3. The MS$_E$ is calculated using Equation 5.4 and the confidence intervals for E(y|x$_0$=365) are calculated with Equation 5.5. The two resulting values for the confidence interval were named upper and lower bound. The results of this analysis are given in the following Table 5.4. Where the result of the lower bound was below zero, since no „negative numbers of alarms“ are possible it was set to zero (this is indicated by *). Also an MTBF value was calculated, where possible (** indicates that 0 failures have been observed, so the MTBF would be infinite).

HMO	Failure Type	$\hat{\beta}_1$ [Failures / day]	$\hat{y}_{365}$ [Failures]	MS_E	lower bound C.I. 95 % [Failures]	upper bound C.I. 95 % [Failures]	MTBF [year]
acdc	Short-Term	0.000E+00	0.000	0.000	0.000	0.000	**
actam	Short-Term	0.000E+00	0.000	0.000	0.000	0.000	**
bbsig	Short-Term	4.160E-03	1.519	404.221	0.977	2.060	0.659
cclk	Short-Term	4.203E-05	0.015	0.039	0.010	0.021	65.186
cctrl	Short-Term	4.781E-04	0.174	17.999	0.060	0.289	5.731
gpsu	Short-Term	0.000E+00	0.000	0.000	0.000	0.000	**
li	Short-Term	1.674E-01	61.117	2825229	15.809	106.425	0.016
pa	Short-Term	2.933E-03	1.071	354.410	0.563	1.578	0.934
rxamd	Short-Term	1.299E-05	0.005	0.077	0*	0.012	210.949
rxmuc	Short-Term	3.848E-04	0.140	93.682	0*	0.401	7.121
tpu	Short-Term	2.230E-03	0.814	149.545	0.484	1.144	1.228
txamd	Short-Term	0.000E+00	0.000	0.000	0.000	0.000	**
acdc	Long-Term	7.522E-04	0.275	17.316	0.162	0.387	3.642
actam	Long-Term	4.393E-03	1.604	683.809	0.899	2.308	0.624
bbsig	Long-Term	2.285E-03	0.834	130.884	0.526	1.143	1.199
cclk	Long-Term	4.339E-05	0.016	0.757	0*	0.039	63.141
cctrl	Long-Term	0.000E+00	0.000	0.000	0.000	0.000	**
gpsu	Long-Term	1.679E-04	0.061	0.633	0.040	0.083	16.320
li	Long-Term	0.000E+00	0.000	0.000	0.000	0.000	**
pa	Long-Term	1.404E-03	0.512	63.518	0.298	0.727	1.952
rxamd	Long-Term	2.217E-03	0.809	2073.101	0*	2.036	1.236
rxmuc	Long-Term	2.162E-03	0.789	1988.815	0*	1.991	1.267
tpu	Long-Term	3.892E-03	1.421	347.329	0.918	1.923	0.704
txamd	Long-Term	1.878E-05	0.007	0.062	0.000	0.014	145.920

Table 5.4: Regression Results and Confidence Intervals

5.2.3 Estimating Confidence Intervals for PTT and PRP

In the last chapter we used linear regression to obtain the confidence intervals for the input samples. In the following it will be shown how an estimation for the confidence intervals of the new metrics PTT and PRP can be derived on this basis.

To achieve this, the number of alarms for each HMO and Failure Class (Short Term/Long Term) that one BTSEs might produce in 10 years operation are estimated ($\hat{y}_{3650}$). The Confidence Intervals are also calculated. Then three new alarmlog files are generated which correspond to the estimations or confidence bounds. The alarms are chosen in a way that exactly the number of wanted failures will be produced. To generate integer values for the number of alarms needed the numbers for the lower bound are rounded down, for the mean and upper file the are rounded up. Where a number of 0 was estimated since no failure of this type existed in the eleven given alarm samples, we corrected this to estimate one failure of this type in 10 years operation for the mean and upper bound file (marked with *), where such a type was sensible. Table 5.5, Table 5.6:, Table 5.7 show the used number of failures of the generated logs. The alarms for the logs have been distributed uniformly over the 10 year period.

Mean Estimation File	**Acdc**	**actam**	**bbsig**	**cclk**	**cctrl**	**gpsu**	**li**	**pa**	**rxamd**	**rxmuc**	**tpu**	**txamd**
Short Term Failures	0	0	16	1	2	0	612	11	1	2	9	0
Long Term Failures	3	17	9	1	1*	1	1*	6	9	8	15	1

Table 5.5: Properties of the Generated Alarmlog for the Mean Estimat. of Failures (1 BTSE, 10 y)

Lower Bound File	**Acdc**	**actam**	**bbsig**	**cclk**	**cctrl**	**gpsu**	**li**	**pa**	**rxamd**	**rxmuc**	**tpu**	**txamd**
Short Term Failures	0	0	9	0	0	0	158	5	0	0	4	0
Long Term Failures	1	8	5	0	0	0	0	2	0	0	9	0

Table 5.6: Properties of the Generated Alarmlog for the Lower Bound of Failures (1 BTSE, 10 y)

Upper Bound	**Acdc**	**actam**	**bbsig**	**cclk**	**cctrl**	**gpsu**	**li**	**pa**	**rxamd**	**rxmuc**	**tpu**	**txamd**
Short Term Failures	0	0	21	1	3	0	1065	16	1	5	12	0
Long Term Failures	4	24	12	1	1*	1	1*	8	21	20	20	1

Table 5.7: Properties of the Generated Alarmlog for the Upper Bound of Failures (1 BTSE, 10 y)

By using these generated alarm log files with the alarm evaluation tool the PRP and PTT metrics where calculated. This gives an estimation for the confidence intervals of these metrics and allows an estimation of the mean. The results are also presented in the next chapter together with the real results of the logfiles. Note that the confidence intervals and the estimated mean are not derived from data of log 11 in contrast to the other results.

5.3 PTT Results

5.3.1 Availability

Table 5.8 shows the results of availability as defined by Definition 3.13.

Log	1	2	3	4	5	6	7	8	9	10	11	12	13	Weighted Average
availability	99.805 %	99.853 %	99.918 %	99.669 %	99.447 %	99.569 %	99.695 %	99.954 %	99.827 %	99.996 %	98.82 %	98.793 %	99.823 %	99.770%

Table 5.8: Result of Availability Calculations for the Different Logfiles

The generated logfile for the mean estimation revealed 0.998328 and the confidence interval [0.999495, 0.997275].

5.3.2 PTT_{Loss} by FMO

To be comparable the following PTT values are scaled to one BTSE in one year.

Loss in [Erlang·h] $\text{PTT}_{\text{LossFMO}}$	Log 1	Log 2	Log 3	Log 4	Log 5	Log 6	Log 7	Log 8	Log 9	Log 10	Log 11	Log 12	Log 13	Weighted Avg.**
TRX	157.25	72.89	43.67	96.63	210.35	68.53	362.67	0.00	0.00	0.25	1056.57	0.08	191.28	135.34
CELL	70.02	132.03	38.32	417.91	84.65	221.40	83.67	55.35	266.50	0.00	283.24	1490.48	49.57	162.19
BTSM	78.60	26.33	46.66	4.81	574.53	387.33	32.37	16.41	5.30	5.78	506.19	405.96	36.70	65.35
PTT_{Loss} (Sum)	305.86	231.24	128.65	519.35	869.53	677.26	478.71	71.76	271.80	6.03	1846.00	1896.53	277.55	362.88

Table 5.9: Results of PTT_{Loss} by FMO

** Weighted by importance (duration of log and number of base stations)

Loss in [Erlang·h] $\text{PTT}_{\text{LossFMO}}$	Estimated Mean Log File	Confidence Interval Upper/Lower Log File
TRX	73.39	[39.13,97.86]
CELL	138	[31.54,260.23]
BTSM	61.55	[9.76,75.60]
PTT_{Loss} (Sum)	272.94	[80.44,433.67]

Table 5.10 Results of the Generated Log Files for PTT_{Loss} by FMO

The confidence intervals of the results given in Table 5.10 are very large, this again indicates that the analyzed log files are too short in their duration or the network sizes are too small to yield good statistical results, we will see this in most of the other presented confidence intervals, too.

The contributions of each FMO to PTT_{Loss} in percent is given by the following figure:

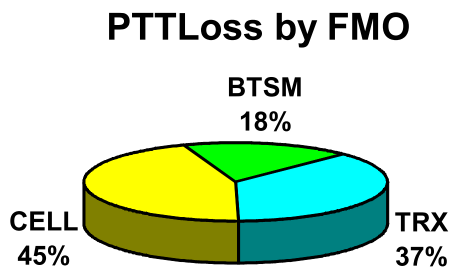


Figure 5.3: Average Contributions of the FMOs to PTT_{Loss}

There exist two methods to get confidence intervals for the contributions in percent:
- the upper/lower bound is taken for the FMO, where we want to know the percentage, and accordingly the lower/upper bounds of the other two FMOs are taken and the percentages a resolved. This yields very large intervals.
- the upper/lower bound is taken for the FMO, where we want to know the percentage and accordingly the mean values of the other two FMOs are taken and the percentages a resolved. This yields much smaller intervals.

The second method was preferred since the first one didn't provide any usable results, again a sign that the used samples are not good enough.
The contribution percentages of the estimated mean log file are: TRX: 26.89%, CELL: 50.56%, BTSM: 22.55%.

The confidence intervals calculated with the second method are:
- TRX [16.40%,32.91%]
- CELL [18.95%,65.85%]
- BTSM: [4.41%,26.34%]

5.3.3 PTT_{Loss} by HMO

The following Figure 5.4 shows the distribution of PTT_{Loss} on the different HMOs. The tpu+bbsig+pa make up the TRX contribution. Mostly the central (BTSM) contribution is caused by the li and gpsu HMOs. The acdc was put in brackets, because in the real system there exists a very good redundancy scheme (multiple acdc units plus backup batteries), which was not modeled in such detail with the tool. This means that the probability for a complete loss of all acdc units is very low, the 4.87 % should not be counted. The biggest contributions are caused by the rxamd+rxmuc+acom, (FMO BTS, cell hardware).

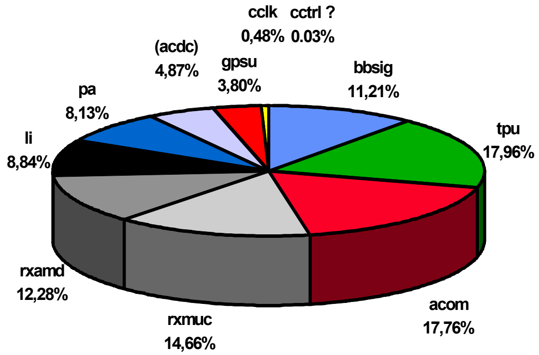


Figure 5.4: PTT_{Loss} by HMO

The following Table shows the results of the estimated mean of PTT_{Loss} for each HMO and the contributions in percent. For calculation of the upper and lower bound contributions the same method as for the FMOs was used (relation between upper/lower value to the sum of all means of the other HMOs plus the upper/lower value).

HMO	Est. Mean Log File $\text{PTT}_{\text{LossHMO}}$	Contribution to PTT_{down}	Contribution Lower Bound	Contribution Upper Bound
(acdc)	22.52	8.25%	2.91%	10.71%
actam	67.03	24.56%	13.28%	31.48%
bbsig	22.01	8.06%	4.65%	10.47%
cclk	7.60	2.78%	0.00%	2.78%
cctrl	7.68	2.82%	0.00%	2.85%
gpsu	7.51	2.75%	0.00%	2.75%
li	16.23	5.95%	0.87%	8.12%
pa	14.71	5.39%	1.87%	7.06%
rxamd	35.49	13.00%	0.00%	25.85%
rxmuc	31.55	11.56%	0.00%	24.63%
tpu	36.67	13.43%	8.52%	17.15%
txamd	3.94	1.44%	0.00%	1.44%

Table 5.11: Results of Estimated Mean and Upper/Lower Bounds for PTT_{Loss} HMO Contributions

5.3.4 PTT_{Loss} by Type

The following Figure 5.5: shows the types of PTT_{Loss} contributions of the FMOs. The chart in the upper left corner shows how the influence of different fault types is on the PTTLoss for the FMO BTSM, in the lower left corner the same chart for PTTLoss for the CELL FMOs and in the lower right corner the chart for PTTLoss of the TRX FMOs. This figure depicts how these three differnt charts of the FMOs contribute to the overall $PTT_{LossTOTAL}$ (They contribute according to Figure 5.3 which is again shown in the chart in the middle).

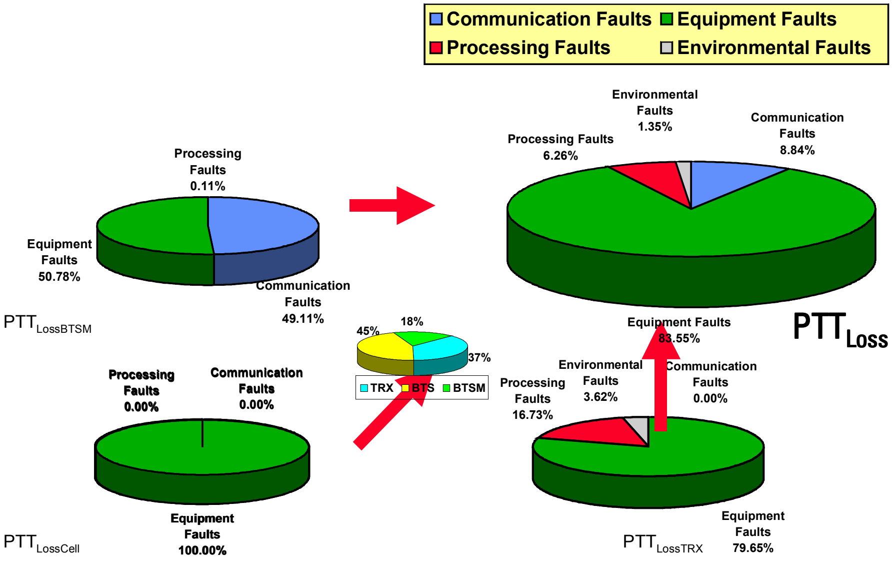


Figure 5.5: Types of PTT_{Loss} by FMO and the Contributions to TOTAL PTT_{Loss}

Some more notes about this Figure:

- Processing faults only occur in the TRX and BTSM equipment (only those have software), CELL equipment has no processors; here the only visible type is Equipment Error.
- The only environmental cause that influences directly the function of the system is an overtemperature of the power amplifier leading to a TRX-FMO failure.
- The major cause of the communication faults are line problems on the A_{bis} Interface. Those problems are supposed to be from bad weather conditions that influence the transmission on the radio relay equipment.

5.3.5 PTT_{Loss} by ID

An even more detailed view of PTT_{Loss} is possible. The contribution of every Alarm can be calculated separately (which is shown in Table 5.12). Such a calculation may be more useful for the development to hunt down the major bugs, but we could also identify the hot spots here. The top three ones are problems with the RF Cell hardware. The fourth one is loss of board, which boards are not identified in this summary. Then we can see two alarm types concerning power problems. The seventh most important one is communication problem on Abis (PCM_ALARM up). Note that no confidence intervals could be provided for this type of PTT_{Loss} contribution, in the calculation of the upper and lower bounds of failures just short term or long term failures where distinguished not different IDs. Distinguishing IDs for confidence intervals would require much longer Alarmlogs.

Alarm ID	PTT_{Loss} in [Erlang·h]
CAE_AEID_DUCOM_VSWR2_ALARM	49.754
CAE_AEID_RXAMOD_DC_LOOP_F	44.100
CAE_AEID_RXMUCO_DC_LOOP_F	42.517
CFP_AEID_BOARD_LOST	38.534
CAE_AEID_DCTPU_FAIL	36.543
CA1_AEID_PCM_ALARM_UP	32.064
CAE_AEID_GPSU_FAIL	13.781
CFP_AEID_CODE_ERROR	12.200
CAE_AEID_RXAMCO_BF	9.0463
CAE_AEID_ACDC_BATTBD_RED	9.0442
CAE_AEID_TXAMOD_VSWR2_ALARM	8.5939
CAE_AEID_ACDC_BATTBD	7.7522
CAE_AEID_DCPA_12V5VOOT	7.000
CRC_AEID_NO_PROC_ID_AFTER_RES	6.160
CBM_AEID_HW_ERROR_EXT	6.051
CPA_AEID_TRANSMITTER_FAULT	5.329
CDM_AEID_TIMEOUT_RESET_REMOTE	4.766
CRC_AEID_HW_RESET_WATCHDOG	4.644
CPA_AEID_HPA_POWER_CONTROL_LOOP	4.075
CCH_AEID_SLAVE_IS_DEAD	3.782
CPA_AEID_PA_COMB_INTERF_FAULT	3.240
CPA_AEID_HPA_OVER_TEMPERATURE	2.520
CPA_AEID_POWER_CONTROL_LOOP	2.387
CPA_AEID_OVER_TEMPERATURE	2.380
CDG_AEID_TEST_FAILED	1.400
CRC_AEID_TIMEOUT_PER_RESET	0.987
CCS_AEID_BOARD_LOST	0.700
...	...

Table 5.12: PTT_{Loss} by Alarm ID

5.4 Results for Premature Release Probability

5.4.1 Results for the Premature Release Probability

NAME	1	2	3	4	5	6	7	8	9	10	11	12	13	Weighted Average
Total PRP	4.10 E-04	3.26 E-04	2.20 E-04	3.67 E-05	1.53 E-02	4.70 E-03	8.92 E-04	4.35 E-04	1.55 E-04	1.59 E-04	9.05 E-03	1.09 E-02	1.00 E-03	8.93 E-04

Table 5.13: PRP Results for the Different Alarm Logs

About every 1200th call is prematurely released due to hardware or software problems of the BTSE. Note that this figure does not contain the contributions of premature release due to failed handovers or mobility of the user. This figure represents only the value for the BTSE network element for the case of software or hardware failures and not the value for the whole path from end-user to end-user including other network elements. The values above also do not contain the contribution of the unreliable air interface.
In the ITU Recommendation E.850 [ITU92] it is recommended that Premature Release of telephone connection should be better than 4E-04 - 8E-04 for 90 percentile of international connections. This is not a general recommendation for mobile services, but gives a good comparison of the wanted size for this measure.

The generated logfile for the mean estimation for the PRP revealed 2.534E-04 (Sample 11 is also not used here). The confidence interval [6.723E-05, 4.364E-04] was calculated.

5.4.2 Premature Release Probability by HMO

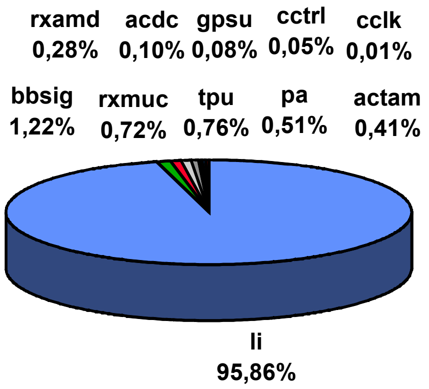


Figure 5.6: PRP by HMO Distribution

The following Table 5.14: shows the results of the estimated mean of PTT_{Loss} for each HMO and the contributions in percent. For the calculation of the upper and lower bound contributions the same method as with the PTT_{Loss} was used.

	estimated mean contribution to PRP	lower bound of contribution to PRP	upper bound of contribution to PRP
acdc:	0.445%	0.593%	0.149%
actam:	1.487%	2.088%	0.704%
bbsig:	1.663%	2.184%	0.940%
cclk:	0.297%	0.297%	0.000%
cctrl:	0.445%	0.593%	0.000%
gpsu:	0.148%	0.148%	0.000%
li:	90.950%	94.593%	72.124%
pa:	1.131%	1.589%	0.470%
rxamd:	0.875%	1.906%	0.000%
rxmuc:	0.875%	2.160%	0.000%
tpu:	1.596%	2.118%	0.872%
txamd:	0.088%	0.088%	0.000%

Table 5.14: Results for PRP Contributions of Estimated Mean and Upper/Lower Bounds for HMOs

As Figure 5.6 and Table 5.14: clearly show up the major problem is the link interface with the above named Abis link problems. Here the major difference between reliability and availability becomes evident. These link failures dominate in frequency, but do not contribute in the same way to availability! See also the following two chapters.

5.4.3 Premature Release Probability by Type

The following Figure 5.7 shows the contributions of several types of failures to the PRP. As stated in the last chapter those problems come from the link interface and are communication faults, where the communication link to the BSC is unstable.

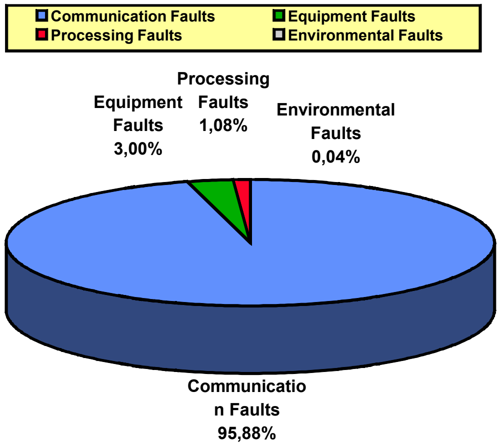


Figure 5.7: PRP by Type Distribution

5.4.4 Premature Release Probability by Alarm ID

Like for the PTT we can evaluate the PRP more in detail. The following table again gives the contribution of several alarm IDs, this time for the PRP. The most frequent one is the PCM alarm up (Abis link failure). The second is an alarm of the cell equipment. The third one a HW reset by a watchdog. The fourth the loss of a board. Again, confidence intervals haven't been calculated for the ID contributions.

Alarm ID	PRP
CA1_AEID_PCM_ALARM_UP	0.00085597
CAE_AEID_DUCOM_VSWR2_ALARM	2.80E-06
CFP_AEID_BOARD_LOST	2.64E-06
CAE_AEID_DCTPU_FAIL	2.61E-06
CAE_AEID_RXAMOD_DC_LOOP_F	2.49E-06
CAE_AEID_RXMUCO_DC_LOOP_F	2.40E-06
CAE_AEID_RXAMCO_LEFT_BF	2.13E-06
CRC_AEID_HW_RESET_WATCHDOG	1.83E-06
CBM_AEID_HW_ERROR_EXT	1.35E-06
CAE_AEID_RXAMCO_RIGHT_BF	1.25E-06
CFP_AEID_CODE_ERROR	1.25E-06
CPA_AEID_PA_COMB_INTERF_FAULT	8.09E-07
CPA_AEID_HPA_POWER_CONTROL_LOOP	7.47E-07
CAE_AEID_GPSU_FAIL	7.00E-07
CPA_AEID_TRANSMITTER_FAULT	6.16E-07
CPA_AEID_FP_MESSAGE_MISSED	5.84E-07
CDM_AEID_TIMEOUT_RESET_REMOTE	5.76E-07
CAE_AEID_RXAMCO_BF	5.18E-07
CAE_AEID_DCPA_12V5VOOT	4.94E-07
CAE_AEID_TXAMOD_VSWR2_ALARM	4.83E-07
CDG_AEID_RSP_TIMEOUT	4.80E-07
CAE_AEID_ACDC_BATTBD_RED	4.53E-07
CRC_AEID_NO_PROC_ID_AFTER_RES	4.47E-07
CCH_AEID_SLAVE_IS_DEAD	4.29E-07
CAE_AEID_ACDC_BATTBD	3.88E-07
...	...

Table 5.15: PRP by Alarm ID

5.5 PURE Simulation Results

To evaluate the benefits of fault tolerance on the network level which is realised by multiple cell coverage the Probability of Unsuccessful Re-establishment (PURE) was defined in chapter 3.6. The simulation to determine PURE (described in chapter 3.6.4) revealed the following results:

If we increase the Blocking probability of 2%, we introduce a higher traffic scenario and the probability that the re-establishment will fail is also increased. In the following Figure 5.8: results for various blocking probabilities are presented.

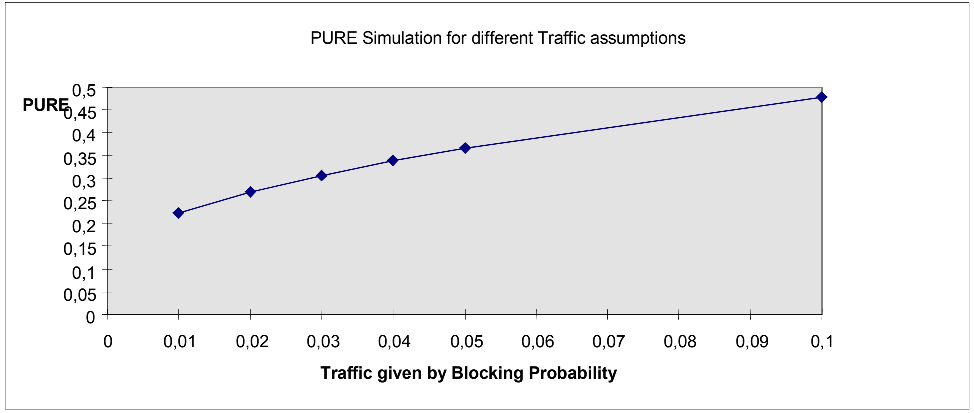


Figure 5.8: PURE for Different Blocking Probabilities

The overall PURE for the 2% blocking assumption is 26,9%. That means under the condition of FMO-Failure using the distribution of different FMO types of Table 3.9 (which was derived from the PRP results), then 26.9 % of active calls cannot be re-established, neither on the same BTSE nor in the network.

The general result of this is: even if there is a high multiple cell coverage, in case of a failure some calls will travel to the same alternate site (the strongest BCCH will guide this), which will eventually result in the fact that calls cannot be re-established.

If Directed Retry would be used, the re-establishment would be negotiated among nearly all the alternate servers. For this case, a much better PURE near the blocking probability is assumed, further simulations have to be made to evaluate this.

5.6 Identified Weak Spots of Availability and Reliability

In the last chapter 5 the results for the PRP, PTT and PURE are given. The weak spots are already visible and stated there. In this chapter a short summary is given about the results:

- For the PTT or availability the major problems lie in the TRX FMOs and CELL FMOs, only 18 % of PTT_{Loss} is contributed by the FMO BTSM and its central HMO components.
- All the components of the FMO TRX contribute to 37 % of PTT_{Loss}, therefore a complete redundancy scheme of the TRX is sensible (see chapter 6.2.2).
- The RF components of the CELL FMO could also be identified as weak spots: together they contribute to 45.4 %, but as the confidence intervals and the samples indicate the observed failures are very different among the logs. For the Cell equipment several different hardware variants exist, which may or may not be reliable. This should be investigated in a future analysis, where more equipment related data about the analyzed networks must be available. For the CELL equipment we also didn't analyze any new fault tolerance schemes in the following chapter.

- The main contributor to the unreliability in form of the PRP is the Abis link (96 %).

- Hardware Failures dominate over Software Failures by a factor of 13 in the PTT_{Loss} statistic. This is clear, since the MTTR is much higher than the average duration for automated recovery. A faster MTTR maintenance/repair would adjust this factor a bit, but from today's point of view there seems to be no need to improve the speed of the software recovery routines. The availability would not significantly change.

6 Fault Tolerance Schemes

6.1 Possible and Already used Fault Tolerance Schemes

In this chapter it will be discussed, which fault tolerance schemes are possible within a BTSE, which schemes have already been implemented and which ones are not applicable or not worthwhile at all.
In Figure 2.11 redundancy was characterized by four categories. In the following the question will be discussed, which of these redundancy schemes are already used and which may be possible or useful:

6.1.1 Time Redundancy

Since voice calls are a type of critical real-time services, the use of time redundancy is difficult. The retransmission and en-/decoding of a bad frame (user data packet) must happen within the same 20 ms period of the original frame. For GSM a retransmission of speech frames is not realized. For non real-time services as for example normal data calls, retransmission is implemented.
Another kind of time redundancy is used for the described mechanism of resetting and rebooting boards after certain failures. Established calls aren't saved with this procedure, but it is tried to bring the system in an operable state again. Non-realtime services (e.g. delivery of SMS messages) and maintenance services (e.g. Software Upgrade) are redone afterwards. Users may establish speech calls again.

6.1.2 Information Redundancy

High sophisticated coding standards and methods are used in several layers of the air interface. The maximum of available computing power is used together with the decoding algorithms to recover erroneous received data frames. For the interboard and interprocessor communication and also for the communication on Abis information redundancy is used for error detection. Information redundancy is also used for the non-volatile data and program storage (Flash EPROMs) of the BTSE, since in contrast to the volatile RAM such devices have much shorter MTBFs and the number of writes is limited. Also several software upgrades can be held in one Flash memory.

6.1.3 Functional Redundancy

Functional redundancy is not used yet. Software N-Version Programming is not convenient, the effort spent for one version is large enough and the advantages are unclear and not proven. Additionally the software failures imply the smallest amount in unreliability and unavailability. Even if NVP would be successful there would be no improvement by it.

6.1.4 Structural Redundancy

Structural redundancy is used in some areas: If more than 1 TRX is configured per cell this represents a structural redundancy. Cells that are overlapping also are a kind of structural redundancy, although dependant on the location of the mobile user. Receiver diversity is a structural redundancy scheme, which allows reception if one antenna path does not work, but which also improves decoding and reception in general. The redundancy of ACDC units with backup batteries is a kind of structural redundancy. Nevertheless, this technique still offers the biggest potential for reliability and availability improvements. Due to this fact only such techniques are described and evaluated in this dissertation.

6.2 Evaluation of new Candidate Redundancy Schemes

This chapter describes some new candidate redundancy schemes. The ideas for development of these schemes are often discussed in the BTS development departments. Up to now their performance was unclear and only estimated without a good statistical basis. In the following the techniques will be described shortly and a performance evaluation of each technique is given using the above-derived metrics PTT and PRP and the current evaluated data from the alarm logs.

6.2.1 Core Redundancy

6.2.1.1 Description

By the technique „Core Redundancy“ a hardware duplication of the central boards central clock (cclk) and core controller (cctrl) is proposed. This duplication should be realized in the following way:

- The second core should be held in a „semi-cold“ state, i.e. it will receive software updates like the primary core via Abis and additionally all configuration data should be mirrored from the primary to the secondary core.
- Only in case of a permanent loss of the primary core the secondary should take over all functions automatically. The switch over to the secondary core isn’t designed to be very fast, since a loss of all active calls will be accepted.
- The failed primary core could be replaced later, while call processing is handled by the secondary.

With this technique only availability aspects might improve. Calls will be dropped anyway so no significant rise in PRP (beside the small rise of handled calls during the MTTR) will be observed. So this technique might save as a maximum (assuming perfect implementation while not introducing more faults by this technique) all the equipment faults of the HMOs cctrl and cclk.

6.2.1.2 Evaluation

As visible in Figure 5.4 the core controller (cctrl) and central clock (cclk) HMO sum up to a 0.5% contribution to $\text{PTT}_{\text{Loss.}}$ The largest portion of this contribution is induced by Equipment Faults. The following Figure 6.1 is basically the same as Figure 5.5 (refer to it for more description) but illustrates the possible gain of the scheme.

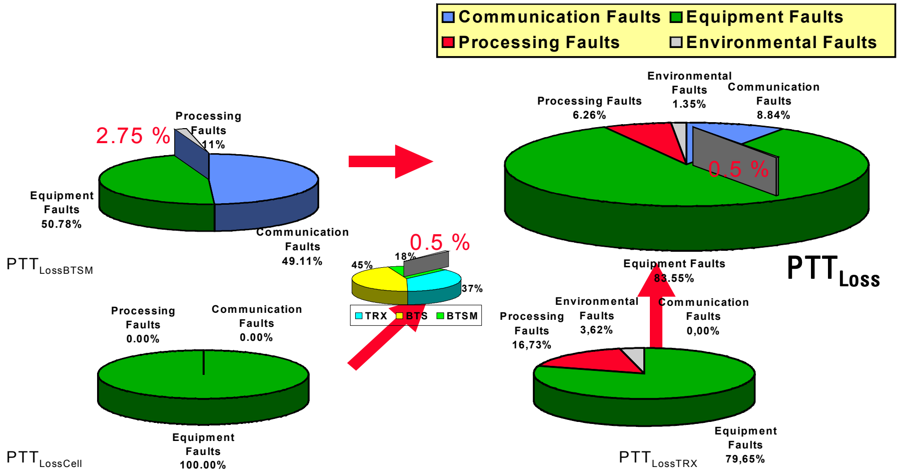


Figure 6.1: Benefits of Core Redundancy Scheme

Since 0.5 % to PTTdown is a rather small contribution, there are several other points which should be improved before a realisation of this fault tolerance scheme seems to be useful. Probably the small benefit of this scheme will be lost again by the added complexity to system (new software and hardware for core redundancy scheme).

6.2.2 Spare or Floating Transceiver

6.2.2.1 Description

With the spare or floating transceiver (TRX) another hardware redundancy scheme is proposed. Operators here should plug in one additional TRX per cell (spare TRX) or one additional TRX per site (floating TRX) that will not be used for call processing. This (these) TRX(s) will remain „cold“ until another TRX of this cell (site) fails. The TRX will then be powered on and take over. A loss of the calls of the failing TRX will be accepted, so the take over hasn’t to be fast. The failed TRX could be replaced later; the same amount of traffic could be handled anytime, even if a single TRX fails. The development and equipment costs of floating TRX seem to be very expensive, because of a very complicated cabling.
As visible in Figure 3.2: TRXs are only connected to one cell, this connection is also not a digital signal, but an analog RF signal. For a Floating TRX scheme it must be possible to connect or switch every TRX (or at the least the floating one) to any cell. The Spare TRX scheme avoids this problem.

6.2.2.2 Evaluation

As visible in Figure 5.3 the TRX contribution to PTT_{Loss} is about 37 %. 80% of this contribution is induced by Equipment Faults, some (10% estimated) of the Processing Faults would also be saved by the Floating/Spare TRX scheme, since disables after exceeding certain thresholds would not happen as often. So a proper implemented Floating or Spare TRX would be able to save a maximum of 33 % of PTT_{Loss}. The following Figure 6.2 is basically the same as Figure 5.5 (refer to it for more description) but illustrates the possible gain of the scheme.

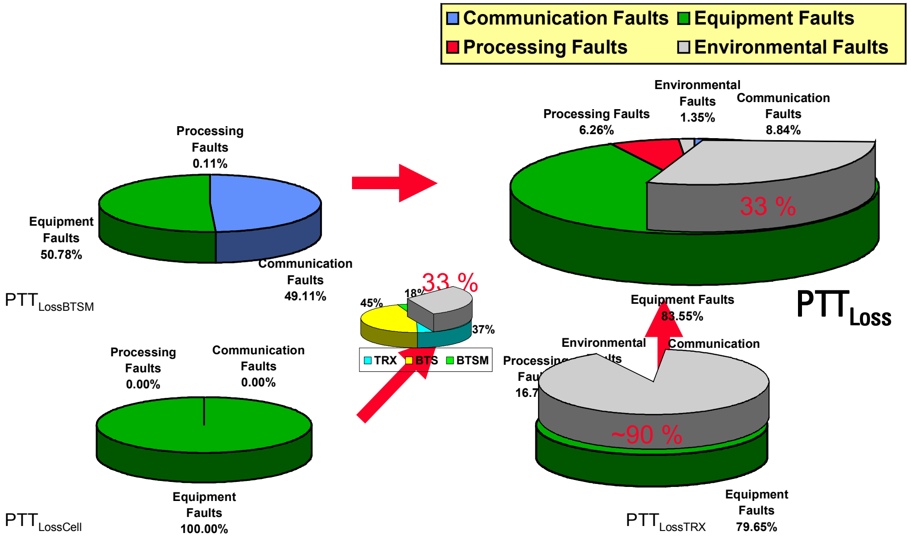


Figure 6.2: Benefits of Spare/Floating TRX Scheme

6.2.3 Abis Link Redundancy

6.2.3.1 Description

Abis (BTSE-BSC interface, see chapter 2.2.4) Link Redundancy means using two independent Abis connections to receive/send the speech or data packets from/to the network. In operation the two links could be always active and the packets would be sent via both links. If two identical packets were received, the last incoming would be discarded. Another possibility seems establishing the link only, when the other link fails. Both solutions are very expensive for the operators, since it is believed that the leased lines or radio relay channels used for the Abis connections constitute about 30 % of the costs for operating a network, but a fast switch over to the other line could even reduce premature releases. There exists a third possibility for large configurations of BTSEs (needed in the future), that need two Abis links due to amount of traffic channels. Traffic for the cells or sectors could be equally distributed on the both links. So in case of a single link failure not the whole cell becomes unavailable, we would encounter only a loss of about (traffic in Erlang is non-linear) 50 % of the traffic.

6.2.3.2 Evaluation

Depending on what scheme is used half or nearly all of the Communication faults shown in Figure 5.5: (about 4.5-8.8 %) could be saved as a maximum of $\mathrm{PTT}_{\mathrm{Loss}}$. The following Figure 6.3 is basically the same as Figure 5.5 (refer to it for more description) but illustrates the possible gain of the scheme. More dramatically are the benefits in case of fast switch over to the PRP: The PRP could be reduced by 48-96 %, since this Abis link faults are the most frequent ones in the whole BTSE (see Figure 5.7).

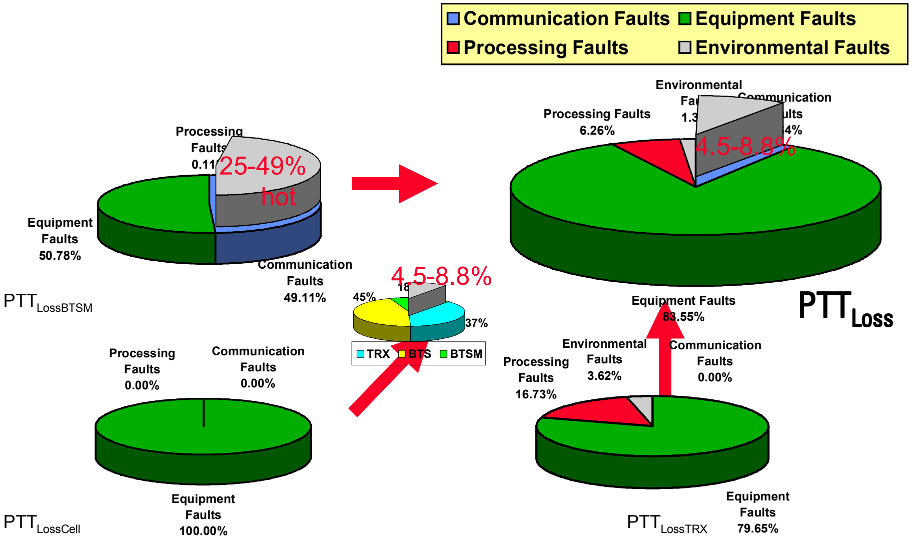


Figure 6.3: Benefits of Abis Link Redundancy Scheme

For the final recommendation on BTSE fault tolerance improvement see the following chapter 6.3.

6.3 Selecting new FT Schemes

Deciding which of the presented FT schemes should be realised is done in the following steps:

- In the first step we must decide which metric needs to be improved. Here different operators or different applications scenarios may have different priorities.
- In the next step the effort to realise the FT-schemes must be estimated.
- Finally the relative improvement of this metric needs to be divided by the effort that it is needed to realise the FT-scheme. Based on this metric - the FT-improvement-effort ratio - a ranking will be defined which FT-scheme needs to be implemented first to get with the minimum effort the maximum of FT-improvement.

Note that by repeating the whole procedure of improving, measuring, calculating the statistics and evaluating new FT-schemes again and again, we will finally have a system where no more new schemes can be found or the improvement-effort ratio to realise any new scheme will be identically low.

6.3.1 Selecting the Metric

In chapter 3.2 the need for a single criterion to select the FT-schemes accordingly was discussed. Since the analysis revealed more than one metric, it has to be decided which metric needs to be improved. If more than one metric needs to be improved weights for the importance of the metrics are needed. In the following the PRP and the $\mathrm{PTT}_{\mathrm{Loss}}$ are selected. Weights for the importance are not known therefore the final results will be two rankings.

6.3.2 Estimating Effort for the Realisation of the FT Schemes

An exact development effort estimation in man months was made in the SIEMENS AG for the described FT-schemes. Unfortunately these estimations cannot be published in this thesis, therefore in Table 6.1 only relative "efforts" based on our own estimations are given. For the further calculations and showing up the general method these estimations are sufficient.

FT-scheme	Relative Effort
Core Redundancy	100
Floating/Spare TRX	80/50
Abis-Link Redundancy	30

Table 6.1: Relative Effort of the Presented FT-Schemes

6.3.3 Calculating a Ranking by the Benefit - Effort Ratio

The final metric to determine the best or next candidate scheme for FT-improvement is a benefit - effort ratio:

Definition 6.1: Benefit-Effort Ratio (BER)

$$BER = \frac{\textit{relative}\ \text{benefit FT metric in \%}}{\textit{effort}}$$

In the following Table 6.2 the BER was calculated and the resulting ranking is given:

FT-Scheme	Relative FT-improvement		Relative Effort	BER		Ranking	
	PTT$_{Loss}$	PRP		With PTT	With PRP	With PTT	With PRP
Core Redundancy	0.5%	-	100	5*10E-5	0	3.	
Floating/ Spare TRX	37%	-	80/ 50	4.6*10E-3/ 7.4*10E-3	0	1.	
Abis Link Redundancy	4.5-8.8%	48-96%	30	1.5*10E-3-2.9*10E-3	1.6*10E-2-3.2*10E-2	2.	1.

Table 6.2: Ranking and Calculated Benefit - Effort Ratio

6.3.4 Recommendation on the Presented FT Schemes

Deciding which of the presented FT schemes should be proposed now is easy:

- From the given PTT and PRP figures and also the effort estimation semi-cold core redundancy isn't worth it (Rank 3 PTT, no improvement/rank for PRP).
- Abis link redundancy would be very useful to improve the reliability (rank 1). The availability could also be improved a bit (Rank 2). The implementation seems to be feasible, only the high operating fixed costs for an additional line are a problem. Here the operator must decide in what cases this FT scheme should be used.
- TRX redundancy would be a good candidate to improve availability (rank 1). It seems to be possible to implement the spare TRX scheme. Additional fixed costs for the additional TRX equipment have to be paid by the operators.

The unavailability problems of CELL equipment have to be investigated in further studies. Here another potential for a possible FT strategy may be found.

7 Discussion

In this chapter the presented results are discussed under various aspects which have not been considered yet. In chapter 7.1 the importance of the BTSE in the general network context is discussed. Chapter 7.2 compares the results to the unreliability of the air interface. In Chapter 7.3 an extrapolation to the behaviour in the future networks is made and the future impact on the metrics is discussed. Chapter 6.3 finally summarizes which FT-strategies are suitable and should be selected.

7.1 Importance of the BTSE NE in the Network

As already mentioned in the GSM introduction (see chapter 2.2) the BTSE is not the only network element in the path from end user to end user.

We are now at the point to discuss the worth of possible improvements in an overall context. The suggested new fault tolerance strategies may change the availability and reliability of the BTSE significantly, but the benefits may be hidden by stronger impairments of other NEs in the path. Let us have a look at the traffic figures of the other NEs:

- a BSC is able to handle up to 1000-2000 Erlang
- a MSC is able to handle 4000 to 10000 Erlang
- the following public telephone switches may handle traffic equal to the MSC or even more.

The MSC is in fact a modified version of the SIEMENS EWSD switch for the public telephone system. Due to the high traffic values in the BSC and MSC the implementation of these NEs allowed the use of high sophisticated FT schemes. In all the core parts of the BSC and MSC hot redundancy schemes were used. This means that a failure of any single component could be tolerated without a loss of any call. Unfortunately, we cannot provide any figures about availability of the BSC and MSC or about the number of calls lost in spite of the good FT schemes.

However, due to the high traffic figures we can assume that it has been tried to incorporate every technically achievable and useful fault tolerance mechanism in these NEs. This isn't the case for the BTSE: here we encounter 40 to 50 times less traffic than on a BSC. The BER for hot redundancy schemes would be much better for NEs in the higher hierarchy (BSC, MSC) than for the BTSE to achieve similar premature release probabilities. In a Mobile Network the BTSEs are needed for every cell. So in comparison to the BSCs or MSCs the BTSEs are more a mass product, which is subject to different cost aspects. Some operators would not buy expensive redundancy schemes. Introducing those schemes would only make sense for special secure applications (like GSM-Railway) or for dedicated BTSEs, which need to handle a lot of traffic.

On the other hand the BTSE offers opportunities for improvement of FT, because up to now there isn't any hardware redundancy apart from the power supply unit or the receiver diversity. Basic software FT is already used there like restarts, reconfiguration and different forms of recoveries (see chapter 3.3.2.5), but a hot redundancy/reconfiguration scheme without a loss of calls is not used there.

7.2 Comparison with the Unreliability of the Air Interface

Another important reliability factor on the path from end user to end user is the air interface itself (see Figure 2.15: and Figure 2.16: for illustrations of this path). Its influence on reliability cannot be neglected, although there exist several possibilities for improvement of bad air interface parameters:
- improvement of the cell or network plans,
- usage of advanced features like frequency hopping, discontinuous transmission (Sending only when something is spoken)
- or other methods to reduce the overall interference level in the air.

Operators rarely publish measurements and statistics about the quality of the air interface, this would enable a quantitative comparison of mobile network competitors, which is not wanted. Two exceptions have been found [BF97] and [GSCL98]. [BF97] publishes several figures from the network of the German Telecom. Some of them are:

- The mean call holding time (or average call length) lies between 90 and 130 seconds. For mobile originated calls it is about 10-20 seconds shorter than for mobile terminated calls (calls from the fixed network to the mobile station) (See also Assumption 3.39)
- The number of handovers per call during the busy hour lies between 0.8 and 2.1 with an average of about 1.3.
- The successful call ratio of mobile originating traffic is given by 55% for mobile originating traffic and 37 % for mobile terminating traffic (this includes all possibilities that a call cannot be successful: premature release, network blocking, and the probability that nobody picks up the call on the other side: no answer ratio...)

[GSCL98] contains figures from Airtouch International about the growth of a network and estimated number of subscribers and penetrations, they also give several performance values of the handover procedure. A set of 7 cells was measured: a handover could be either successful with an average of 72.1 %, it could fail and return to old cell with an average of 23.7 % and in 4.2 % of the cases the call was dropped (prematurely released). With handover rate from [BF97] of 1.3 handovers per call we can calculate a mean premature release probability of calls due to handover of 5.5 %. This is about a factor of 60 higher than the premature release probability due to failures in the BTSE (8.93 E-04). Although the samples of both statistics about the air interface may be a bit too bad and the cells measured may be improved by certain measures, we can conclude that the PRP due to hard and software problems is completely hidden by the unreliability of the air interface.

7.3 Extrapolation to the Future

The growth of popularity of mobile communications is still immense. The penetration values of mobile services increases dramatically all over the world. As there are constantly more mobile subscribers, the operators need to enhance their network for more traffic. There are several possibilities to achieve this:

- Replanning the network and using smaller cell structures where necessary or splitting a cell into several cells. This may need installations of more sites, which increases the fixed costs for the operators and is expensive.
- Equipping the existing BTSEs with more TRXs: Each TRX requires the use of another frequency. So this implies either using free available frequencies or using a tighter reuse scheme in the network.

In general it could be observed, that the mean number of TRXs would increase as the penetration rate increases. At the start of the network operation small configurations of BTSEs are used (e.g. 1/1/1), when high subscriber figures are reached and also a high geographical density of subscribers is given much larger configurations are necessary (4/4/4 or even more). Extreme examples could be found in overcrowded cities of China.

In this chapter the following questions will now be discussed:

- What influence will the growth of the BTSE's size have on reliability and availability aspects?
- What can we derive for the PRP and PTT metrics?
- Which FT-Strategy might by useful in the future?

For an extrapolation of the future an assumption about the major configuration of the future is needed. For high traffic scenarios mostly sectorized BTSEs are used with 3 cells and many TRX. For the following calculation we assume a network in the future which will only consist out of 4/4/4 BTSEs. It is also assumed that this 4/4/4 BTSE will be realized with only one central controller. A 4/4/4 network would need about 3.368 times more TRX equipment than today. In addition, every BTSE will be expected to be sectorized in three sectors, i.e. 1.371 times more CELL equipment than today. Table 7.1: gives the Erlang values of such a configuration in comparison to the mean of today's configuration mix.

Permissible Traffic in Erlang at 2% Blocking	Normal Operation	TRX Failure	Cell Failure	BTSM Failure
Today's mean (world sample)	18.167	12.605	9.183	1.059
Future Extrapolation: 4/4/4	78.306	70.607	49.644	78.306

Table 7.1: Permissible Traffic Comparison Today/Future

7.3.1 Future Extrapolation for PTT

For the PTT two different effects will be observable in the future:
As Table 7.1: shows the traffic values of a future BTSE will be much higher, in case of the FMO failures significantly more Traffic channels will be lost, since the cells or the complete BTSEs are configured with more TRX. For the time needed for recovery or repair the PTT curve will drop down much more than today (see Table 7.2:).
It could be expected that the ΔA of a TRX-Failure would remain the same but it also increases by a factor of 1.384. This has two reasons:

- The number of signalling channels needed per TRX decreases a bit. More channels could share one common signalling channel.
- In general there is a “bundling gain” for the permissible traffic if more channels/servers were added. For example for 8 channels at blocking probability of 2% the permissible traffic is 3.63 Erlang. For 16 channels (B=2%) it is 9.83 and for 32 it is.23.7 Erlang.

Loss of permissible Traffic in Erlang	TRX Failure ΔA	Cell Failure ΔA	BTSM Failure ΔA
Future (4/4/4)	7.699	28.662	77.247
Today’s mean	5.562	8.984	17.108
Factor Future/Today	1.384	3.19	4.515

Table 7.2: Loss of Permissible Traffic Today/Future by FMO Failures

This first effect is illustrated in Figure 7.1. A (4/4/4) BTSE will serve 4.31 times more traffic. The additional loss for the different FMOs is marked in the figure.

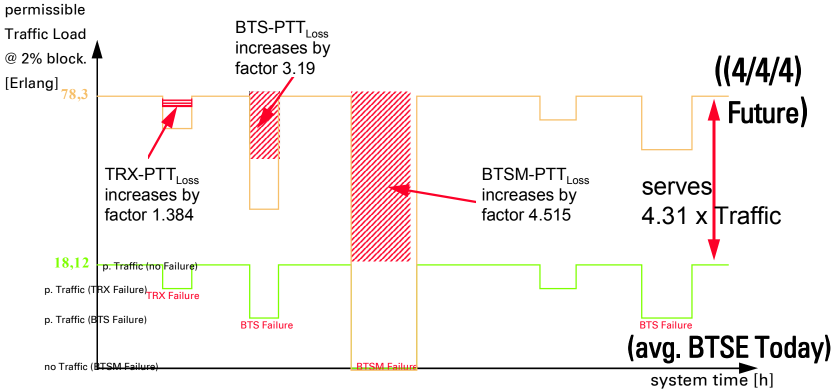


Figure 7.1: Effect 1 „More Loss by More Traffic“

The second effect that is observable stems from the fact that future BTSEs consist of more components. If we assume the same components with the same reliability and availability as today, the number of failures will increase according to the increase of components used. These increase factors were given above in the introduction of the chapter 7.3. The following Figure 7.2 illustrates the second effect.

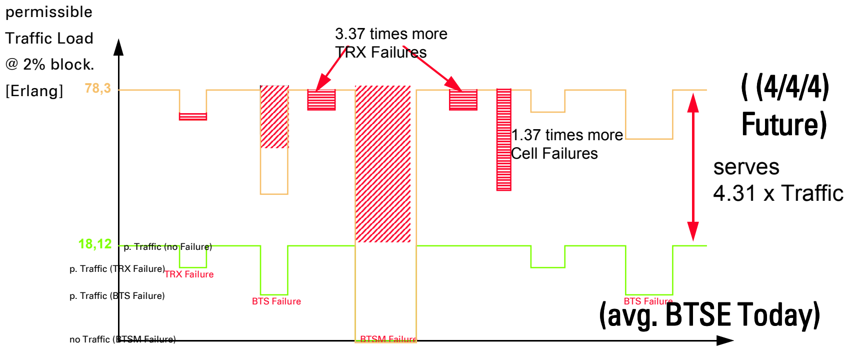


Figure 7.2: Effect 2: „More TRX/Cell Equipment leads to more Failures“

Looking at the contributions of either the FMOs or the HMOs it can be shown that both effects implicate that the contributions to PTT_{Loss} remain nearly the same. No component gains in importance in the future. Table 7.3 illustrates this.

PTT_{Loss} by FMO	Today's mean	relative contribution Today	Future extrapolation (4/4/4)	relative contribution Future
TRX	205.05	37.40%	955.82	38.57%
CELL	162.19	44.82%	709.62	43.39%
BTSM	65.35	18.06%	295.07	18.04%
SUM	432.59	100.00%	1960.50	100.00%

Table 7.3: Future Contribution of the FMOs to PTT_{Loss}

A slight increase in the contribution of the TRX FMO (or its HMO components) can be noticed in favor of a decrease of the CELL contribution (or its HMO components).

To summarize: Since for the calculation of the availability (see Definition 3.13) also the maximum permissible traffic (which is in case of the (4/4/4) is much higher) is taken into account, the availability for today does not significantly differ from the future availability (both values are about 0.998). For the evaluation of the fault tolerance schemes can be concluded that the components will have nearly the same influence on the PTT_{Loss} figures in the future, i.e. the ranking and the given benefits of the FT schemes will remain the same in the future.

7.3.2 Future Extrapolation for PRP

For the PRP also two effects can be observed. If we again examine the PRP formula (see Equation 3.43 or below) we notice that it depends on the factors k1, k2, k3, which are specific for the BTSEs used. For a (4/4/4) configuration, realized with one central controller, the factors are k1=1, k2=0.33333, and k3=0.0953 (assuming the same ratio of permanent TRX failures). This is different from today's values of k1=0.985 and k2=0.5815 and k3=0.4514.

$$PRP = \frac{T_{call} \cdot (\#BTSM_{fail} \cdot k_1 + \#CELL_{fail} \cdot k_2 + \#TRX_{fail} \cdot k_3(p_{permTRX}))}{Availability \cdot T}$$

The number of BTSM Failures will remain the same, since not any more components need to be used there. The Cell Failures will increase by factor 1.371 and the TRX failures will increase by a factor of 3.368 like it was assumed in the last chapter.
The PRP extrapolation calculates by:

$$PRP_{future} = \frac{T_{call} \cdot (\#BTSM_{failtoday} \cdot \frac{1}{k1_{today}} k1_{today} + \frac{1.371 \cdot 0.33333}{k2_{today}} \cdot k2_{today} \cdot \#CELL_{failtoday} + \frac{3.368 \cdot 0.0953}{k3_{today}} k3_{today} \#TRX_{failtoday})}{Availability \cdot T}$$

$$PRP_{future} = \frac{T_{call} \cdot (\#BTSM_{failtoday} \cdot 1.02 \cdot 0.985 + 0.7811 \cdot 0.5815 \cdot \#CELL_{failtoday} + 0.711 \cdot 0.4514 \#TRX_{failtoday})}{Availability \cdot T}$$

Equation 7.1: PRP Extrapolation

In Equation 7.1 it can be noticed that the BTSM failure term is modified by factor 1.02 (nearly remains the same), the Cell Failure term is modified by 0.7811, i.e. cell failures loose importance in the future, and the TRX failure term is modified by 0.711, which means that the TRX failures also will loose importance. The general extrapolation is that the PRP will be a bit smaller, the traffic will be distributed on more components and the BCCH-reconfiguration will not be so harmful, since there are more TRXs that could be taken as a reserve if a BCCH reconfiguration is necessary.
It could also be observed that the BTSM FMO and its components will gain in importance and the CELL and TRX FMOs and its components will loose importance.
This makes hot FT-schemes (schemes without dropping calls) for central HMOs, especially the Abis Link redundancy, more important in the future.

8 Conclusion

8.1 Presented Work

In this thesis we set up appropriate metrics to measure availability and reliability for the described class of systems. It was shown that the standard metrics are inappropriate, because they don't take the telephone service itself into account.
Then we explained how we can calculate those metrics out of real measured fault rates in form of logfiles about faults of long surveillance periods from many systems. Since the logfiles gave also the information about where the alarm originated we were able to calculate the contribution to the metrics for the components of the system. Standard statistic formulas can be used to estimate confidence intervals for these metrics. Based on the results for the metrics it is demonstrated how the weak spots can be identified.
Already used and new possible FT-schemes for the example BTSE have been categorized and described.
Finally we used the results of the metrics for the evaluation of new fault tolerance schemes. Benefits of several new candidate fault tolerance schemes have been calculated and predicted. This thesis clarified the performance of the schemes and gave the basis for selecting them according to the benefit-effort ratio.
Various other problems like the general unreliability of the air-interface, the possible behaviour in the future and the general importance of the BTSE in the network have been discussed.

8.2 Generalization

As already mentioned adapting the method to other telecommunication systems is possible. This generalization is a major conclusion of the thesis; the following steps have to be performed in a way similar to how we showed this for the BTSE:

1. Gather alarm data from many systems and for long observation periods. The gathered data needs to be detailed enough to identify a probable component that produced this alarm and probable cause or reason for the alarm.
2. Model the system behaviour in detail: i.e. models are needed on how the faults that produce the alarms have an effect on failures and finally on
   - the permissible traffic the system is able to handle,
   - the premature release,
   - or even the re-establishment in the network if a premature release happened.

   It is also necessary to measure and/or estimate recovery and repair times for all possible classes of failures.
3. Calculate the PTT, PRP and PURE metrics based on the system behaviour models and the gathered data.
4. Determine the component contributions to the metrics and detect the weak spots.
5. Eliminate weak spots directly, or develop new FT-strategies to cover the weak spots.

6. Estimate effort for the new FT-strategies
7. Rate the new FT-strategies using the benefit-effort ratio
8. Select the best FT-strategies on the rating

8.3 Future Work

8.3.1 Verification of the Presented Results

As already mentioned some results should be examined more in detail by other methods or investigations to gain more confidence about them. In the following it is briefly summarized which problems need to be researched in more detail to verify the results.

- More logfiles and also logfiles that observe longer time periods should be investigated. It is assumed that logfiles with a network size >300 BTSEs and a duration of >2-3 months would be needed for good results. Also different networks should be investigated

- In some of the logfiles the Abis-Link is the major cause of unreliability. It should be clarified if and how these problems are dependent on the type of link. If a cable connection is used it is supposed that the Abis link is more reliable than using the radio relay equipment version.

- Due to the large amount of Abis link alarms the modeling of this alarm is very important. The mean time that is needed for recovery of an Abis link failure was assumed to be 30 seconds this value was determined from some alarm logs with time stamps. A larger or smaller value would have a strong influence on the given results. Further studies have to be made to verify this value. Due the high number of alarms it also needs to be verified that one Abis link alarm equals one Abis link failure. It may either be possible that some failures haven't produce any alarm or the alarm has not been logged or while the link was down the alarm erroneously logged multiple times.

- Measurements from real networks about blocking, premature release or performance and availability measurements of base stations should be compared with the results from the presented method. This would give an estimation of how perfect the used models are and what parts of the models should be improved even more.

- The operators and SIEMENS service departments make statistics about Hardware MTBF values of components. A comparison with the presented long-term failure rates may also be useful.

8.3.2 Applying the Method to other Network Elements

The presented method should be applied to all network elements in the chain from end user to end user. This aspect was already discussed in chapter 7.1. Further studies have to be made for the other network elements (PSTN, MSC, TRAU, and BSC). Only with exact figures for all network elements the real weak spots could be detected. This dissertation covered only the BTSE and it is unknown if the unreliability and unavailability of the BTSE are or are not hidden by stronger impairments of other NEs.
Also different effort may be necessary to improve the fault tolerance behaviour in other NEs than in the BTSE, it may be easier or more difficult, this is still unknown and should be researched.

8.3.3 Comparing Different BTSE Versions

In the last years more and more different BTSE versions came up, they not only differ in the number of TRX used and aren't built from identical components any more. Especially several versions with different RF equipment exist now. Also special lightweight BTSEs for very small cells (hotspots) are built. Alarm log data should be separated by the kind of BTSE and the reliability and availability of the BTSEs should be examined separately. In future work the component-based differences and its effect on availability and reliability could be determined. The different variants may afterwards be compared to each other. In chapter 5.6 we already suspected that different RF equipment has a strong influence on the presented results, this should be clarified.

8.3.4 Validating new System Architectures and other FT-Techniques

Especially for third generation system a change in the system architecture may be necessary. Our method may be used to evaluate the use of such architectures. In this work not all possible FT-techniques where listed. New ones may be identified and could be evaluated in the same way. For the unreliable RF-equipment it may be worthwhile searching a redundancy solution.

8.4 New Areas for Research

Now we return to the central question of this thesis and specify what else should be examined in future research to solve the problem of fault tolerance improvement of mobile telephony in more detail. The following chapters summarize the possible topics for future research:

8.4.1 Taking Network Structures into Account

In chapter 3.6 we presented a model for mobile networks, with real cell structures and how to determine the re-establishing probability in a simple way. For network planning and also for air interface research and improvement several real network

simulators exist. These simulators allow a dynamical simulation of moving mobile users in several cells of an underlying real or homogenous cell structure.
The simulators give several results about the network quality:
- mean C/I (carrier to interferer),
- call drops (premature release),
- FER Frame Erasure Rate (dropped data packets).
Such simulators could be modified and additionally failures of base stations could be simulated, too.
The failure rates have already been determined in this dissertation. New network simulations would reveal results about the following:

- Using network simulations the PRP and PURE metrics could be determined more precisely, especially if the network plan of a given scenario is used for the simulation.

- Network simulations could be used to compare different network structures and scenarios:
  For different network structures and with different BTSE configurations a fault tolerance improvement may be more useful or not. Also different FT-techniques may be more useful or not. Simulations with different network structures might solve this problem. The potential for FT-improvement could be different for dense urban cell structures from that for rural areas. In dense urban scenarios more traffic is handled in rural areas the MTTR will be higher. Further studies on this topic are necessary.

8.4.2 Evaluating High-Availability Networks

Special network structures for high-availability networks (like GSM-Railway) use a sophisticated network redundancy pattern (overlapping cells and BSC areas). For such networks the research of fault-tolerance methods is highly needed. Network simulation could give also more results here.

8.4.3 Fault Tolerance of Third Generation Systems

Third generation CDMA systems use a technique called Soft Handover. This means a mobile user may be connected to more than one base station at a time and receive and send from/to two or more base stations. It is estimated that up to 50 % of the mobile users are in such a state while traveling through the cells. If the quality on one link drops below a certain barrier speech is not interrupted and no hand over is necessary, since another link should be active. This also represents a high sophisticated fault tolerance technique. Equipment failures of a single base station may also be tolerated by this technique. For the mobile station there will be no difference of loosing a link by equipment problems or by bad link quality. So especially for the third generation systems, but also for the current generation an effort comparison between multiple cell coverage as FT technique and equipment related FT techniques is highly needed and should be researched.

9 Terms and Abbreviations

Abis:	BTSE-BSC Interface
ACDC	HMO for the power supply units
ACOM	Antenna Combiner
AE	SW Subsystem Alarm- and Errorhandling
ALCO	Alarm Controller Board
ARPA	Advanced Research Projects Agency
ASCII	American Standard Code for Information Interchange
BBSIG	Base Band and Signalling
BCCH	Broadcast Channel
BER	Benefit-Effort Ratio
BSC	Base Station Controller
BTS	Base Transceiver Station, can be used also for the FMO CELL Note: should not be confused with BTSE
BTSE	Base Transceiver Station Equipment, complete equipment installed at one site
BTSM	FMO Base Transceiver Station Manager, contains all central HW objects of a site
CCITT	Comite Consultatif International de Telegraphie et de Telephonie
CCLK	Master Clock
CCTRL	Core Controller
CDMA	Code Division Multiple Access
CELL	FMO Cell, contains all cell related hardware
CS	SW Subsystem Configuration and State Management
CU	Carrier Unit
DARPA	Defense Advanced Research Projects Agency
DCS1800	Digital Cellular (Telecommunication) System, 1800 Mhz
DG	SW Subsystem Diagnosis
ESPRIT	European Strategic Programme for Research in Information Technology
ETSI	European Telecommunications Standards Institute
EWSD	Elektronisches Waehlsystem, digital
FER	Frame Erasure Rate
FM	Failure Model
FMO	Functional related Managed Object
FP	SW Subsystem Fault Protection
FT	Fault Tolerance
FU	Functional Unit
GHz	Giga Hertz
GMSK	Gaussian Minimum Shift Keying
GOS	Grade of Service
GPSU	Generic Power Supply Unit
GSM	Groupe Special Mobile, Global System for Mobile Communications digital mobile standard

9 Terms and Abbreviations

GSM900	GSM 900MHz
GUI	Graphical User Interface
HEX	Hexadecimal Notation-Base 16
HMO	Hardware related Managed Object
HW	Hardware
IEEE	Institute of Electrical and Electronics Engineers
IMT-2000	International Mobile Telecommunications 2000
ITU	The International Telecommunications Union (former CCITT)
Standards	
	Organization
ISDN	Integrated Services Digital Network (ISDN)
LI	Link Interface
LMT	Local Maintenance Terminal
PCM	Pulse Coded Modulation
PTT	Permissible Traffic Time
PTT$_{\textbf{down}}$	Loss of Permissible Traffic Time
PRP	Premature Release Probability
MCLK	Master Clock
MoU	Memorandum of Understanding
MoU SERG	MoU-Service Group
MSC	Mobile services Switching Centre
MS	Mobile Station
MSE	Error Mean Square
MTBF	Mean Time Between Failures
MT	Mission Time
MTTF	Mean Time To Failures
MTTR	Mean Time To Repair
NE	Network Element
NVP	N-Version Programming
O&M	Operation and Maintenance
OMC	Operation and Maintenance Center
PA	Power Amplifier
PCM30	Pulse Code Modulation (with 30 data channels/timeslots)
PCS	Personal Communications System
PRP	Premature Release Probability
PSTN	Public Switched Telephone Network (PSTN)
PTT	Permissible Traffic Time
PURE	Probability of Unsuccessful Re-establishment
RC	SW Subsystem Recovery
RF	Radio Frequency
RT	Recovery/Repair Time
RXAMOD	Receiver Antenna Module
RXMUCO	Receiver Multicoupler
SDCCH	Standalone Dedicated Control Channel
SMS	Short Message Service
SW	Software

TCH	Traffic Channel
TDMA	Time Division Multiple Access
TMR	Triple Modular Redundancy
TPU	Transceiver and Processor Unit
TRAU	Transcoding and Rate Adaption Unit
TRX	Transceiver (FMO) contains all hardware related for sending and receiving on eight full rate channels
TRXA	Transceiver Analogue
TRXD	Transceiver Digital
TTR	Time To Repair
TXAMOD	Transmit Antenna Module
UMTS	Universal Mobile Telecommunication System
y	Average Served Traffic Intensity

10 Literature

[ABBG96] A. Alwan, R. Bagrodia, N. Bambos, M. Gerla, and others: „Adaptive Mobile Multimedia Networks“
1996 IEEE Personal Communications, Apr. 1996, pages 34-51
ISSN 1070-9916
A fault tolerant technique for rapidly deployable military wireless networks is described.

[All90] Arnold O. Allen: „Probability, statistics, and queueing theory: with computer science applications“
1990 Academic Press, Inc,
ISBN 0-12-051051-0
Good practical book about queueing theory. Contains derivation of the Erlang formula from the M/M/c/c queueing system.

[AL81] T. Anderson, P.A. Lee: „Fault Tolerance, principles and practice“
1981 Prentice-Hall International,
ISBN 0-13-308254-7
Standard literature about FT

[BKCG98] E. A. Brewer, R. H. Katz, Y. Chawathe, S. D. Gribble, and others: „A Network Architecture for Heterogeneous Mobile Computing“
1998 IEEE Personal Communications, Oct. 1998, pages 8-24
ISSN 1070-9916
Describes a fault tolerant wireless data communication by using multiple overlaying networks.

[Bre69] L. Breiman: „Probability and stochastic processes. With a view toward applications“
1969 Mifflin
Standard lit., contains derivations of formulas for Markov processes

[BF97] V. Brass, W. F. Fuhrmann: „Traffic Engineering Experience from Operating Cellular Networks“
1997 IEEE Communications Magazine, August 1997, pages 66-71,
ISSN 0163-6804
Rare publication of measured traffic and performance data.

[BFS83] Paul Bratley, Bennett L. Fox, Linus E. Schrage, „A guide to simulation“
1983 Springer-Verlag New York, Berlin, Heidelberg
ISBN 0-387-90820-X
Good practical book about modeling and simulation

[CKLS97]: Tsu Wei Chen, P. Krzyzanowski, M. R. Lyu, C. Sreenan: „Renegotiable quality of service- a new scheme for fault tolerance in wireless networks“
1997 IEEE Comput. Soc., 27. Symposium on Fault-Tolerant Computing
ISBN 0-8186-7831-3
Conference paper

[CMAO97] R. D. Carsello, R. Meidan, S. Allpress, F. O´Brien, and others: „IMT-2000 Standards: Radio Aspects“
1997 IEEE Personal Communications, Aug.1997, pages 30-40
ISSN 1070-9916
Discusses the objectives of IMT-2000 architectures and systems.

[Dal79]: M. Dal Cin: "Fehlertolerante Systeme"
1979 Teubner, Stuttgart
ISBN 3-519-02352-0
Mathematical Models for reliability, availability, diagnosis and repair

[DCS97] Siemens: „D-Spec for Configuration and State Management“
1997 Siemens internal design specification of the CS Subsystem (Software) 1
Siemens DocNr. P30862-A0001-D119-01-76D8
Detailed description of the CS Subsystem.

[Ech90] Klaus Echtle: „Fehlertoleranzverfahren“
1990 Springer-Verlag, Berlin Heidelberg
ISBN 3-540-52680-3
Standard literature about FT, gives a good overview and describes techniques for all the different phases.

[F1S94] Siemens: „F1-Specification, BTSE for GSM900 and DCS1800 BR.2“
1994 Siemens internal functional specification level 1
Siemens DocNr. P30862-A1-A11--02-7659 Rel. 02/02
This specification gives the most abstract view of this system for hardware and software.

[FT97] Karl Fuchs, Jean-Michel Traynard: "Telecommunication network and state propagation method"
1997 Patent of Siemens AG
European Patent Office, Patent No. 97114935.6

[Gar97] B. Garner: „PCS base station maintenance calls for a fresh approach“
1997 Mobile Radio Technology, volume 15, pages 11-14
ISSN 0745-7626
Journal paper

[Gib76] T. Gibbons: „Integrity and recovery in computer systems“
1976 NCC Publications, Manchester
ISBN 0-85012-130-2
Chapter 3 contains appropriate fault/error/failure definitions

[Goe92]: I. Goetz: „Ensuring mobile communications transmission quality in the year 2000“
London 1992, Digest No. 139 of IEE Colloquium on „Mobile Communications in the year 2000“
Conference paper

[Gör89]: W. Görke: „Fehlertolerante Rechensysteme“
1989 Oldenbourg Verlag, München, Wien
standard lit., with commercial and research examples
ISBN 3-486-20776-8

[Gri96] D. Grillo: „Personal Communications and Traffic Engineering in ITU-T: The Developing E.750 Series of Recommendations“
1996 IEEE Personal Communications, Dec.1996, pages 16-28
ISSN 1070-9916
Describes the named ITU rec. for traffic engineering and performance measurement.

[GSCL98] D. Grillo, R. A. Skoog, S. Chia, K. K. Leung „Teletraffic Engineering for Mobile Personal Communications in ITU-T Work: The Need to Match Practice and Theory“
1998 IEEE Personal Communications, Dec.1997, pages 38-58
ISSN 1070-9916
Contains measured traffic model parameters, especially handover failure probabilities.

[GT96] M. Greiner, G. Tinhofer: „Stochastik für Studienanfänger der Informatik“
1996 Hanser Verlag, München, Wien
Introduction to statistics and probability theory for computer science
ISBN 3-446-18636-0

[HE95] W. Hufnagel, M. Ermer: „Functionality and architecture of the call handling software in the DCMS base station controller for the GSM system“
1995 Philips Telecommunication Review, vol. 53, pages 46-53
ISSN 0926-129X
Journal paper

[ITU92] The International Telegraph and Telephone Consultative Commitee: "Connection Retainability Objective for the International Telephone Service"
10/1992 Published by the ITU
ITU/CCITT Recommendation E.850
Standard recommendation of the ITU about premature release and related measures

[Jab96] B. Jabbari: „Teletraffic Aspects of Evolving and Next-Generation Wireless Communication Networks“
1996 IEEE Personal Communications, Dec.1996, pages 4-9
ISSN 1070-9916
Gives an overview to performance modeling of wireless networks.

[Jal94]: Pankaj Jalote „Fault Tolerance in Distributed Systems"
1994 Prentice Hall, Englewood Cliffs
ISBN 0-13-301367-7
Describes problems and solutions for implementations of software fault tolerance in distributed systems.

[Joh89]: Barry W. Johnson: „Design and Analysis of Fault-Tolerant Digital Systems“
1989 Addison-Wesley
standard lit.
ISBN 0-201-07570-9

[JV86]: E. Jessen, R. Valk: „Rechensysteme“
1987 Springer, Berlin, Heidelberg
ISBN 3-540-16383-2
Describes methods for modeling and analysis of computer systems.

[Keh98]: Hubertus Kehl: „Determining Multiple Cell Coverage by use of simple image processing on network planning predictions“
1998 Internal Siemens Paper

[Lan86]: Graham Langley: „Telephony’s Dictionary, Second edition“
1986 Telephony Publishing Co.
ISBN 0-917845-04-8
Definitions of telecommunication terms and abbreviations.

[LB97] Anne Y. Lee, Bohdan L. Bodnar: „Architecture and performance analysis of packet-based mobile switching center-to-base station traffic communications for TDMA“
1997 Bell Labs Technical Journal, volume 2, number 3, page 46-56
ISSN 1089-7089
Journal paper

[LHKC97] M.-S. Lee, W.-P. Hong, T.-G. Kim, Y.-H. Choi, and others: „Current Status and Development Strategy of Personal Communication Services of Korea Telecom“
1997 IEEE Personal Communications, Apr.1997, pages 44-50
ISSN 1070-9916
Description of the architecture and design of Korean mobile radio network elements.

[LRS96] B. M. Leiner, R. J. Ruth, A. R. Sastry: „Goals and Challenges of the DARPA GloMo Program“
1996 IEEE Personal Communications, Dec.1996, pages 34-43
ISSN 1070-9916
States the requirements and goals of a military wireless information system.

[MOU94] MOU SERG: „GSM handbook version 3.00.01 Phase I MOU SE1“
1994 Siemens internal electronic document
Good short introduction to GSM

[MoPa92] Michel Mouly, Marie Bernadette Pautet: „The GSM System for Mobile Communications“
1992 Published by Authors
ISBN 2-9507190-0-7
Good GSM standard lit. gives an overview and introduction to GSM

[MoPe92] Douglas C. Montgomery, Elizabeth A. Peck: „Introduction to Linear Regression Analysis“
1992 John Wiley & Sons Inc.
ISBN 0-471-53387-4
Detailed book about linear regression, covers also the derivations of the special case „regression through the origin“.

[PKV96]: Dhraj K. Pradhan, P. Krishna, Nitin H. Vaidya: „Recoverable Mobile Environment: Design and Trade-off Analysis“
1996 IEEE Comput. Soc. Press, 26. Symposium on Fault-Tolerant Computing
ISBN 0-8186-7261-7
Conference paper

[Rao65] C. Radhakrishna Rao: „Linear Statistical Inference and Its Applications“ and
„Lineare statistische Methoden und ihre Anwendungen“
1965 John Wiley & Sons, Inc. New York / 1973 Akademie Verlag Berlin
Standard Literature about Statistics

[RLKL95] B. Randell, J.-C. Laprie, H. Kopetz, B. Littlewood (Editors) „Predictably Dependable Computer Systems“
1995 Springer, Berlin Heidelberg, ESPRIT Basic Research Series
ISBN 3-540-59334-9
Various fault tolerance articles from the ESPRIT research project.

[RRD95]: S. Rangarajan, K. Ratnam, A.T. Dahbura: „A fault-tolerant protocol for location directory maintenance in mobile networks“
1995 IEEE Comput. Soc. Press, 25. Symposium on Fault-Tolerant Computing
ISBN 0-8186-7079-7
Conference paper

[RX95]: Brian Randell, Jie Xu: „The Evolution of the Recovery Block Concept“ in Michael R. Lyu (Editor): „Software Fault Tolerance“
1995 John Wiley & Sons, England
ISBN 0-471-95068-8
Various articles about software fault tolerance, like N-version programming and evaluation of those.

[SS82]: Daniel P. Siewiorek, Robert S. Swarz: „The Theory and Practice of Reliable System Design“
1982 Digital Press, Bedford
ISBN 0-932376-13-4
Standard lit., many examples from praxis, models for evaluation.

[TC96]: Patrick Trane, David Carlier: „Diagnosis Algorithm for Mobility-Oriented System“
1996 IEEE Comput. Soc. Press, International Conference on Application-Specific Systems, Architectures and Processors 1996
ISBN 0-8186-7542-X
Conference paper

[TF81] Siemens: „Tabellenbuch Fernsprechverkehrstheorie“
1981 Siemens AG, Berlin, München
ISBN 3-8009-1343-7
Explanation and tables of the Erlang formulas. Also includes good formulas for numerical evaluation.

[TMWH92] K.S. Trivedi, J.K. Muppala, S.P. Woolet, B.R. Haverkort: „Composite performance and dependability analysis“
1992 Performance Evaluation, vol. 14
ISSN 0166-5316
Conference paper that describes Markov reward models for evaluation of performance and dependability of fault tolerant systems.

[Tri82] K.S. Trivedi: „Probability and Statistics with Reliability, Queuing and Computer Science Applications“
1982 Prentice Hall, Inc., Englewood Cliffs
ISBN 0-13-711564-4
Standard literature about probability, stochastic processes and statistics with application to reliability and performance analysis

[WSH90] K. Watanabe, H. Shimizu, T. Hori: „Development of GSM base station“
1990 Proceedings of Singapore ICCS ‘90: „Communications Systems: Towards Global Integration“
ISBN 0-444-88953-1
Conference paper