

VERIFICATION OF THE TRANSMISSION SYSTEM
MODEL RBTS USING MONTE CARLO
SIMULATION METHODS

Project report
by Johan Setréus

**Project report in EG2080 written at KTH, the Royal Institute of Technology,
School of Electrical Engineering, 2008**

Supervisor: Mikael Amelin, KTH School of Electrical Engineering

TRITA-EE 2009:019

Abstract

This project report describes the work of simulating the transmission system reliability model RBTS. Given the system data, including the component reliability, the system reliability index EENS (MWh/yr) is estimated with different Monte Carlo simulation methods. The main objective is to validate earlier results of the system. One other objective is to study how much the simulation efficiency can be improved by introducing variance reduction techniques.

The results shows that, (i) the results is in accordance with earlier published results, and (ii), with even a quite straightforward and simple model for stratified sampling, the efficiency can be improved with more than a factor five compared to simple sampling in this model of RBTS.

Contents

Abstract	iii
1 Introduction	1
1.1 Background	1
1.2 Problem Definition	1
1.3 Related Research	2
1.4 Overview of the Report	2
2 Modelling	3
2.1 Assumptions and Limitations	3
2.2 Symbols	4
2.3 Mathematical Model	5
3 Variance Reduction Techniques	7
3.1 Introduction to the problem	7
3.2 Complementary Random Numbers	7
3.3 Dagger Sampling	8
3.4 Control Variates	8
3.5 Correlated Sampling	9
3.6 Stratified Sampling	9
3.7 Importance Sampling	9
3.8 Combination of Variance Reduction Techniques	10
4 Case Study	11
4.1 System Description	11
4.2 Simulation Method	16
5 Conclusions	25
5.1 General Conclusions	25
5.2 Future Work	25
A Implemented code in Matlab	27
A.1 Simple sampling code	27
A.2 Stratified sampling code	33

A.3	Importance sampling code	39
A.4	The DC load flow method code	44

Chapter 1

Introduction

1.1 Background

An electrical power transmission system is a complex technical structure. It consists of a large number of interconnected subsystems and components which interact with and influence the overall system reliability. One definition of reliability is the ability of a component or system to perform required functions under stated conditions for a stated period of time [1]. The required function is here to supply the aggregate electric power and energy requirements of the customer, both generation and end users (load points), at all times.

Reliability assessments of the transmission system are often included in the decision making of e.g. new investments, maintenance and operation of the system. Power system reliability is often divided by the two functional aspects of system *adequacy* and *security*. Adequacy, which is considered in this report, is the ability of the power system to perform its required function. System security is the ability of the power system to withstand sudden disturbances, such as non-anticipated loss of system components, without violating any system constraints [2].

Various reliability test systems have been developed in order to evaluate and compare the results from developed reliability methods. Two advantages with test systems are that published results can be reproduced more easily and that the researcher also save some time. One such system is the Roy Billinton Test System (RBTS)[3] which is analyzed in this report.

1.2 Problem Definition

RBTS is a relatively small reliability test system consisting of six buses connected with nine overhead lines. Eleven generators produce power to the five load points where the customers are located. Each component (e.g. line and generator) has an associated reliability which influence the overall system reliability. Each line and generator has maximum power limits which are the main constraints of the system. The power demand of the load points has in this report been set to the system peak

load.

The objective of this report is to evaluate RBTS overall system reliability with a number of different Monte Carlo simulation (MCS) techniques. The inputs are the load point power demand and the components availability. The output of the model is the *expected energy not served* (EENS) in MWh per year (MWh/yr) for the total system. This is a commonly used reliability index for power systems.

The transmission system is generally operated and designed with a redundant criterion. Single component failures do not normally result in disconnected load for the load points. Furthermore, the components are normally very reliable and this makes reliability analysis a quite difficult task which often requires large computational efforts.

1.3 Related Research

In my own Ph.D. project the RBTS is used for demonstrating models and methods for quantifying the transmission system security level and the major risks to the same. The security of the system is quantified indirectly by determine each components impact on the systems capability to transfer active power together with the probability of the event to occur. Within the research the commercial power system program Neplan [4] has been used together with Matlab. Neplan uses an enumeration technique which evaluates up to second order component outages (two overlapping component failures or maintenance actions). RBTS has been modelled in Neplan and one benefit of this report is to be able to compare the result from Neplan with the MCS methods in this report. This comparison is not the aim of this report and not presented here.

1.4 Overview of the Report

Chapter 2 describes the mathematical modelling for a general power system model, with the limitations and assumptions made.

Chapter 3 evaluates the possible use of six different variance reduction techniques that might be used for simulation of the model.

Chapter 4 presents the case study system RBTS, which simulation methods that have been used and the result for these.

Chapter 5 summarizes the results with general conclusions and suggest possible future work with the model.

Chapter 2

Modelling

This chapter describes the mathematical model of the implemented load flow method used for the power system analysis.

2.1 Assumptions and Limitations

2.1.1 Power system

The model of the power transmission system in this report only considers and evaluates the steady states of the system and not the transitions between different states. Dynamical phenomenon that may result an instability of the system are neglected.

The transmission system in this report consists of three main *components*: generators, load points and lines. Other components exists in a real transmission system and this is one implantation. RBTS also has several additional components defined with e.g. the substation configuration, but this report is limited to a smaller version of the system that is commonly used.

The DC load flow method has in this report been adapted to evaluate the power flows in different steady states of the system. The following properties are some of the limitations of the DC load flow method:

- voltage constraints cannot be evaluated,
- reactive power constraints and power flow cannot be evaluated,
- transmission line losses are not included in the model.

The accuracy of this linear model approximation is not as good as compared to the AC load flow method. The advantages are however good convergency properties, easy implementation and relatively small computational requirements.

2.1.2 Component reliability model

The failure rates for the components are in this report assumed to be constant. This also applies to the restoration (repair) times for the components. These assump-

tions is in accordance with the RBTS data and are generally used within the area of research.

2.2 Symbols

2.2.1 General

$$\mathcal{B} = \text{set of buses} = 1, \dots, \text{nr_B} \quad (2.1)$$

$$\mathcal{L} = \text{set of lines} = 1, \dots, \text{nr_L} \quad (2.2)$$

$$\mathcal{L}_b = \text{set of lines connected to bus } b \quad (2.3)$$

$$\mathcal{G} = \text{set of generators} = 1, \dots, \text{nr_G} \quad (2.4)$$

$$\mathcal{G}_b = \text{set of generators connected to bus } b \quad (2.5)$$

$$\mathcal{LP} = \text{set of load points} = 1, \dots, \text{nr_LP} \quad (2.6)$$

$$\mathcal{LP}_b = \text{set of load points connected to bus } b \quad (2.7)$$

$$\mathcal{K} = \text{set of components } (\mathcal{L}, \mathcal{G}, \mathcal{LP}) = 1, \dots, \text{nr_K} \quad (2.8)$$

$$\theta = \text{set of bus voltage angles} = 1, \dots, \text{nr_B} \quad (2.9)$$

2.2.2 Inputs

$$A_k = \text{component } k \text{ available} \quad (2.10)$$

$$D_{lp} = \text{power demand in load point } lp \text{ [MW]} \quad (2.11)$$

2.2.3 Outputs

$$\theta_b = \text{voltage angle in bus } b \text{ [}^\circ\text{]} \quad (2.12)$$

$$PG_g = \text{power production in generator } g \text{ [MW]} \quad (2.13)$$

$$PLP_{lp} = \text{power production in load point } lp \text{ [MW]} \quad (2.14)$$

$$PL_l = \text{power transfer in line } l \text{ [MW]} \quad (2.15)$$

$$PU_{lp} = \text{disconnected power in load point } lp \text{ [MW]} \quad (2.16)$$

2.2.4 Parameters

$$c_g = \text{operation cost for generator } g \text{ [♣/MW]} \quad (2.17)$$

$$c_{lp} = \text{penalty for disconnected power in load point } lp \text{ [♣/MW]} \quad (2.18)$$

$$\overline{PG}_g = \text{maximum production in generator } g \text{ [MW]} \quad (2.19)$$

$$\underline{PG}_g = \text{minimum production in generator } g \text{ [MW]} \quad (2.20)$$

$$\overline{PL}_l = \text{transfer limit in transmission line } l \text{ [MW]} \quad (2.21)$$

$$T_l = \text{bus connections for line } l \text{ [from bus } b_i, \text{ to bus } b_j] \quad (2.22)$$

$$x_l = \text{reactance for transmission line } l \text{ [}\Omega] \quad (2.23)$$

$$y_l = \text{admittance for transmission line } l \text{ [}1/\Omega] \quad (2.24)$$

2.3 Mathematical Model

The mathematical model is an minimization problem.

2.3.1 Objective function

The objective is to minimize the total cost of supplying power to the system customers. This mi

$$\text{minimize } \sum_{g \in \mathcal{G}} c_g PG_g + \sum_{lp \in \mathcal{LP}} c_{lp} PLP_{lp} \quad (2.25)$$

2.3.2 Constraints

Subject to:

$$\sum_{g \in \mathcal{G}_b} PG_g + \sum_{lp \in \mathcal{LP}_b} PLP_{lp} + \sum_{l \in \mathcal{L}_b} PL_l = 0 \quad (2.26)$$

The interpretation of (2.26) is that the sum of generation, load demand and line flows are zero in each bus b of the system.

$$| PL_l | \leq \overline{PL}_l \quad (2.27)$$

Interpretation of (2.27): the active power flow in line l has to be below the maximum allowed transfer limit for that line.

$$\underline{PG}_g \leq PG_g \leq \overline{PG}_g \quad (2.28)$$

Interpretation of (2.28): each generator g can only produce active power within its minimum and maximum limits.

$$-D_{lp} \leq PLP_{lp} \leq 0 \quad (2.29)$$

Interpretation of (2.29): the production in each load point is between the demand production and zero production. The load point is here formulated as a generator with a negative production.

Chapter 3

Variance Reduction Techniques

This chapter describes six different variance reduction techniques for Monte Carlo simulation. Each technique is described if it is suitable or not to be used for the analyses of the power system model.

3.1 Introduction to the problem

The transmission system normally is designed to manage single component outages, without any influence on the load points. Overlapping component outages (two component outages or more) is normally the case when there is disconnected load in the system. The output result (disconnected load) population is highly duogeneous, e.g it consists of a majority of homogenous units and a minority of diverging units [5]. This property makes these types of system difficult to evaluate and to determine an estimate of the average disconnected load over a longer time period.

3.2 Complementary Random Numbers

The complementary random numbers method is a relatively straightforward method to implement for a power system model. In simple sampling each sample (observation) input to the model are independent of each other. However, in this method the target is to find the inputs, A_i (component availability) and the load demand, where a negative correlation between the samples, results in a negative correlation of the output which here is the disconnected load in the system. A negative correlation between the outputs have an variance reducing effect.

For simulations of the power systems complementary random numbers can be applied for the load demand input variables D_{lp} . But normally there is a much stronger correlation between the system reliability and the total demand than for each individual load point. For this reason the complementary random numbers instead are applied on the total load, and each load points demand are then scaled with new random numbers to match this total demand.

The same procedure can also be applied for the generation in the system. The different states for the generators in each node can be enumerated and the associated probability for each state determined. If there are too many states and generators, one alternative is to only apply the method to only the largest generation units.

For the lines this method can be applied for critical transfer sections which consist of several lines that transfer power in the same direction. Between these there might be a strong correlation. One problem is however that the outage rate for the lines normally are very small, and this is not suitable

3.3 Dagger Sampling

Dagger sampling is similar to the complementary random numbers method, but here the input is a two-state probability distribution and with one of the outcomes having a low probability. The aim is as before to identify the inputs where a negative correlation between the scenarios will result in negative correlations for one or more outputs.

For power system reliability analysis dagger sampling at a glance seems well suited. Each component has two states (function or not) with a specific outage rate and this is normally very low. The method improves the spread of the samples and the probability of only generating states where all components are functioning is small [5]. But, normally the system generation capacity benefits from being randomized as a total and not for each generation unit separately (as in dagger sampling).

The lines might however gain to be sampled with dagger sampling. These are more difficult to simulate with complementary random numbers, since it normally is harder to find lines with a strong correlation.

3.4 Control Variates

By introducing a new simplified model that is used together with the existing more complex model, the difference between them can be sampled. The expectation value from the simplified model needs to be analytically calculated and it needs to behave in approximately the same way as the complex model.

For the studied power transmission system model, it is difficult to make a simplified analytical model that still behaves as the complex. One possible way could be to model the output (disconnected load) only as a function of the available generation. The output (disconnected load) should be quite strongly correlated to the available generation, especially if the lines are very reliable and redundant.

3.5 Correlated Sampling

This method is similar to the control variates, but this instead study the difference between two different system which have almost the same properties.

In the studied power system model this could be useful if the expected reliability improvement is to be determined when e.g. a new line or generator is to be invested in.

3.6 Stratified Sampling

This method is useful if there is a knowledge of how the system behaves in certain parts of the total population. By dividing the population into parts which react more or less the same, the samples can be concentrated in those stratum (parts) which are most important to be studied. The results from each stratum are then weighted together to form the total system output.

The disadvantage of this method is the work it takes to define the population into different stratum. The time it takes to get accurate estimates of each stratum has to be less than the time it take to obtain the same estimate from the entire population.

For the studied power system model this a well suited method. With simple sampling the majority of the samples are "wasted" on system states were (i) all components is in function and (ii) only one component has failed. As mentioned earlier, the normal design of the transmission system prevent any of these event (samples) to influence the output result.

The system load or generation can normally be divided into different distinct stratum. The topology of the system may also give useful information how the stratum can be defined. Single lines or critical line sections that transfer large amounts of energy can be used as one strata.

3.7 Importance Sampling

This method is similar to stratified sampling, in the way that it aims to concentrate the samples were they are most important for estimating the output. This is achieved by modifying random number generation to a new probability distribution. The results for parts that are over and under represented in the population are the weighted to get the correct result.

This method is suitable for the studied power system model. A standard component may have an expected unavailability on e.g 0.1%. With a system with 20 of these components, the probability of sampling an intact system state is around 98%. By increasing the component unavailability to 10%, this probability is instead approximately 12%. The samples can be used more efficient

3.8 Combination of Variance Reduction Techniques

There are many alternatives to combine the different variance reduction technique to further improve the efficiency of the simulation. For the studied system model two alternatives are suggested.

The first combination is to use complementary random numbers together with a stratified sampling. The largest generation units and the most important lines can form the strata tree with the stratum. The load demand can then be generated in complementary scenarios with complementary random numbers.

The second alternative is to combine importance sampling with complementary random numbers. The lines in the power system has relatively low outage rates, and these are therefor sampled with the importance sampling method. The combined generator capacity in each node (e.g. two nodes with five generators in each) are then sampled with complementary random numbers.

Chapter 4

Case Study

This chapter presents the case system RBTS that has been modelled and simulated in MATLAB. The results from the applied variance reduction techniques and simple sampling are compared.

4.1 System Description

The following description of RBTS is aimed to be as general as possible. This reports implemented version of RBTS is here referred to as RBTS(A), and this is one commonly used configuration of RBTS used in literature.

4.1.1 General overview RBTS

The Roy Billinton test system (RBTS) is a model for reliability studies at the transmission level. Table 4.1 gives a brief summary of the system properties. The system data is defined in [3] and results with system and load point reliability indices are presented in [6]. In five of RBTS six buses (Bus 2-Bus 6) load points are present and for two of these (Bus 2 and Bus 4) the underlying distribution system is defined in detail in [7]. In this paper only the transmission system of RBTS is studied, with the distribution systems represented as single load points (LP).

The level of detail and complexity, as well as the different assumptions and approximations when RBTS is modelled depends on the purpose of the reliability study. Various implementations of RBTS in published material shows that it does not exist one version of the system, but many. One reason is that the specification of RBTS in [3] consists of a large amount of data in order to perform different types of reliability analysis for electrical power systems. The version of RBTS that has been modelled in this report is referred to as RBTS(A). If nothing else is mentioned, the term RBTS refers to the general system configuration presented in [3] and with RBTS(A) as one special case.

Figure 4.1 shows the single line diagram with the included components of RBTS.

Table 4.1: RBTS summary of system data [3]

Number of buses	6
Number of generators	11
Number of load points	5
Number of transmission lines	9
Number of generation buses	2
Installed generation [MW]	240
System peak load [MW]	185
AC nominal voltage [kV]	230

Table 4.2: RBTS transmission line data [3]

Line	From Bus	To Bus	Length (km)	Impedance (p.u.)		Susceptance B/2 (p.u.)
				R	X	
L1	1	3	75	0.0342	0.18	0.0106
L2	2	4	250	0.114	0.6	0.0352
L3	2	1	200	0.0912	0.48	0.0282
L4	4	3	50	0.0228	0.12	0.0071
L5	3	5	50	0.0228	0.12	0.0071
L6	1	3	75	0.0342	0.18	0.0106
L7	2	4	250	0.114	0.6	0.0352
L8	4	5	50	0.0228	0.12	0.0071
L9	5	6	50	0.0228	0.12	0.0071

$$S_{base} = 100 \text{ MVA}, U_{base} = 230 \text{ kV}$$

Max current capacity for lines:

$$\text{L1 and L6} \rightarrow I_{max} = 0.85 \text{ p.u.}, \text{ other line} \rightarrow I_{max} = 0.71 \text{ p.u.}$$

L1, L6 and L2, L7 has common towers for their entire length

The system frequency has not been specified in [3], but in this report it is assumed to be 60 Hz. The π -model is assumed for the transmission lines. The electrical and reliability data for the components is assumed to be ideal if noting else is specified.

4.1.2 Electrical data

The system has two generator (PV) buses and four load (PQ) buses. The voltages at the PV buses (Bus 1 and Bus 2) are controlled to 1.05 p.u. The voltage limits in the system are between 0.97 p.u. and 1.05 p.u.[3]. Table 4.2 shows the electrical data and lengths of the nine lines in the system.

Table 4.3 shows the load data for the LP:s at a system peak load. The load variation in RBTS (i.e. in p.u. of the peak load) is specified as the same as in

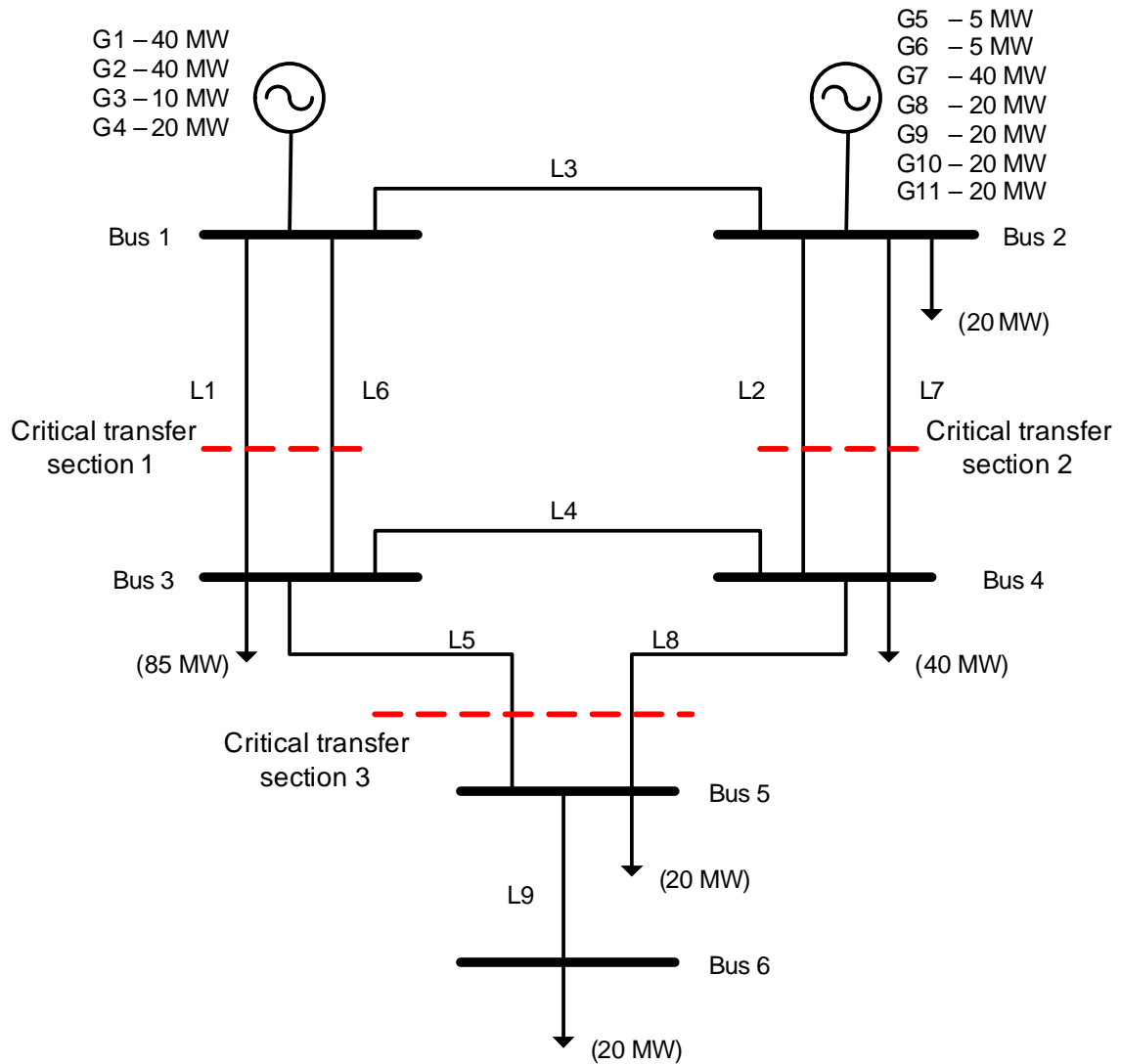


Figure 4.1: Single line diagram of RBTS. The load points LP2-LP6 refers to the attached load points in each bus2-bus6.

IEEE-RTS [8]. This report has limited the analysis to a peak load level. No load shedding policy, in case of severe outage events, has been specified in [3]. In this paper the priority order policy, defined for RBTS in [9], is implemented.

The ratings for the eleven generators in RBTS are given in Table 4.4. Bus 1 is defined as the slack bus in load flow analysis of the model. Bus 2 act as slack bus if Bus 1 is isolated. G1 (in Bus 1) is set as the slack generator in this report. The scheduled generation in Bus 2 during peak load, given an intact system situation, is 120 MW.

Table 4.3: Load data for the five load points in RBTS [3]

Load point	Active load [MW]	Reactive load [Mvar]	Priority order ¹ [9]
LP2	20.0	0	1
LP3	85.0	0	5
LP4	40.0	0	3
LP5	20.0	0	2
LP6	20.0	0	4
Total	185.0	0	-

¹LP3 is curtailed first (priority 5), with up to 20% of its load. After this, if necessary, the LP:s is curtailed with up to 20%, following the priority list order.

Table 4.4: Ratings for the eleven generators in RBTS [3]

Generator	Location	Rating [MW]	Capability [Mvar]		Type
			Min	Max	
G1	Bus 1	40	-15	17	Thermal
G2	Bus 1	40	-15	17	Thermal
G3	Bus 1	10	0	7	Thermal
G4	Bus 1	20	-7	12	Thermal
G5	Bus 2	5	0	5	Hydro
G6	Bus 2	5	0	5	Hydro
G7	Bus 2	40	-15	17	Hydro
G8	Bus 2	20	-7	12	Hydro
G9	Bus 2	20	-7	12	Hydro
G10	Bus 2	20	-7	12	Hydro
G11	Bus 2	20	-7	12	Hydro

4.1.3 Reliability data

Table 4.5 shows the component reliability data for the lines, circuit breakers (CB), busbars (BB), transformers and common line towers in RBTS. Table 4.6 shows the reliability data for the generators. The disconnectors (DC) in RBTS are assumed to be ideal. It can be noted that the reliability of the components are relatively poor compared to statistics for existing systems.

Table 4.5: RBTS component reliability data [3]

Transmission Line	
Permanent outage rate [f/yr,km]	0.02
Average outage duration [h]	10
Circuit Breaker	
Active failure rate [f/yr]	0.0066
Passive failure rate ¹ [f/yr]	0.0005
Average outage duration [h]	72
Maintenance outage rate [f/yr]	0.2
Maintenance time [h]	108
Busbar	
Failure rate [f/yr]	0.22
Average outage duration [h]	10
Station Transformer	
Failure rate [f/yr]	0.02
Average outage duration [h]	768
Maintenance outage rate [f/yr]	0.2
Maintenance time [h]	72
Common mode data for towers, L1 and L6	
Failure rate [f/yr]	0.15
Average outage duration [h]	16
Common mode data for towers, L2 and L7	
Failure rate [f/yr]	0.5
Average outage duration [h]	16

¹Unintended opening of circuit breaker

4.1.4 Specification of RBTS(A)

The system setup of RBTS(A) has been chosen to agree with the system in [10]. The result from the analysis in this report can then be verified with the ones from this reference. The load flow method in [10] as well as the type of non-sequential MCS is similar to the technique used in this report. The results should thereby be possible to compare.

Figure 4.1 shows the single line diagram of RBTS(A). This version of RBTS has a single busbar as substation configuration. All outgoing feeders from the busbars include a perfect CB, with a DC at each side.

RBTS(A) only includes reliability data for the transmission lines and the eleven generators. All other components are 100% reliable. The reliability for the load points are included in the presentation, but they are all 100% reliable.

In order to simplify the reading of this report, the reliability data for the lines

Table 4.6: Generator reliability data

Generator g	Outage rate λ_g [f/yr]	Outage duration r_g [h]
G1	0.0299	45
G2	0.0299	45
G3	0.0201	45
G4	0.0250	45
G5	0.0102	45
G6	0.0102	45
G7	0.0201	60
G8	0.0148	55
G9	0.0148	55
G10	0.0148	55
G11	0.0148	55

and generators are presented in a separate table, in the same format that is used later in the report. Table 4.7 shows the component reliability data in the form of the component unavailability U_i in for each component i . The unavailability has no unit here and can be seen has the proportion of the year when the component is unavailable. Equation (4.1) provides the component unavailability with the given data. The number 8760 represent the number of hours per year. The unavailability U_i has for the generators been calculated directly with the data in table 4.6. For the lines the outage rate λ_i is given in outages per year and kilometer. Therefor the line lengths in 4.2 has to be combine with the line reliability data in table 4.5 in order to calculate U_i for the lines.

$$U_i = \frac{\lambda_i r_i}{8760 + \lambda_i r_i} \quad (4.1)$$

The cost data for the generators (c_g) and for the load points c_{lp} are specified in (4.8) and (4.9) respectively. For the generators these have been determined be letting G1 be the regulating unit in bus1 and G7 in bus2. For the load points the costs have been set to a 1000 times higher value than for the generators and with an relation so that the priority of the load points in table 4.3 is preserved. LP3 is in this way going to be disconnected first, since the penalty cost for LP3 is the lowest (1100) amongst the load points.

4.2 Simulation Method

The following simulation methods have been tested, investigated and compared:

Table 4.7: The reliability data for the 25 components, arranged in a vector.

Component k	Name	Unavailability U_k
1	G1	0.0299
2	G2	0.0299
3	G3	0.0201
4	G4	0.0250
5	G5	0.0102
6	G6	0.0102
7	G7	0.0201
8	G8	0.0148
9	G9	0.0148
10	G10	0.0148
11	G11	0.0148
12	LP2	0
13	LP3	0
14	LP4	0
15	LP5	0
16	LP6	0
17	L1	0.0017
18	L2	0.0057
19	L3	0.0045
20	L4	0.0011
21	L5	0.0011
22	L6	0.0017
23	L7	0.0057
24	L8	0.0011
25	L9	0.0011

Table 4.8: The cost for generating power in each generator in RBTS(A).

Generator	Name	Cost [\clubsuit /MW]
g		c_g
1	G1	11
2	G2	9
3	G3	9
4	G4	9
5	G5	9
6	G6	9
7	G7	10
8	G8	8
9	G9	8
10	G10	8
11	G11	8

Table 4.9: The penalty cost for disconnecting power in each load point in RBTS(A).

Load point	Name	Cost [\clubsuit /MW]
lp		c_{lp}
2	LP2	1500
3	LP3	1100
4	LP4	1400
5	LP5	1300
6	LP6	1200

1. Simple sampling
2. Stratified sampling
3. Importance sampling

The preparation and implementation of these four methods are described in section 4.2.1-4.2.3 below. The number of simulations were set to 40 ($sim = 40$) with $n = 10000$ samples in each simulation. In all simulations the Expected Energy Not Served (EENS) (MWh/yr) was used as an index of the system reliability. This measure was chosen since this is one of the indices that converges slowest, so if this has converged the other system indices should also have converged. In the following part of the report, the random variable X stands for EENS, and this is for each scenario i determined by (4.2). This is the total disconnected load in the system multiplied by the numbers of hours per year, which is 8760.

$$x_i = \sum_{lp \in \mathcal{LP}} PU_{lp} \times 8760 \quad (4.2)$$

Many things can go wrong when a model of this size is set up implemented and analyzed. A verification if the results are reasonable are needed in order to ensure that no systematic failures are present. One useful source in this work was the master thesis report by Ran Mo in [10]. This report provides the results for this reports implemented RBTS(A), with as well an even more simplified model of RBTS which was used during the work. These verification results provided a good guideline for the implantation of the model and methods.

4.2.1 Simple sampling

The simple sampling method was implemented in order to determine the benefit of the variance reduction techniques. The estimated expected value from one simulation j is calculated with (4.3). This is the fundamental law of large numbers, which is a very important property in Monte Carlo simulations.

$$m_{x,j} = \frac{1}{n} \sum_{i=1}^n x_i \quad (4.3)$$

The estimated expected value for all simulations are then determined with (4.5).

$$m_x = \frac{1}{sim} \sum_{j=1}^{sim} m_{x,j} \quad (4.4)$$

The expected variance is determined with (4.5).

$$\sigma_X^2 = \frac{1}{sim} \sum_{j=1}^{sim} (m_{x,j} - m_x)^2 \quad (4.5)$$

The implementation of this method can be found in section A.1 in the appendix.

4.2.2 Stratified sampling

As mentioned in section 3.7 one idea would be to make a stratification where the largest generation units and the most important lines are included. As can be seen in figure 4.1 line L9 is of greatest importance for LP6. The system also has three major generation units of 40 MW (G1, G2 and G7) which together stands for approximately half of the total available generation. These four components are therefor used to form the strata tree shown in figure 4.2.

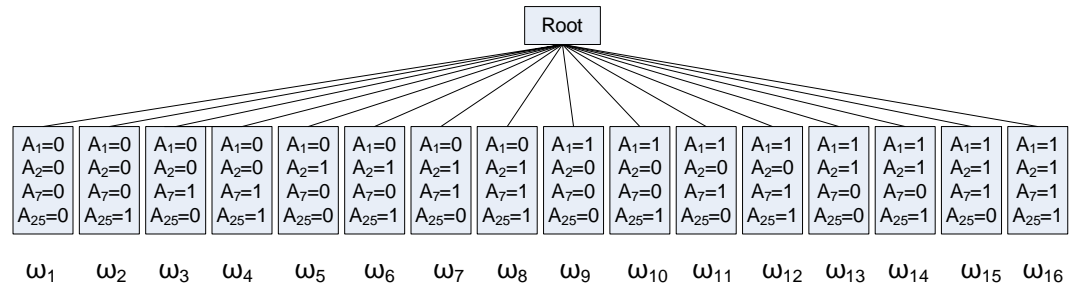


Figure 4.2: Strata tree for the stratified sampling on RBTS(A). The included components are G1, G2, G7 and Line 9, with component number 1,2,7 and 25 in the same order.

The stratum weights w_i in the figure 4.2 are calculated by calculating the probability for each of these 16 states. How stratum weight w_2 is calculate is shown in (4.6) as an example.

$$w_2 = U_1 U_2 U_7 (1 - U_{25}) \quad (4.6)$$

The estimates from each strata, m_{Xh} , gives the total estimate for the system as shown in (4.7).

$$m_x = \sum_{h=1}^{16} w_h m_{Xh} \quad (4.7)$$

In order to determine where the $n = 10000$ samples should be distributed in the way, the Neyman allocation is used [5]. This is shown in (4.8).

$$n_h = n \frac{w_h \sigma_{Xh}}{\sum_{k=1}^{16} w_k \sigma_{Xk}} \quad (4.8)$$

A pilot study of 100 samples per stratum is then used to estimate the $\sigma_{Xh} = \sqrt{\text{Var}[X_h]}$, in order to determine n_h .

The implementation of this method can be found in section A.2 in the appendix.

4.2.3 Importance sampling

As described in section 3.7 this method should be suitable for the simulation of a power system such as RBTS(A). Table 4.7 shows the component unavailability is around 1.5-3% for the generators and 0.1-0.6% for the lines. An approximately value for all components' unavailability is then around 1%. The probability of for an functioning component is set to $p = 0.99$. The probability of selecting the scenario where all 20 components (lines and generators) are functioning is around 80% (p^{20}). The importance sampling function is then chosen to reduce this number of "waste" samples on the intact system to around 5% of the total samples. The system never respond to inputs where all components are functioning and, and almost never on inputs where one component has failed in the system. This approximation should be justified. Since $p_Z^{20} = 0.05$, this result in $p_Z \approx 0.85$. This results in (4.9), which is the importance sampling function for the component availability.

$$f_{Zi}(\Psi) = \begin{cases} 0.15 & \text{if } \Psi = 0 \\ 0.85 & \text{if } \Psi = 1 \\ 0 & \text{all other } \Psi \end{cases} \quad (4.9)$$

By doing this there is a much larger probability of selecting scenarios which results in a non-functioning state of the system. The modification of the random number generation of the inputs is then compensated in the output result by the weighting factors in (4.10).

$$w_i = \prod_{k=1}^{nr-K} \frac{f_Y(\Psi_{k,i})}{f_Z(\Psi_{k,i})} \quad (4.10)$$

In (4.10) $f_Y(\Psi_{k,i})$ can be defined with the help of the data in table 4.7 as shown in (4.11).

$$f_Y(\Psi_k) = \begin{cases} U_k & \text{if } \Psi = 0 \\ 1 - U_k & \text{if } \Psi = 1 \end{cases} \quad (4.11)$$

The implementation of this function can be found in section A.3 in the appendix. For each new sample the weight w_i is calculated and multiplied with the

result, which is the disconnected load PU_{lp} . The sum in 4.12 is calculated for each simulation and the estimates are then based on this.

$$\sum_{i=1}^n w_i x_i \quad (4.12)$$

4.2.4 Software and method implementation

The implementation and all practical testing of the methods have been performed in MATLAB. One drawback of this program is that it is slower than e.g. C++, AMPL or any other appropriate programming language. The major advantage of MATLAB is that it is very easy to implement the methods and to find faults in the code. This is good.

The power system model described in chapter 2 has been implemented as a function in MATLAB. The code for this function is included in appendix A. The output variables from this function are many more than described in section 2.2.3, and this is because it has been used in earlier studies and is reused for this work. The disconnected load in each load point is included in the vector `Shed_P`, which represent the symbol PU_{lp} in this report.

The linear optimization problem is solved with the `linprog` function in MATLAB. This is probably not the most efficient solver on the market, but it works.

One speciality of all implemented methods, is that the program neglects scenarios where all components are functioning. In order to do this the program first check if no load disconnections exists for the intact system before the simulation starts. This procedure safes a large amount of time, since the analysis of this state otherwise would stand for approximately 80-90% of the total simulation time.

4.2.5 Results

The results from the simulations are shown in table 4.10. All simulations' mean value is in good accordance with the reference verification EENS value of 1070 MWh/yr from page 25 in [10]. This reference value may not be more correct than the ones evaluated in this report, but it is a good indicator in the verification of the results.

The accuracy of the stratified sampling is remarkable, compared to simple sampling. The deviation σ of the result is about six times lower than for simple sampling. The maximum and minimum value is in a very small interval. This method is definitely the most efficient of the implemented methods.

The accuracy of importance sampling is not as good as for the stratified sampling. It is better than simple sampling, but not much. The reason for this is not known. The importance function might be re-formulated to give better results.

The time it takes to perform 40 simulates is considerable longer for the stratified and importance sampling. The reason for this is mainly that these have a

Table 4.10: Simulation results for RBTS(A).

Method	Sim. nr	EENS [MWh/yr]				Sim. time [h:min:s]	Total nr of samples [$\times 10^4$]
		Min	Mean	Max	σ		
Verification from [10]	-	-	1070	-	-	-	-
Simple sampling	1	737	1076	1323	126	00:20:22	40
Stratified sampling	3	966	1021	1088	22	01:32:27	40
Importance sampling	4	910	1058	1247	85	01:29:47	40
Stratified sampling 2	5	1003	1037	1080	21	00:23:11	10
Importance sampling 2	6	948	1042	1169	69	00:22:33	10

considerably fewer amount of scenarios were all components are functioning. As described in section 4.2.4 these fully functioning scenarios are not evaluated in the load flow algorithm to save time.

In order to compare the efficiency, two more simulations were made for stratified and importance sampling. In table 4.10 these are named with Stratified sampling 2 and Importance sampling 2. 10 simulations each with 10000 samples gave almost the same computational time as for simple sampling (20 minutes), as seen in table 4.10. Also, it is notable that the result is approximately the same as for the earlier simulations (Sim 3 and Sim 4) for the two methods.

The result from stratified sampling with 10 simulations is comparable with the simple sampling with 40 simulations. The deviation σ is still five times better for stratified sampling. If the same computational time is given to simple sampling and to stratified sampling the efficiency is more than four times better with the later method.

Chapter 5

Conclusions

5.1 General Conclusions

The results in section 4.2.5 shows that the Monte Carlo simulation method can be used for evaluating total system reliability, here quantified with EENS (MWh/yr). The results are in good accuracy in comparison with earlier published material for the case system RBTS(A).

The results from the stratified sampling method shows on a clear improvement in the efficiency of using this method for RBTS(A), compared to simple sampling. The efficiency improvement is remarkable high. With the same computational time as for simple sampling the deviation is about 5 times lower as for stratified sampling.

5.2 Future Work

The customer power demand of RBTS has in this report been set to a constant peak load value. For future work is left to evaluate the reliability indices for an annual load variation. This assessment should be quite straightforward since the power demand is defined as one of the inputs to the presented model. Section 3.8 provides ideas of how different variance reduction techniques can be combined, especially when the load demand is included as an input variable.

Appendix A

Implemented code in Matlab

A.1 Simple sampling code

```
%RBTS Ver 2 (not so simple)
%Lines can fail, Generators can fail

tic
clear all
format short

%-----
%General data for RBTS
%-----
NrNodes = 6;
RefNode = 1; %in this node the angle is 0
NrLines = 9;
NrGen = 11;
NrLP = 5;
%NrLP = 10;

%Generator (both Gen and LP) in which NODE nr
%Bus1 - G1-G4
%Bus2 - G5-G11, LP2(80%),LP2(20%)
%Bus3 - LP3(80%),LP3(20%)
%Bus4 - LP4(80%),LP4(20%)
%Bus5 - LP5(80%),LP5(20%)
%Bus6 - LP6(80%),LP6(20%)
Gen_LP_Node = [1 1 1 1 2 2 2 2 2 2 2 2 2 3 4 5 6];

%Unit 1-11 is generators (0) in in bus 1 and 2
%Unit 12-21 is a load point (1) LP2-LP6
LPGen = [zeros(1,11) ones(1,5)];
%LPGen = [zeros(1,11) ones(1,10)];

%Costs for generators (Priority) (LP is also a gen)
%G1 is slack, G8 is scheduled to 10MW
%G7 is slack if G1 fails?
%20%curtailable part of LP
%Ran Mo prio for LPs (differs from Wijard Wangdee):
Gen_LP_Cost = [11 9 9 9 9 9 10 8 8 8 8 1500 1100 1400 1300 1200];
%Gen_LP_Cost = [11 9 9 9 9 9 10 8 8 8 8 1500 600 1100 200 1400 500 1300 400 1200 300];

%Matrix with the line connections between nodes
```

```

%Line1 , 1->3 ,
%Line2 , 2->4 ,
%Line3 , 1->2 ,
%Line4 , 3->4 ,
%Line5 , 3->5 ,
%Line6 , 1->3 ,
%Line7 , 2->4 ,
%Line8 , 4->5 ,
%Line9 , 5->6 ,
LineDef = [1 3; 2 4; 1 2; 3 4; 3 5; 1 3; 2 4; 4 5; 5 6];

%Per unit base
Sbase = 100; %MVA
Ubase = 230; %kV

%Line reaktances (in ohm per line)
%L1 - L9 , here in pu
XLines_temp_pu = [0.18; 0.6; 0.48; 0.12; 0.12; 0.18; 0.6; 0.12; 0.12];
%XLines_temp_pu = [0.0342+i*0.18; 0.114+i*0.6; 0.0912+i*0.48; 0.0228+i*0.12; 0.0228+i*0.12; 0.0
%Convert to in ohm per line)
Zb = Ubase^2/Sbase;
XLines = XLines_temp_pu*Zb;

%Generator- or load-nodes, max and min active power (MW)
%Unit 1-11 generators, 12-16 LPs
Gen_LP_MaxP = [40; 40; 10; 20; 5; 5; 40; 20; 20; 20; 20; 0; 0; 0; 0; 0
Gen_LP_MinP = [ 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; -20; -85; -40; -20; -20
%Line max and min active power capacity (MW)
LineMaxP = [85; 71; 71; 71; 71; 85; 71; 71; 71];
LineMinP = -LineMaxP;

%Reliability data
%Forced outage rate in, (Unavailability over the year)
%for the generators (bot Gen and LP) and lines
lambda = [6 6 4 5 2 2 3 2.4 2.4 2.4 2.4 -1*ones(1,5) 1.5 5 4 1 1 1.5 5 1 1];
r = [ 45 45 45 45 45 45 60 55 55 55 55 zeros(1,5) 10 10 10 10 10 10 10 10 10];
FOR_comp = (lambda.*r)./(8760+(lambda.*r));

%-----
%End of system data
%-----

%-----
%Simulation data
%-----
%Minimum simulations
min_nr_sim = 40;
%Minimum coefficient of variance
aY_min = 1.0;

%Number of samples per batch (per sim)
nr_samples = 10000;

hours_per_year = 8760;

%-----
%End of simulation data
%-----

```

```

%-----
%Print out system info
%-----
disp(' ')
disp(' ')
disp('*****')
disp('-----')
disp('  RBTS MCS Version 2.0 (Lines and Generators can fail)')
disp('-----')
disp('*****')
disp(' ')

disp('General system load flow:')
%Run one intact system load flow (all comp ok)
CompStatus = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
[angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflowVer3(NrNodes, RefNode);

disp('exitflag:')
disp(exitflag)

if sum(Shed_P)>0
    disp('Intact system setup has load curtailments: (MW)');
    disp(Shed_P);
else
    disp('General load flow of system is OK. No load shed.')
end
disp(' ')

%Set a new random state
rand('state', sum(100*clock))

%Error control
error_vec=[];
error_state_vec=[];

%Data vectors
Nr_Curtailments_LP_vec = [];
PLC_vec=[];
PLC_LP_vec=[];
EDNS_vec=[];
EDNS_LP_vec=[];
EENS_vec=[];
EENS_LP_vec=[];
var_EENS_vec = [];
aY_vec = [];

temp = [];
%m_EENS = 0;
%current_nr_samples=0;
nr_sim=0;
aY=0;

%Start Sim
while ((nr_sim<min_nr_sim) || (aY>aY_min))
    nr_sim = nr_sim + 1;

    %New data vectors
    Total_Shed_LP_P = zeros(1,NrLP);
    Nr_Curtailments_LP = zeros(1,NrLP);

    %Keep record of errors
    error=0;

```

```

for n=1:nr_samples

%-----
%Step 1, select system state x
%-----
%1a Generate uniform random numbers for the components
random_numbers = rand(1,NrGen+NrLP+NrLines);
%1b Compare these with the define values, if the generated values are
%less - > give the component a 0 (down), else 1 (up)
CompStatus = (random_numbers > FOR_comp)';

%-----
%Step 2, Calculate F(x) for the selected state
%-----
%Skip DC load flow if system intact (we know that no load is shed)
if (sum(CompStatus)<(NrGen+NrLP+NrLines))
    %Use DC_loadflow to check if this state means any load curtailments
    [angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflow
else
    exitflag = 1;
    Shed_P = 0;
end

%ERROR Control
%Keep control of the errors (no convergence)
if (exitflag ~= 1)
    error=error+1;
    %Only check lines first
    error_state_vec=[error_state_vec; CompStatus(22:30)'];
end

%-----
%Step 3, Update the estimates
%-----

%Only required if there is a load curtailment
if sum(Shed_P)>0
    %Update curtailment energy (Shed_P is a vector for each LP)
    Total_Shed_LP_P = Total_Shed_LP_P + Shed_P';

    %Update curtailments per LP
    Nr_Curtailments_LP = Nr_Curtailments_LP + (Shed_P'>0);

    if Shed_P(2)>0
        temp = [temp; CompStatus'];
    end
end

end %end sample

%Total nr of curtailments
Nr_Curtailments_LP_vec = [Nr_Curtailments_LP_vec; Nr_Curtailments_LP];

%PLC = Probability of load curtailment
%First per LP
PLC_LP_sample = Nr_Curtailments_LP/nr_samples;
PLC_LP_vec=[PLC_LP_vec; PLC_LP_sample];
%Then total system

```

```

PLC_sample = sum(Nr_Curtailments_LP)/nr_samples;
PLC_vec=[PLC_vec; PLC_sample];

%EDNS = Expected demand not supplied (MW)
%First per LP
EDNS_LP_sample = Total_Shed_LP_P/nr_samples;
EDNS_LP_vec=[EDNS_LP_vec; EDNS_LP_sample];
%Then total system
EDNS_sample = sum(Total_Shed_LP_P)/nr_samples;
EDNS_vec=[EDNS_vec; EDNS_sample];

%EENS
EENS_sample = EDNS_sample*hours_per_year;
EENS_vec=[EENS_vec; EENS_sample];
EENS_LP_vec=[EENS_LP_vec; EDNS_LP_sample*hours_per_year];
% $m_{EENS} = \text{sum}(EENS\_vec) / \text{sim}$ ;

%variation of EENS until this sim
var_EENS_estimate = (1/nr_sim)*sum(EENS_vec.^2)-((1/nr_sim)*sum(EENS_vec))^2;
var_EENS_vec = [var_EENS_vec; var_EENS_estimate];

%var_mY =
aY = sqrt(var_EENS_estimate)/((1/nr_sim)*sum(EENS_vec)*sqrt(nr_sim));
aY_vec =[aY_vec; aY];

%Error control
error_vec = [error_vec; error];

disp('---')
disp('Simulation:')
disp(nr_sim)
disp('EENS:')
disp(EENS_sample)
disp('sigma[EENS]:')
disp(sqrt(var_EENS_estimate))
disp('Var[EENS]:')
disp(var_EENS_estimate)
disp('Coefficient of variance:')
disp(aY)
disp('---')

end %end sim

%PLC estimate
%First per LP
PLC_LP_estimate = sum(PLC_LP_vec)/nr_sim;
%Then total system
PLC_estimate = sum(PLC_vec)/nr_sim;

%EDNS estimate
%First per LP
EDNS_LP_estimate = sum(EDNS_LP_vec,1)/nr_sim;
%Then total system
EDNS_estimate = sum(EDNS_vec)/nr_sim;

%EENS estimate (Expected Energy not supplied)
%First per LP
EENS_LP_estimate = sum(EENS_LP_vec,1)/nr_sim;
%Then total system
EENS_estimate = sum(EENS_vec)/nr_sim;

disp('Nr of curtailments per LP:')

```

```

disp(Nr_Curtailments_LP_vec)

disp('PLC (Probability of load curtailment), For each load point LP2-LP6:')
fprintf(1, '%10.5f\n', PLC_LP_estimate);
disp('PLC (Probability of load curtailment), Total System:')
fprintf(1, '%10.5f\n', PLC_estimate);

disp('EDNS (Expected demand not supplied) [MW], For each load point LP2-LP6:')
fprintf(1, '%10.5f\n', EDNS_LP_estimate);
disp('EDNS (Expected demand not supplied) [MW], Total System:')
fprintf(1, '%10.5f\n', EDNS_estimate);

disp('EENS (Expected energy not supplied) [MWh/yr], For each load point LP2-LP6:')
fprintf(1, '%10.5f\n', EENS_LP_estimate);
disp('EENS (Expected energy not supplied) [MWh/yr], Total System:')
fprintf(1, '%10.5f\n', EENS_estimate);

disp('min EENS sample:')
fprintf(1, '%10.5f\n', min(EENS_vec));

disp('max EENS sample:')
fprintf(1, '%10.5f\n', max(EENS_vec));

disp('Var[EENS] (Variance of EENS), For all samples, after Simulation i:')
fprintf(1, '%10.2f\n', var_EENS_vec);

disp('Coefficient of Variance, based on EENS:')
fprintf(1, '%10.4f\n', aY_vec);

plot(1:nr_sim, aY_vec)

%Error Control
if(sum(error_vec)>0)
    disp('-----')
    disp('*****')
    disp('One or more errors in DC load flow!')
    disp('*****')
    disp('-----')
end
%disp(error_vec)
%disp(error_state_vec)

disp('Total nr of samples:')
disp(nr_sim*nr_samples);
disp('Total simulation time: (h:min:s)')
sim_time = toc;
nr_hours = floor(sim_time/3600);
nr_minutes = floor((sim_time-nr_hours*3600)/60);
nr_seconds = ceil(sim_time-nr_hours*3600-nr_minutes*60);
fprintf(1, '%2.0f:%2.0f:%2.0f\n', nr_hours, nr_minutes, nr_seconds);
%disp(toc);
disp('-----')
disp('The End')
disp('-----')

```

A.2 Stratified sampling code

```

%RBTS Ver 2 (not so simple)
%Lines can fail, Generators can fail
%Stratified sampling

tic
clear all
format short

%-----
%General data for RBTS
%-----
NrNodes = 6;
RefNode = 1; %in this node the angle is 0
NrLines = 9;
NrGen = 11;
NrLP = 5;
%NrLP = 10;

%Generator (both Gen and LP) in which NODE nr
%Bus1 - G1-G4
%Bus2 - G5-G11, LP2(80%),LP2(20%)
%Bus3 - LP3(80%),LP3(20%)
%Bus4 - LP4(80%),LP4(20%)
%Bus5 - LP5(80%),LP5(20%)
%Bus6 - LP6(80%),LP6(20%)
Gen_LP_Node = [1 1 1 1 2 2 2 2 2 2 2 2 3 4 5 6];

%Unit 1-11 is generators (0) in in bus 1 and 2
%Unit 12-21 is a load point (1) LP2-LP6
LPGen = [zeros(1,11) ones(1,5)];

%Costs for generators (Priority) (LP is also a gen)
%G1 is slack, G8 is scheduled to 10MW
%G7 is slack if G1 fails?
%20%curtailable part of LP
%Ran Mo prio for LPs (differs from Wijard Wangdee):
Gen_LP_Cost = [11 9 9 9 9 9 10 8 8 8 8 1500 1100 1400 1300 1200];

%Matrix with the line connections between nodes
%Line1 , 1->3 ,
%Line2 , 2->4 ,
%Line3 , 1->2 ,
%Line4 , 3->4 ,
%Line5 , 3->5 ,
%Line6 , 1->3 ,
%Line7 , 2->4 ,
%Line8 , 4->5 ,
%Line9 , 5->6 ,
LineDef = [1 3; 2 4; 1 2; 3 4; 3 5; 1 3; 2 4; 4 5; 5 6];

%Per unit base
Sbase = 100; %MVA
Ubase = 230; %kV

%Line reaktances (in ohm per line)
%L1 - L9 , here in pu
XLines_temp_pu = [0.18; 0.6; 0.48; 0.12; 0.12; 0.18; 0.6; 0.12; 0.12];
%Convert to in ohm per line)
Zb = Ubase^2/Sbase;

```



```

disp('exitflag:')
disp(exitflag)

if sum(Shed_P)>0
    disp('Intact system setup has load curtailments: (MW)');
    disp(Shed_P);
else
    disp('General load flow of system is OK. No load shed.')
end
disp(' ')
%disp(Shed_P);
%disp('Angles (grades):')
%disp(angNodes)
%disp('Production in each unit (MW):')
%disp(P_Unit)
%disp('Load in each node (MW) (Sum up all LPs per node):')
%disp(P_LP_Node)
%disp('Generation in each node (MW) (Sum up all Gen per node):')
%disp(P_Gen_Node)
%disp('LineFlows (MW):')
%disp(LineFlows)

%Set a new random state
rand('state', sum(100+clock))

%Error control
error_vec=[];
error_state_vec=[];

%Data vectors
Nr_Curtailments_LP_vec = [];
PLC_vec=[];
PLC_LP_vec=[];
EDNS_vec=[];
EDNS_LP_vec=[];
EENS_vec=[];
EENS_LP_vec=[];
var_EENS_vec = [];
aY_vec = [];

temp = [];
%m_EENS = 0;
%current_nr_samples=0;
nr_sim=0;
aY=0;

%*****
%Start to determine the stratum weights
%G1,G2,G7, L9 -> 16 stratum
w_i = [];
u1 = FOR_comp(1);
u2 = FOR_comp(2);
u3 = FOR_comp(7);
u4 = FOR_comp(25);
a1=1-u1;
a2=1-u2;
a3=1-u3;
a4=1-u4;
%Enumerate to calc w_i
stratum_vec=[];
for i1=0:1
    for i2=0:1

```

```

        for i3=0:1
            for i4=0:1
                w_i=[w_i; (i1*a1+(1-i1)*u1)*(i2*a2+(1-i2)*u2)*(i3*a3+(1-i3)*u3)*(i4*a4+(1-i4)*u4);
                    stratum_vec=[stratum_vec; i1 i2 i3 i4];
            end
        end
    end
end
end
%*****

%Start Sim
total_nr_samples = 0;
while ((nr_sim<min_nr_sim) || (aY>aY_min))
    nr_sim = nr_sim + 1;

    %New data vectors
    Total_Shed_LP_P = zeros(1,NrLP);
    Nr_Curtailments_LP = zeros(1,NrLP);

    %Keep record of errors
    error=0;

    %Start with pilot study of 100 samples per stratum
    %in order to estimate s2_Xh
    %my_
    %*****
    s2_Xh_pilot_vec = [];
    m_Xh_pilot_vec = [];
    %x_hi_tot=[];
    test_samples = 0;
    for j =1:nr_strata
        if known_mXh(j) ~= 1
            x_hi = [];
            for n=1:pilot_samples
                %Simple sampling for these
                random_numbers = rand(1,NrGen+NrLP+NrLines);
                CompStatus = (random_numbers > FOR_comp)';
                %Use the properties from the stratum j
                CompStatus(1)= stratum_vec(j,1);
                CompStatus(2)= stratum_vec(j,2);
                CompStatus(7)= stratum_vec(j,3);
                CompStatus(25)= stratum_vec(j,4);

                %Use DC_loadflow to check if this state means any load curtailments
                [angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflow(CompStatus);

                EENS_est = sum(Shed_P)*hours_per_year;
                x_hi = [x_hi; EENS_est];

                test_samples = test_samples + 1;
            end
            %x_hi_tot=[x_hi_tot; x_hi];
            m_Xh_value = (1/pilot_samples)*sum(x_hi);
            s2_Xh = (1/pilot_samples)*sum((x_hi-m_Xh_value).^2);
        else
            %In these stratum we know that EENS = 0;
            m_Xh_value = 0;
            s2_Xh = 0;
        end
    end
end

```

```

    m_Xh_pilot_vec = [m_Xh_pilot_vec; m_Xh_value];
    s2_Xh_pilot_vec = [s2_Xh_pilot_vec; s2_Xh];
    total_nr_samples = total_nr_samples + pilot_samples;
end

%Neyman allocation
n_rest = (nr_samples_tot-test_samples);

sum_neyman = w_i'*s2_Xh_pilot_vec;
n_h = round((n_rest/sum_neyman)*(w_i.*s2_Xh_pilot_vec));

%This is the vector with the estimate values for 9860 samples
%of EENS
m_Xh_vec = [];

%Start the real one, now then we have the Neyman
for strata = 1:nr_strata
    sum_x_h = 0;
    %m_Xh = 0;
    if n_h(strata)>0
        for strata_sample = 1:n_h(strata)
            random_numbers = rand(1,NrGen+NrLP+NrLines);
            CompStatus = (random_numbers > FOR_comp)';
            %Use the properties from the stratum j
            CompStatus(1)= stratum_vec(strata,1);
            CompStatus(2)= stratum_vec(strata,2);
            CompStatus(7)= stratum_vec(strata,3);
            CompStatus(25)= stratum_vec(strata,4);

            %Skip DC load flow if system intact (we know that no load is shed)
            if (sum(CompStatus)<(NrGen+NrLP+NrLines))
                %Use DC_loadflow to check if this state means any load curtailments
                [angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflowV
            else
                exitflag = 1;
                Shed_P = 0;
            end

            %ERROR Control
            %Keep control of the errors (no convergence)
            if (exitflag ~= 1)
                error=error+1;
                %Only check lines first
                error_state_vec=[error_state_vec; CompStatus(22:30)'];
            end

            sum_x_h = sum_x_h + sum(Shed_P)*hours_per_year;

        end
        %Determine the estimate for this strata
        m_Xh = (1/n_h(strata))*sum_x_h;
    else
        %Take the estimate vaule - if no sample in this strata
        m_Xh = m_Xh_pilot_vec(strata);
    end
    %Collect the stratum estimates
    m_Xh_vec = [m_Xh_vec; m_Xh];

    %Update the nr of samples
    total_nr_samples = total_nr_samples + n_h(strata);
end
end

```

```

%Multiply with the stratum weights w_i
EENS_sample = m_Xh_vec'*w_i;

%EENS
EENS_vec=[EENS_vec; EENS_sample];
%EENS_LP_vec=[EENS_LP_vec; EDNS_LP_sample*hours_per_year];
%m_EENS = sum(EENS_vec)/sim;

%variation of EENS until this sim
var_EENS_estimate = (1/nr_sim)*sum(EENS_vec.^2)-((1/nr_sim)*sum(EENS_vec))^2;
var_EENS_vec = [var_EENS_vec; var_EENS_estimate];

%var_mY =
aY = sqrt(var_EENS_estimate)/((1/nr_sim)*sum(EENS_vec)*sqrt(nr_sim));
aY_vec =[aY_vec; aY];

%Error control
error_vec = [error_vec; error];

disp('---')
disp('Simulation:')
disp(nr_sim)
disp('EENS:')
disp(EENS_sample)
disp('sigma[EENS]:')
disp(sqrt(var_EENS_estimate))
disp('Var[EENS]:')
disp(var_EENS_estimate)
disp('Coefficient of variance:')
disp(aY)
disp('---')

end %end sim

%EENS estimate (Expected Energy not supplied)
%Then total system
EENS_estimate = sum(EENS_vec)/nr_sim;

disp('EENS (Expected energy not supplied) [MWh/yr], Total System:')
fprintf(1,'%10.5f\n',EENS_estimate);

disp('min EENS sample:')
fprintf(1,'%10.5f\n',min(EENS_vec));

disp('max EENS sample:')
fprintf(1,'%10.5f\n',max(EENS_vec));

disp('Var[EENS] (Variance of EENS), For all samples, after Simulation i:')
fprintf(1,'%10.2f\n',var_EENS_vec);

disp('sigma[EENS]:')
disp(sqrt(var_EENS_vec(length(var_EENS_vec))))

disp('Coefficient of Variance, based on EENS:')
fprintf(1,'%10.4f\n',aY_vec);

plot(1:nr_sim, aY_vec)

%Error Control
if(sum(error_vec)>0)

```

```

disp('-----')
disp('*****')
disp('One or more errors in DC load flow!')
disp('*****')
disp('-----')
end
%disp(error_vec)
%disp(error_state_vec)

disp('Total nr of samples:')
disp(total_nr_samples);
disp('Total simulation time: (h:min:s)')
sim_time = toc;
nr_hours = floor(sim_time/3600);
nr_minutes = floor((sim_time-nr_hours*3600)/60);
nr_seconds = ceil(sim_time-nr_hours*3600-nr_minutes*60);
fprintf(1,'%2.0f:%2.0f:%2.0f\n',nr_hours,nr_minutes,nr_seconds);
%disp(toc);
disp('-----')
disp('The End')
disp('-----')

```

A.3 Importance sampling code

```

%RBTS Ver 2 (not so simple)
%Lines can fail, Generators can fail

tic
clear all
format short

%-----
%General data for RBTS
%-----
NrNodes = 6;
RefNode = 1; %in this node the angle is 0
NrLines = 9;
NrGen = 11;
NrLP = 5;

%Generator (both Gen and LP) in which NODE nr
%Bus1 - G1-G4
%Bus2 - G5-G11, LP2(80%),LP2(20%)
%Bus3 - LP3(80%),LP3(20%)
%Bus4 - LP4(80%),LP4(20%)
%Bus5 - LP5(80%),LP5(20%)
%Bus6 - LP6(80%),LP6(20%)
Gen_LP_Node = [1 1 1 1 2 2 2 2 2 2 2 2 3 4 5 6];

%Unit 1-11 is generators (0) in in bus 1 and 2
%Unit 12-21 is a load point (1) LP2-LP6
LPGen = [zeros(1,11) ones(1,5)];
%LPGen = [zeros(1,11) ones(1,10)];

%Costs for generators (Priority) (LP is also a gen)
%G1 is slack, G8 is scheduled to 10MW
%G7 is slack if G1 fails?
%Ran Mo prio for LPs (differs from Wijard Wangdee):
Gen_LP_Cost = [11 9 9 9 9 9 10 8 8 8 8 8 1500 1100 1400 1300 1200];

```

```

%Matrix with the line connections between nodes
%Line1 , 1->3 ,
%Line2 , 2->4 ,
%Line3 , 1->2 ,
%Line4 , 3->4 ,
%Line5 , 3->5 ,
%Line6 , 1->3 ,
%Line7 , 2->4 ,
%Line8 , 4->5 ,
%Line9 , 5->6 ,
LineDef = [1 3; 2 4; 1 2; 3 4; 3 5; 1 3; 2 4; 4 5; 5 6];

%Per unit base
Sbase = 100; %MVA
Ubase = 230; %kV

%Line reaktances (in ohm per line)
%L1 - L9 , here in pu
XLines_temp_pu = [0.18; 0.6; 0.48; 0.12; 0.12; 0.18; 0.6; 0.12; 0.12];
%XLines_temp_pu = [0.0342+i*0.18; 0.114+i*0.6; 0.0912+i*0.48; 0.0228+i*0.12; 0.0228+i*0.12; 0.0
%Convert to in ohm per line)
Zb = Ubase^2/Sbase;
XLines = XLines_temp_pu*Zb;

%Generator- or load-nodes, max and min active power (MW)
%Unit 1-11 generators, 12-16 LPs
Gen_LP_MaxP = [40; 40; 10; 20; 5; 5; 40; 20; 20; 20; 20; 0; 0; 0; 0; 0
Gen_LP_MinP = [ 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; -20; -85; -40; -20; -20

%Line max and min active power capacity (MW)
LineMaxP = [85; 71; 71; 71; 71; 85; 71; 71; 71];
LineMinP = -LineMaxP;

%Reliability data
%Forced outage rate in, (Unavailability over the year)
%for the generators (bot Gen and LP) and lines
lambda = [6 6 4 5 2 2 3 2.4 2.4 2.4 2.4 -1*ones(1,5) 1.5 5 4 1 1 1.5 5 1 1];
r = [ 45 45 45 45 45 45 60 55 55 55 55 zeros(1,5) 10 10 10 10 10 10 10 10 10];
%FOR_comp = (lambda.*r)/8760
FOR_comp = (lambda.*r)./(8760+(lambda.*r));

%-----
%End of system data
%-----

%-----
%Simulation data
%-----
%Minimum simulations
min_nr_sim = 10;
%Minimum coefficient of variance
aY_min = 1.0;

%Number of samples per batch (per sim)
nr_samples = 10000;

hours_per_year = 8760;

```

```

%-----
%End of simulation data
%-----

%-----
%Print out system info
%-----
disp(' ')
disp(' ')
disp('*****')
disp('-----')
disp(' RBTS MCS Version 2.0 (Lines and Generators can fail)')
disp('-----')
disp('***** Importance Sampling *****')
disp(' ')

disp('General system load flow:')
%Run one intact system load flow (all comp ok)
CompStatus = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1];
[angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflowVer3(NrNodes, RefNode);

disp('exitflag:')
disp(exitflag)

if sum(Shed_P)>0
    disp('Intact system setup has load curtailments: (MW)');
    disp(Shed_P);
else
    disp('General load flow of system is OK. No load shed.')
end

%Set a new random state
rand('state', sum(100*clock))

%Error control
error_vec=[];
error_state_vec=[];

%Data vectors
Nr_Curtailments_LP_vec = [];
PLC_vec=[];
PLC_LP_vec=[];
EDNS_vec=[];
EDNS_LP_vec=[];
EENS_vec=[];
EENS_LP_vec=[];
var_EENS_vec = [];
aY_vec = [];

temp = [];
nr_sim=0;
aY=0;

FOR_comp_z = [0.15*ones(1,11) zeros(1,5) 0.15*ones(1,9)];

%Start Sim
while ((nr_sim<min_nr_sim) || (aY>aY_min))
    nr_sim = nr_sim + 1;

    %New data vectors
    Total_Shed_LP_P = zeros(1,NrLP);

```

```

Nr_Curtailments_LP = zeros(1,NrLP);

%Keep record of errors
error=0;

sum_x_h = 0;
%w_i_all_samples = [];

for n=1:nr_samples

%-----
%Step 1, select system state x
%-----
%1a Generate uniform random numbers for the components
random_numbers = rand(1,NrGen+NrLP+NrLines);
%1b Compare these with the define values, if the generated values are
%less - > give the component a 0 (down), else 1 (up)

%OBS importance sampling
CompStatus = (random_numbers > FOR_comp_z)';
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Determine weights
fy=1;
fz=1;
for k=1:length(CompStatus)
    fy = fy*(CompStatus(k)*(1-FOR_comp(k)) + (1-CompStatus(k))*FOR_comp(k));
    fz = fz*(CompStatus(k)*(1-FOR_comp_z(k)) + (1-CompStatus(k))*FOR_comp_z(k));
end
w_i = fy/fz;
%w_i_all_samples = [w_i_all_samples; fy/fz];

%-----
%Step 2, Calculate F(x) for the selected state
%-----
%Skip DC load flow if system intact (we know that no load is shed)
if (sum(CompStatus)<(NrGen+NrLP+NrLines))
    %Use DC_loadflow to check if this state means any load curtailments
    [angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflow
else
    exitflag = 1;
    Shed_P = 0;
end

%ERROR Control
%Keep control of the errors (no convergence)
if (exitflag ~= 1)
    error=error+1;
    %Only check lines first
    error_state_vec=[error_state_vec; CompStatus(22:30)'];
end

%-----
%Step 3, Update the estimates
%-----

%Include weight w_i
sum_x_h = sum_x_h + w_i*sum(Shed_P)*hours_per_year;

```

```

end %end sample

%Multiply with the stratum weights w_i
EENS_sample = sum_x_h/nr_samples;

%EENS
EENS_vec=[EENS_vec; EENS_sample];

%variation of EENS until this sim
var_EENS_estimate = (1/nr_sim)*sum(EENS_vec.^2)-((1/nr_sim)*sum(EENS_vec))^2;
var_EENS_vec = [var_EENS_vec; var_EENS_estimate];

%var_mY =
aY = sqrt(var_EENS_estimate)/((1/nr_sim)*sum(EENS_vec)*sqrt(nr_sim));
aY_vec =[aY_vec; aY];

%Error control
error_vec = [error_vec; error];

disp('---')
disp('Simulation:')
disp(nr_sim)
disp('EENS:')
disp(EENS_sample)
disp('sigma[EENS]:')
disp(sqrt(var_EENS_estimate))
disp('Var[EENS]:')
disp(var_EENS_estimate)
disp('Coefficient of variance:')
disp(aY)
disp('---')

end %end sim

%EENS estimate (Expected Energy not supplied)
%Then total system
EENS_estimate = sum(EENS_vec)/nr_sim;

disp('EENS (Expected energy not supplied) [MWh/yr], Total System:')
fprintf(1,'%10.5f\n',EENS_estimate);

disp('min EENS sample:')
fprintf(1,'%10.5f\n',min(EENS_vec));

disp('max EENS sample:')
fprintf(1,'%10.5f\n',max(EENS_vec));

disp('Var[EENS] (Variance of EENS), For all samples, after Simulation i:')
fprintf(1,'%10.2f\n',var_EENS_vec);

disp('sigma[EENS]:')
disp(sqrt(var_EENS_vec(length(var_EENS_vec))))

disp('Coefficient of Variance, based on EENS:')
fprintf(1,'%10.4f\n',aY_vec);

plot(1:nr_sim, aY_vec)

%Error Control
if(sum(error_vec)>0)

```

```

disp('-----')
disp('*****')
disp('One or more errors in DC load flow!')
disp('*****')
disp('-----')
end
%disp(error_vec)
%disp(error_state_vec)

disp('Total nr of samples:')
disp(nr_sim*nr_samples);
disp('Total simulation time: (h:min:s)')
sim_time = toc;
nr_hours = floor(sim_time/3600);
nr_minutes = floor((sim_time-nr_hours*3600)/60);
nr_seconds = ceil(sim_time-nr_hours*3600-nr_minutes*60);
fprintf(1, '%2.0f:%2.0f:%2.0f\n', nr_hours, nr_minutes, nr_seconds);
%disp(toc);
disp('-----')
disp('The End')
disp('-----')

```

A.4 The DC load flow method code

```

%Ver2
%General Function for DC loadflow using linprog
%Ver2 also include separate Generators connected to one bus
%e.g. Bus1 has 4 independent Generators

%(linear programming).
%Input is:
% + NrNodes - The total number of nodes (buses) in the system
% + RefNode - The reference node where the voltage angle is zero
% + NrLines - The total number of lines that combines the nodes
% + Gen_LP_Node - Vector indicating in which node where each unit (Gen or LP) is
% + LPGen - Vector indicating 0=(Generator), 1=load point
% + NrGen - The number of generators
% + NrLP - The number of load points
% + Gen_LP_Cost - Vector with the Generators and Load Points costs (prio).A load has a very high prio
% + LineDef
% + XLines
% + Sbase
% + Ubase
% + Gen_LP_MaxP - The max of the generator- or LP prod intervall
% + Gen_LP_MinP - The min of the generator- or LP prod intervall
% + LineMaxP - The Max limit (MW) for the line
% + LineMinP - The Min limit for (MW) the line
% + CompStatus - 1=in function, 0=non-function

%Output is:

function [angNodes, P_Unit, P_LP_Node, P_Gen_Node, LineFlows, Shed_P, exitflag] = DC_loadflowVe

%Total number of units (Gen+LP)
NrGen_LP = NrGen + NrLP;

%Total number of components (gen+lines)

```

```

NrTotal = NrGen_LP+NrLines;

%Line reaktances (real ohm)
%L1, L2, L3 .. in XLines
%Convert to per units
Zb = Ubase^2/Sbase;
Xpu = XLines/Zb;
%Admittances in vector
Ypu = 1./Xpu;

%Get the status vector for the generators and lines
StatusGen = CompStatus(1:NrGen_LP);
StatusLine = CompStatus((NrGen_LP+1):NrTotal);

%Generator (and load) max and min active power capacity in per unit
Gen_LP_MaxP_pu = Gen_LP_MaxP./Sbase;
Gen_LP_MinP_pu = Gen_LP_MinP./Sbase;

%Line limits in per unit
LineMaxP_pu = LineMaxP./Sbase;
LineMinP_pu = LineMinP./Sbase;

%Set the real status of the topology (lines may be missing)
%in the Y matrix
%Build admittance matrix Y_mat
Y_mat = zeros(NrNodes);
for j = 1:NrLines
    from = LineDef(j,1);
    to = LineDef(j,2);
    %Include the admittances
    %Check status of line also
    Y_mat(from,to) = Y_mat(from,to) - Ypu(j)*StatusLine(j);
    Y_mat(to,from) = Y_mat(to,from) - Ypu(j)*StatusLine(j);
end

%the diagonal is the neg sum of row
for i=1:size(Y_mat)
    Y_mat(i,i) = -sum(Y_mat(i,:));
end

%Matrix with the power in the lines
Pline_mat = zeros(NrNodes);
for j = 1:NrLines
    from = LineDef(j,1);
    to = LineDef(j,2);
    Pline_mat(j,from) = Ypu(j);
    Pline_mat(j,to) = -Ypu(j);
end

%The reference angle is zero in node Ref_Node (normally node 1)
%Delete this column, Y_red = Y_reduced matrix
Y_red = [Y_mat(:,1:(RefNode-1)) Y_mat(:,(RefNode+1):NrNodes)];

%The reference angle is zero in node Ref_Node (normally node 1)
%Delete this column in PLine
Pline_red = [Pline_mat(:,1:(RefNode-1)) Pline_mat(:,(RefNode+1):NrNodes)];

%Set the real status of generators (and LP:s)
%Generator may be down = 0...
Gen_LP_MaxP_pu_status = Gen_LP_MaxP_pu.*StatusGen;
Gen_LP_MinP_pu_status = Gen_LP_MinP_pu.*StatusGen;

```

```

>Delete lines that are missing (rows)
%Lines may be down = 0...
Pline_status = [];
PLmax_status = [];
PLmin_status = [];
for j = 1:NrLines
    if StatusLine(j) == 1
        Pline_status = [Pline_status; Pline_red(j,:)];
        PLmax_status = [PLmax_status; LineMaxP_pu(j,:)];
        PLmin_status = [PLmin_status; LineMinP_pu(j,:)];
    end
end

%Objective function
%Minimize the production cost in the Generators and LP:s
%The variables x(1),x(2),...,x(NrGen_LP), is the production in each unit
%The variables x(NrGen_LP+1), ... , x(NrGen_LP+NrNodes-1) (-1 is for the
%ref node), is for the angles in Nodes
%For the test system x1-x4, x5-x6
%The objective function's matrix is then(4 units, 3 buses):
%[c1 c2 c3 c4 0 0]
objectivef = [Gen_LP_Cost zeros(1, NrNodes-1)];

%Constraints - The nodes
%The production in the units(Gen or LP) - (minus) the flow out from the
%node (line flows) is equal to zero
%1 - Determine the matrix that relates the Units (Gen or LP) to each Node
A_eq_Prod_Gen = zeros(NrNodes,NrGen_LP);
for i = 1:NrGen_LP
    in_node = Gen_LP_Node(i);
    A_eq_Prod_Gen(in_node,i) = 1;
end
%2 - Determine the matrix that relates the line flows to x5, x6...
% This is in fact the reduced Y matrix, Y_red
A_eq_flows = Y_red;
%3 - The production and the line flows in each node must end up in zero
A_eq_flows = -A_eq_flows;
%4 - The resulting constraints for the nodes , A_eq
A_eq = [A_eq_Prod_Gen A_eq_flows];
%5 - This has to be equal to zero
b_eq = zeros(NrNodes, 1);

%Constraints -
%The flow for each line has to be between min and max
[NrLines_in_function tmp] = size(Pline_status);
A_line = [zeros(NrLines_in_function,NrGen_LP) Pline_status];
A = [A_line;
    -A_line];

b = [PLmax_status; -PLmin_status];

%Generator and Angle restrictions
ub_gen = [Gen_LP_MaxP_pu_status];
lb_gen = [Gen_LP_MinP_pu_status];

ub_angle = ones(NrNodes-1,1)*(pi/2);
lb_angle = -ones(NrNodes-1,1)*(pi/2);

ub = [ub_gen; ub_angle];

```

```

lb = [lb_gen; lb_angle];

options = optimset('Display', 'off');

%Start the opt method in Matlab
[x_results,fval,exitflag,output,lambda] = linprog(objectivef,A,b,A_eq,b_eq,lb,ub,[],options);

if(exitflag==1)
    %Sort out the result from this
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Get the Production of P in the units (Gen and LP) in p.u
    P_Gen_LP = x_results(1:NrGen_LP);

    %Add the reference node to the solution (angle =0)
    %sort out the angles and set the ref node to 0
    angles = [x_results((NrGen_LP+1):(NrGen_LP+RefNode-1)); 0; x_results(NrGen_LP+RefNode:NrGen_LP+NrNodes)];

    %Calculate the Total Generation and Total Load in each Node
    P_Gen_Node = zeros(NrNodes,1);
    P_LP_Node = zeros(NrNodes,1);
    LPShed = zeros(NrGen_LP,1);
    %Also indicate if load is curtailed
    for j = 1:NrGen_LP
        %First production (Generation == 0)
        if LPGen(j)==0
            P_Gen_Node(Gen_LP_Node(j)) = P_Gen_Node(Gen_LP_Node(j)) + P_Gen_LP(j);
        else
            P_LP_Node(Gen_LP_Node(j)) = P_LP_Node(Gen_LP_Node(j)) + P_Gen_LP(j);
            %Check if LP has shed P
            if (-Gen_LP_MinP_pu(j)-(-P_Gen_LP(j))) > 1e-5;
                LPShed(j) = -Gen_LP_MinP_pu(j)-(-P_Gen_LP(j));
            end
        end
    end
    LPShed = LPShed(NrGen+1:NrGen_LP,:);

    %Calculate active power on the lines in p.u
    PL_sol = Pline_mat*angles;
    %Lines that are out of service = 0 MW
    for j = 1:NrLines
        if StatusLine(j) == 0
            PL_sol(j)=0;
        end
    end

    %-----
    %Output results
    %-----

    %Convert from radians to degrees, and return value as output
    angNodes = angles*(180/pi);

    %Production (or load) in each unit (MW), return value as output
    P_Unit = P_Gen_LP*Sbase;

    %Load in nodes (MW), return value as output
    P_LP_Node = P_LP_Node*Sbase;

    %Generation in nodes (MW), return value as output
    P_Gen_Node = P_Gen_Node*Sbase;

```

```
%Return line flows in MW
LineFlows = PL_sol*Sbase;

%Return the active powew curtailed in the nodes
Shed_P = LPShed*Sbase;

else
    angNodes = zeros(NrNodes,1);
    P_Unit = zeros(NrGen_LP,1);
    P_LP_Node = zeros(NrNodes,1);
    P_Gen_Node = zeros(NrNodes,1);
    LineFlows = zeros(NrLines,1);
    Shed_P = zeros(NrLP,1);
end
```

References

- [1] *IEEE Std 493-2007*.
- [2] R. Billinton and R.N. Allan. *Reliability Evaluation of Power Systems*. Plenum publishing corporation, 2 edition, 1994. ISBN 0-306-45259-6.
- [3] R. Billinton et al. A reliability test system for educational purposes- basic data. *IEEE Transactions on Power Systems*, PWRS-3(4):1238–1244, Aug 1989.
- [4] BCP Busarello + Cott + Partner AG. NEPLAN, power system analysis. www.neplan.ch, 2008.
- [5] M. Amelin. *On Monte Carlo Simulation and Analysis of Electricity Markets*. PhD thesis, Department of Electrical Engineering, KTH, Stockholm, Sweden, September 2004. ISBN 91-7283-850-7.
- [6] R. Billinton, S. Kumar, N. Chowdhury, K. Chu, L. Goel, E. Khan, P. Kos, G. Nourbakhsh, and J. Oteng-Adjei. A reliability test system for educational purposes-basic results. *IEEE Transactions on Power Systems*, 5(1):319–325, February 1990.
- [7] R. N. Allan, R. Billinton, I. Sjarief, L. Goel, and K. S. So. A reliability test system for educational purposes-basic distribution system data and results. *IEEE Transactions on Power Systems*, 6(2):813–820, May 1991.
- [8] Subcommittee, P. M. IEEE reliability test system. *IEEE Transactions on Power Apparatus and Systems*, 98(6):2047–2054, November 1979.
- [9] Wijarn Wangdee. *Bulk Electric System Reliability Simulation and Application*. Ph.d. thesis, University of Saskatchewan, 2005.
- [10] Ran Mo. Deterministic/probabilistic evaluation in composite system planning. Master’s thesis, University of Saskatchewan, 2003.