NKS-293
ISBN 978-87-7893-369-0

Using Bayesian Belief Network (BBN) Modelling for Rapid Source Term Prediction – Final Report

Michael Knochenhauer, Vidar Hedtjärn Swaling, Francesco Di Dedda
Frida Hansson, Stina Sjökvist and Klas Sunnegård

Lloyd's Register Consulting AB, SWEDEN

October 2013

Abstract

The project presented in this report deals with a number of complex issues related to the development of a tool for rapid source term prediction (RASTEP), based on a plant model represented as a Bayesian belief network (BBN) and a source term module which is used for assigning relevant source terms to BBN end states. Thus, RASTEP uses a BBN to model severe accident progression in a nuclear power plant in combination with pre-calculated source terms (i.e., amount, composition, timing, and release path of released radio-nuclides). The output is a set of possible source terms with associated probabilities. One major issue has been associated with the integration of probabilistic and deterministic analyses are addressed, dealing with the challenge of making the source term determination flexible enough to give reliable and valid output throughout the accident scenario. The potential for connecting RASTEP to a fast running source term prediction code has been explored, as well as alternative ways of improving the deterministic connections of the tool. As part of the investigation, a comparison of two deterministic severe accident analysis codes has been performed. A second important task has been to develop a general method where experts' beliefs can be included in a systematic way when defining the conditional probability tables (CPTs) in the BBN. The proposed method includes expert judgement in a systematic way when defining the CPTs of a BBN. Using this iterative method results in a reliable BBN even though expert judgements, with their associated uncertainties, have been used. It also simplifies verification and validation of the considerable amounts of quantitative data included in a BBN.

Key words

BBN, Bayesian Belief Network, Severe Accidents, Source Terms, Level 2 PSA, CPT, Conditional Probability Tables

NKS-293
ISBN 978-87-7893-369-0
Electronic report, October 2013
NKS Secretariat
P.O. Box 49
DK - 4000 Roskilde, Denmark
Phone +45 4677 4041
www.nks.org
e-mail nks@nks.org

SCANDPOWER
Risk Management

Lloyd's Register

USING BAYESIAN BELIEF NETWORK MODELLING FOR RAPID SOURCE TERM PREDICTION – FINAL REPORT

Report no. 210829-R-001
Date 25 September 2012
Client NKS

Scandpower AB
P.O. Box 1288
SE-172 25 SUNDBYBERG, SWEDEN
www.scandpower.com
www.riskspectrum.com
Tel. +46 (0)8 445 21 00
Fax +46 (0)8 445 21 01

Report no.: 210829-R001		☐ Open distribution ☒ Distribution only after client's acceptance		
Rev. no.:	Prepared by:	Reviewed by:	Approved by:	Date:
U1	Michael Knochenhauer Vidar Hedtjärn Swaling Francesco Di Dedda Frida Hansson Stina Sjökvist Klas Sunnegård	Michael Knochenhauer Vidar Hedtjärn Swaling (cross-reviewing)	Anders Olsson	2013-10-30

Title:

USING BAYESIAN BELIEF NETWORK MODELLING FOR RAPID SOURCE TERM PREDICTION – FINAL REPORT

Client: NKS

Client specification: N/A

Summary:

The project presented in this report deals with a number of complex issues related to the development of a tool for rapid source term prediction (RASTEP), based on a plant model represented as a Bayesian belief network (BBN) and a source term module which is used for assigning relevant source terms to BBN end states. Thus, RASTEP uses a BBN to model severe accident progression in a nuclear power plant in combination with pre-calculated source terms (i.e., amount, composition, timing, and release path of released radio-nuclides). The output is a set of possible source terms with associated probabilities.

One major issue has been associated with the integration of probabilistic and deterministic analyses are addressed, dealing with the challenge of making the source term determination flexible enough to give reliable and valid output throughout the accident scenario. The potential for connecting RASTEP to a fast running source term prediction code has been explored, as well as alternative ways of improving the deterministic connections of the tool. As part of the investigation, a comparison of two deterministic severe accident analysis codes is being performed.

A second important task has been to develop a general method where experts' beliefs can be included in a systematic way when defining the conditional probability tables (CPTs) in the BBN. The proposed method includes expert judgement in a systematic way when defining the CPTs of a BBN. Using this iterative method results in a reliable BBN even though expert judgements, with their associated uncertainties, have been used. It also simplifies verification and validation of the considerable amounts of quantitative data included in a BBN.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

ADAM	Accident Diagnosis Analysis And Management
AI	Artificial Intelligence
AFW	Auxiliary Feedwater
APRI	Accident Phenomena of Risk Importance (Swedish Research Programme)
ASTRID	Assessment of Source Term for emergency response based on Installation Data
BBN	Bayesian Belief Network
BWR	Boiling Water Reactor
CAMS	Computerized Accident Management System
CANDU	CANada Deuterium Uranium
RCPB	Reactor Coolant Pressure Boundary
CPT	Conditional Probability Table
DBN	Dynamic Bayesian Network
DDET	Discrete Dynamic Event Tree
DPSA	Dynamic Probabilistic Safety Assessment
DSA	Deterministic Safety Assessment
ECC	Emergency Core Cooling
EdF	Electricité de France
ENSI	Eidsgenössisches Nuklearsicherheitsinspektorat (Swiss Federal Nuclear Inspectorate)
EOP	Emergency Operating Procedure
ERI	Energy Research Incorporated Energy Resources International Inc.
ET	Event Tree
FAI	Fauske and Associates
FMEA	Failure Mode and Effect Analysis
FT	Fault Tree
GRS	Gesellschaft für Anlagen- und Reaktorsicherheit GmbH
LOCA	Loss Of Coolant Accident
LRC	Lloyd's Register Consulting – Energy AB (Formerly Scandpower AB)
MAAP	Modular Accident Analysis Program
MARS	Modular Accident Response System
MC	Monte Carlo
MCCI	Molten Core - Concrete Interaction
MCS	Minimal Cut Set
MCDET	Monte Carlo Dynamic Event Tree
MELCOR	Methods for Estimation of Leakages and Consequences of Releases
MVSS	Multi-Venturi Scrubber System
NKS	Nordic Nuclear Safety Research
NPP	Nuclear Power Plant
NRC	U.S. Nuclear Regulatory Commission
O3	Reactor no.3 in Oskarshamn
OECD/NEA	OECD Nuclear Energy Agency
PLASMA	Plant Safety Monitoring Assessment System
PSA	Probabilistic Safety Assessment
PWR	Pressurized Water Reactor
RASTEP	Rapid Source Term Prediction
RELAP	Reactor Excursion and Leak Analysis Program
RHR	Residual Heat Removal

RPV	Reactor Pressure Vessel
SA	Severe Accident
SABINE	Source Term Assessment by Belief Network
SAMG	Severe Accident Management Guideline
SKI	Statens kärnkraftsinspektion (Swedish Nuclear Power Inspectorate, since 2008 part of SSM)
SPRINT	System For The Probabilistic Inference of Nuclear Power Plant Transients
SSI	Statens strålskyddsinstitut (Swedish Radiation Protection Institute, since 2008 part of SSM)
SSM	Strålsäkerhetsmyndigheten (Swedish Radiation Safety Authority)
STERPS	Source Term Prediction Based On Plant Status
USNRC	United States Nuclear Regulatory Commission
VVER	(Vodo-Vodyanoi Energetichesky (Water-Water Power Reactor)

List of notations (mainly used in Chapter 6)

X_i	node i in a BBN
x_i^j	state j in node X_i
X_c	a child node
X_p	a parent node
$pa(X_c)$	the set of parent nodes of X_c
a	a combination of states of the parents of a node
m_{abs}	mean of the absolute difference between an original CPT and a CPT generated with an elicitation methods.
X_r	node of interest
p	output probability of interest
y	parameter under study
SV	sensitivity value
y_v	vertex of the hyperbola branch

1. INTRODUCTION

This final report provides a full presentation of the contents and results of both phases of the NKS project.

Development of analytical tools for use in fast online event or accident diagnosis, and in subsequent forecasting of the radiological source term at a nuclear power plants is increasingly desired by organisations involved in off-site emergency planning and response. Availability of such tools would enhance the efficiency in defining accident response options and make possible a more appropriate off-site response. Large uncertainties are inherent in severe accident situations at nuclear power plants. In attempting to model the progression of a severe accident a mixture of probabilistic and deterministic approaches are typically used. Thus probabilistic safety assessment (PSA) models can be used for creating an over-all logical model representing the response of the plant to various challenges, and for identifying event sequences leading to unacceptable radioactive releases. Deterministic analyses are used to determine critical aspects related to the progression of a severe accident, effects from physical phenomena during the accident, the timing and composition of a releases, etc.

The project presented in this report is related to the development of RASTEP, a computerized source term prediction tool aimed at providing a basis for improving off-site emergency management. The acronym RASTEP stands for Rapid Source Term Prediction. RASTEP makes use of Bayesian belief networks (BBN) to model severe accident progression in a nuclear power plant in combination with pre-calculated source terms. The output is a set of possible source terms with associated probabilities. The approach chosen aims at facilitating decision making in a situation with incomplete, unreliable, or partly contradictive information.

Thus, RASTEP consist of two fundamentally different parts, i.e., a BBN model used to predict plant states and release paths, and a source term definition part used to characterise the source term, i.e., amount, composition, timing, and release path of released radio-nuclides. The information contained in the BBN model is based on a range of information sources, including expert judgment and prior information from the plant PSA model. The BBN information is iteratively updated during the course of the severe accident based on available plant observables, e.g., pressure or temperature measurements. The definition of source term and the modelling of severe accident progression use information from deterministic severe accident analysis tools, e.g., MAAP. RASTEP is intended to interface with commonly used off-site dose calculation tools, e.g., LENA and/or ARGOS.

The work performed is partly based on the outcome from a pilot project performed in 2001-2005 within the EU project STERPS, which was part of the EU framework programmes 5 and 6 [1, 2]. There are parallel ongoing activities related to development of plant specific RASTEP models for Swedish nuclear power plants (NPP:s) performed by the Swedish Radiation Safety Authority (SSM).

The NKS project has been carried out in two project phases (Phase 1: July 2011 to June 2012; Phase 2: July 2012 to June 2013) with the objectives to improve knowledge within the following areas:

A. Improving the flexibility of the source term predicting capability of RASTEP. This involves definition and evaluation of a dynamic source term module for use within RASTEP, i.e., evaluation of feasible fast running deterministic codes for online/dynamic calculation and recalculation of source terms.
B. Comparison between severe accident analysis codes. This involves comparison between analysis codes MAAP and MELCOR.

C. Improvement of some complex BBN functionality issues, including methods for expert judgment, sensitivity analyses and improving the capability of quantifying the conditional probability tables (CPT:s) in an efficient and quality assured way..
D. Comparison of RASTEP with other internationally available fast-running software codes aimed at predicting the source term after a severe accident (participation in initiation of OECD/NEA task).

2. BACKGROUND AND OVER-ALL CONTEXT

2.1 Relation to previous work

RASTEP takes as its starting point the outcome of the EU project STERPS (Source Term Indicator Based on Plant Status), a major international project performed in 2001-2005 [1, 2]. The STERPS project was part of the European Union 5th and 6th Euratom Framework program, and had the objective to explore the possibility to use a computer based tool for rapid and early diagnosis of plant status and subsequent estimation of the possible environmental releases, based on a probabilistic plant model using Bayesian Belief Network (BBN) methodology.

The Swedish contribution to the project aimed at the development of a first prototype version of a BBN model for the Swedish boiling water reactor Oskarshamn 3 (O3), an ASEA-Atom reactor of the type BWR 75. Participation in the EU project was through the Royal Institute of Technology in Stockholm (KTH), with Wiktor Frid (professor at KTH at the time; now at SSM) as project manager, and with the participation of Michael Knochenhauer from Lloyd's Register Consulting (LRC; formerly Scandpower) and with extensive in-kind participation from OKG, owner of the O3 plant.

The project used the generic BBN software Netica (developed by Norsys Inc.), and the prototype user interface SPRINT (System for the Probabilistic Inference of Nuclear Power Plant Transients), which was developed within the STERPS project for handling of the BBN. The user interface includes a set of questions and background information, which are used in order to gain information about crucial plant parameters (“observables”) during the course of a severe accident. SPRINT also includes graphical presentation of analysis results, both in terms of node probabilities and of characteristics for radioactive releases (amount, composition, and timing). The EU project demonstrated the feasibility of using BBN technique for modelling of severe accidents, but also identified a number of issues and challenges related to such an approach.

Within the project, a tentative BBN models were developed and tested for a number of different reactor types. Oskarshamn 3 (O3) was the only boiling water reactor (BWR) in the project, and a rather detailed outline of a BBN model was developed for the unit, including major parts of the basic BBN structure but lacking a number of functions and based on a very simplified quantification of the conditional probability tables, and using a very coarse set of pre-defined source terms.

Starting 2008, SSM has conducted a project which aims at the development of plant specific BBN models for all Swedish nuclear power plants (NPPs). This has been done using the basic approach defined by the STERPS project, i.e. with a plant model consisting of two different parts; a BBN model used to predict plant states and release paths and a source term definition part used to characterise the source term (height, composition, amount and timing). This development is to include the development and documentation of an analysis methodology, including the necessary QA (Quality Assurance) procedures and procedures for validation and verification of developed BBN models, as well as the definition of procedures for update and maintenance of the plant specific models in RASTEP. Initially a basic BBN model (with associated source term definitions) was developed and largely validated for O3. Models are currently being developed for other Swedish plants (generic pressurised water reactor and for the BWR Oskarshamn 2), and development of further plant models is planned. The O3 model developed as part of the SSM project has been used as the reference model also for this NKS project.

The basic aim of NKS project RASTEP is to address a number of advanced topics that constitute R&D challenges in the application of BBN to source term predictions during an NPP severe accident. The project has dealt mainly with the following issues:

- Definition of the source terms (ways to improve precision and functionality of the source term module of RASTEP; supported through two M.Sc. theses)
- Comparison of codes for accident sequence and source term calculation (comparison between analysis codes MAAP and MELCOR; supported by one M.Sc. thesis).
- Challenges in BBN structure and quantification (methods for dealing with sensitivity with respect to parameters and model structure and development of a systematic approach for defining complex CPTs in a BBN; supported through a double M.Sc. thesis)
- Comparison of RASTEP with other internationally available fast-running software codes aimed at predicting the source term after a severe accident (participation in initiation of OECD/NEA task).

2.2 Structure of a RASTEP model

As stated initially, RASTEP consists of two fundamentally different parts, i.e., a BBN model used to model accident progression, predict plant states and release paths, and a source term definition part used to characterise the source term (release height, composition, amount, and timing). These two parts are described in the sections below.

2.2.1 Part 1 – BBN model

Figure 2.1 gives an overview of the general lay-out of a BBN, including the linking to PSA information (prior information) and to relevant observables (plant status parameters, such as pressures, temperatures, water levels etc.).

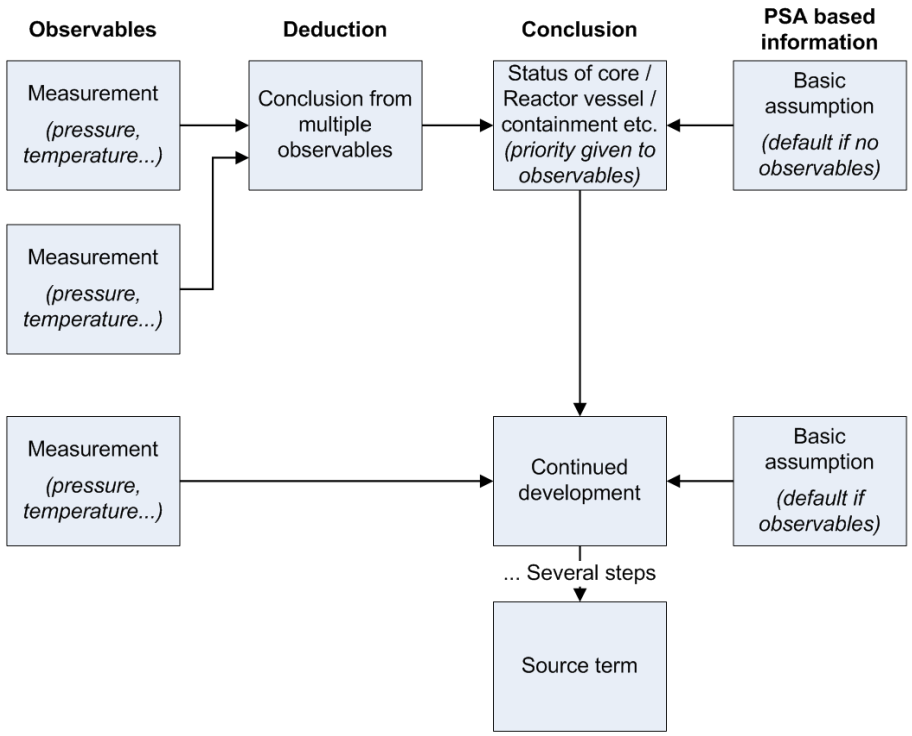


Figure 2.1 RASTEP – Basic structure of the BBN model

Figure 2.2 gives a simplified overview of the Oskarshamn 3 BBN, showing the main blocks of the network and the most important interrelations within the block. In the complete network, each of the sub-networks includes a number of nodes. The total number of nodes in the network is about 90, making this a rather complex BBN.

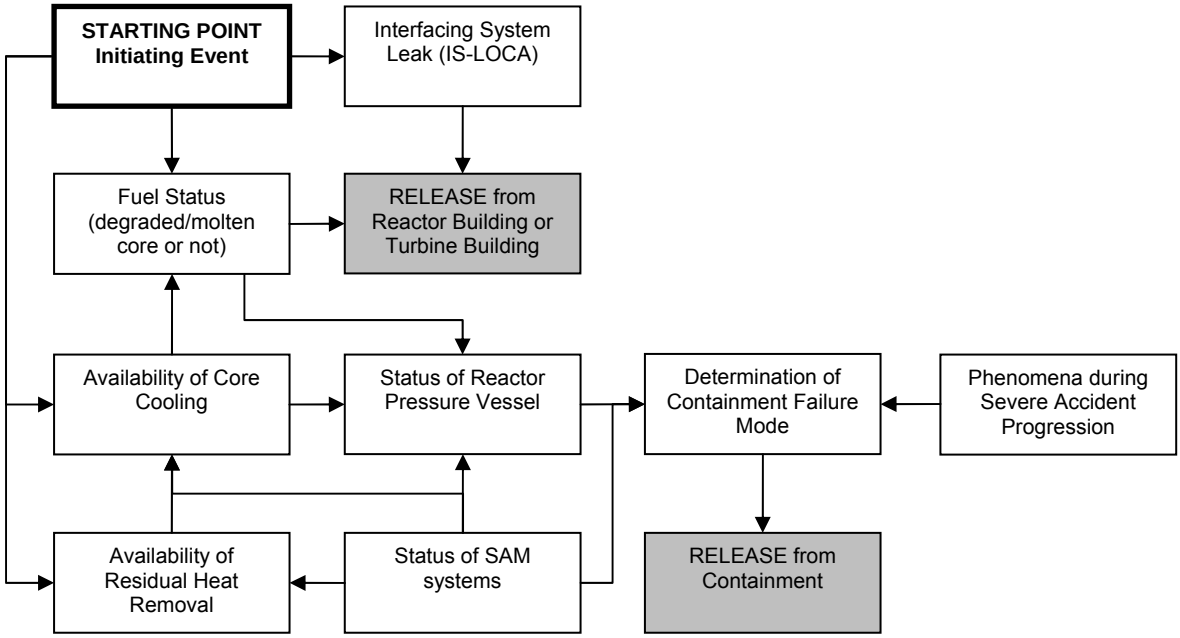


Figure 2.2 Basic BBN structure for ASEA-Atom BWR:s

Sub-networks are as far as possible structured in an order that reflects the accident progression. The following main blocks have been defined:

1. Initiating event
2. Core cooling
3. Residual heat removal
4. Fuel status
5. Reactor pressure vessel status
6. Containment status ➔ Containment source terms
7. Reactor building status ➔ RB source terms
8. Turbine building status ➔ TB source terms

As indicated in the figure, the starting point of the network is the identification of the initiating event of the accident sequence. Thereafter, the probability of a number of different release paths is estimated based on the status of a number of fundamental blocks (fuel status, status of important safety systems, containment status, etc.), as well as of the success or failure of severe accident mitigation systems and of severe accident management actions.

In the use of the SPRINT software, the user is prompted to answer questions about the accident scenario; these answers are entered as findings into the corresponding node in the BBN after which inference is performed. This changes the joint probability distribution of the network and hence the source term probabilities.

The questions are used to provide the network with information on the boundary conditions of the plant. This includes knowledge of parameters such as pressure, temperatures, water

levels, radiation and system statuses etc. A set of alternative answers is presented to each question.

2.2.2 Part 2 – Source term module

The output from RASTEP is a set of possible plant states, ranked in order of probability. This is the outcome of the (shaded) source term nodes illustrated on a high level in Figure 2.2. Each plant state has an associated source term. The source terms are derived from a set of pre-calculated plant specific source terms (largely from the PSA level 2 for the analysed NPP), which have been mapped to each final plant state when creating the RASTEP model. In SPRINT this information is stored in an Excel spreadsheet. The source terms are mainly characterised by:

- Released amount (Becquerel per radionuclide group, e.g., Xe-133, I-131, Te-132, Mo-99, Cs-137, Rb-88)
- Chemical composition (radionuclides included)
- Iodine specification (fractions of elemental, organic and aerosol iodine)
- Release height and thermal energy
- Division of the total time into time spans which correspond to the occurrence of some major changes in the characteristics of the release (as modelled in MAAP)

Figure 2.3 shows the source term interface of SPRINT. The upper part of the window shows the range of possible source terms identified by the BBN and the current set of observables. The lower part illustrates the release amount, composition and timeframe for one of the possible source terms.

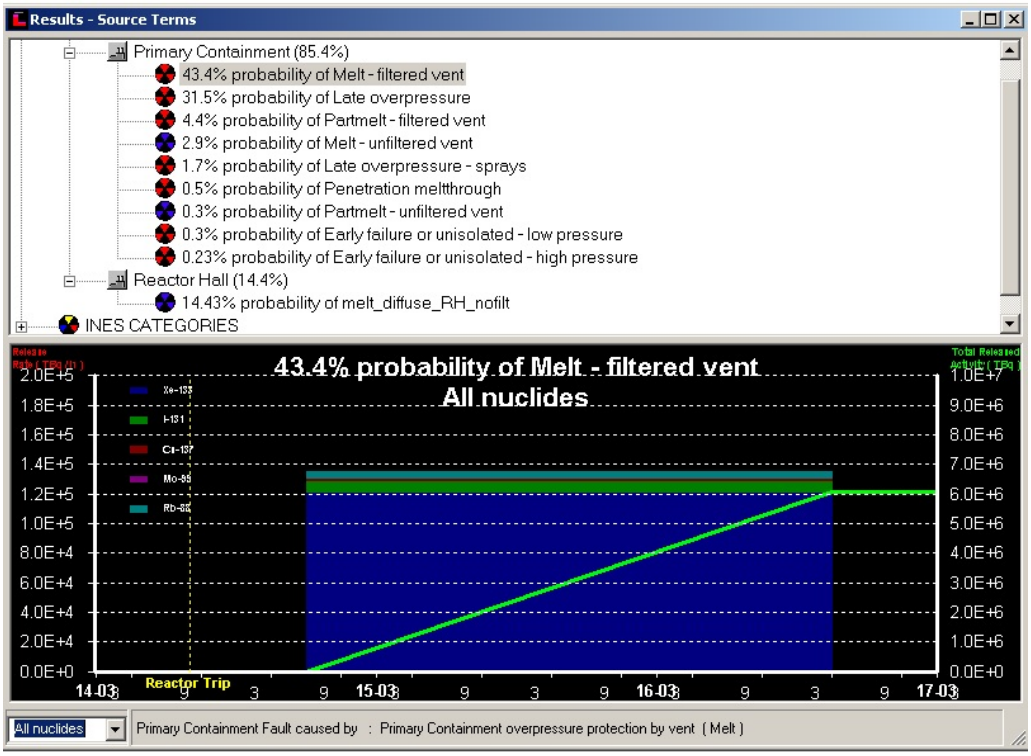


Figure 2.3. Presentation of source terms and their associated probabilities

3. INTRODUCTION TO BAYESIAN BELIEF NETWORKS

3.1 Nodes and conditional probability tables

Bayesian belief networks (BBN) are established in a wide variety of domains, including medicine, ecology and engineering, to model cause and effect [3] [4]. This is because of their ability to capture the probabilistic relationship between events and to combine different sources of data. Another advantage of a BBN is that the result is often convincing and conclusive even if the data is uncertain, which is very common in many applications. [5] [6]

A BBN gives a graphical representation of events that occur in reality and is built as a directed acyclic graph, i.e. a directed graph without cycles. Each node in a BBN represents an event described by a number of possible states, which can be either continuous or discrete. A causal relationship between two nodes is represented by a directed arc, leading from the parent node (cause node) to the child node (effect node), see Figure 3.1. The absence of an arc between two nodes represents conditional independence assumptions.[7]

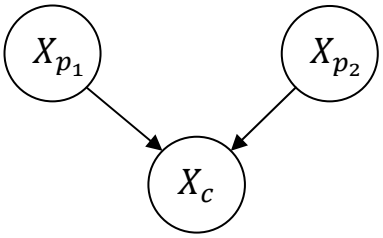


Figure 3.1 Simple Bayesian Network where node X_c has the two parent nodes X_{p_1} and X_{p_2}.

An arc between a parent node X_p and a child node X_c, see Figure 3.1, can be interpreted in different ways and one of the following statements usually holds:

- X_p causes X_c or
- X_p partially causes or predisposes X_c or
- X_c is an imperfect version of X_p or
- X_p and X_c are functionally related or
- X_p and X_c are statistically correlated. [8]

A BBN consists of a qualitative part and a quantitative part; the relationship described by the nodes and arcs being the qualitative part [9]. The strength of influence between a child node and its parent nodes is quantified by conditional probabilities which are represented in a CPT, see Table 3.1. Each probability in a CPT represents the probability of a child node being in a certain state given a set of parent states, e.g. according to Table 3.1 $P\left(x_c^1 \middle| x_{p_1}^1, x_{p_2}^1\right) = 0.1$. For nodes that have no parents the table consists of the unconditional probabilities of each state of the node. The probabilities in the CPTs are the quantitative part of the network and they are based upon different types of information, ranging from well-founded theory, over frequencies of events in a data base, to experts' beliefs. [10].

Table 3.1 A CPT for a child node X_c with the two states x_c^1 and x_c^2. The node has two parents, X_{p_1} and X_{p_2}, with two states each, $x_{p_1}^1$ and $x_{p_1}^2$ respectively $x_{p_2}^1$ and $x_{p_2}^2$.

		Child node X_c	
Parent node X_{p_1}	Parent node X_{p_2}	x_c^1	x_c^2
$x_{p_1}^1$	$x_{p_2}^1$	0.1	0.9
$x_{p_1}^1$	$x_{p_2}^2$	0.3	0.7
$x_{p_1}^2$	$x_{p_2}^1$	0.8	0.2
$x_{p_1}^2$	$x_{p_2}^2$	0.7	0.3

The following is a more formal definition of a BBN.

Definition 2.1. A *Bayesian belief network* is a set $B = (G, P)$ where
- G is a directed acyclic graph with nodes $X(G) = \{X_1, \dots, X_n\}$, $n \geq 1$, and arcs $A(G)$;
- P is a set of conditional probabilities $P(X_i \mid pa(X_i))$, for all $X_i \in X(G)$, where $pa(X_i)$ is the set of immediate predecessors of node X_i. [11]

One valuable property of a BBN is that the product of the conditional probabilities can be used to define the joint probability distribution for all the nodes in a BBN. The joint probability of the nodes $X_1, \dots, X_n$ in a BBN is

$$P(X_1, \dots, X_n) = \prod_{i=1}^{n} P\big(X_i \big| pa(X_i)\big).$$

Using this product, the conditional probabilities and the structure of the BBN the marginal probability of each node being in one of its states can be determined by marginalisation [7].

3.2 Determination of node dependency using the d-separation criterion

The d-separation criterion provides a way to determine whether any pair of nodes in a BBN is dependent given a set of observations. Before defining d-separation the concept of blocking has to be introduced.

Definition 2.2. Let $G = (X(G), A(G))$ be an acyclic directed graph with nodes $X(G)$ and arcs $A(G)$, and let s be a chain, i.e. a path in the undirected underlying graph of G, in G between nodes X_i and X_j. We say that s is *blocked* by the set of nodes $O \subseteq X(G)$, if either X_i or X_j is included in O, or s contains three consecutive nodes X_{k-1}, X_k, X_{k+1} for which one of the following conditions holds:
(1) arcs $X_{k-1} \leftarrow X_k$ and $X_k \rightarrow X_{k+1}$ are on the chain s, and $X_k \in O$;
(2) arcs $X_{k-1} \rightarrow X_k$ and $X_k \rightarrow X_{k+1}$ are on the chain s, and $X_k \in O$;
(3) arcs $X_{k-1} \rightarrow X_k$ and $X_k \leftarrow X_{k+1}$ are on the chain s, and $\sigma(X_k) \cap O = \emptyset$, where $\sigma(X_k)$ is the set composed of X_k and its descendants.

Letting O be the set of observed nodes the definition of d-separation between two nodes X_i and X_j is stated as follows.

Definition 2.3. Let G be an acyclic graph and let X_i, O, $X_j \subseteq X(G)$. The set of nodes O is said to *d-separate* the nodes X_i and X_j in G, denoted $\langle X_i \mid O \mid X_j \rangle_G^d$, if every chain from X_i to X_j in G is blocked by O. [11]

If two nodes are d-separated they are called structurally independent which means

$$P(X_i, X_j|O) = P(X_i|O)P(X_j|O).$$

The three types of blocking causing a d-separation between two nodes X_i and X_j are illustrated in the Figures below.

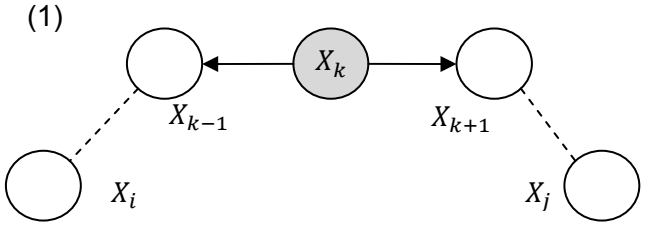


Figure 3.2 An illustration of d-separation. The connection between X_i and X_j is diverging and blocked by the observed node X_k, grey represents an observation.

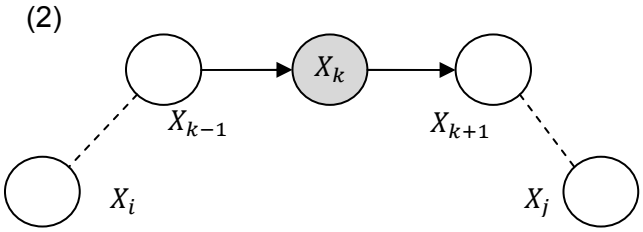


Figure 3.3 An illustration of d-separation. The connection between X_i and X_j is serial and blocked by the observed node X_k, grey representing an observation.

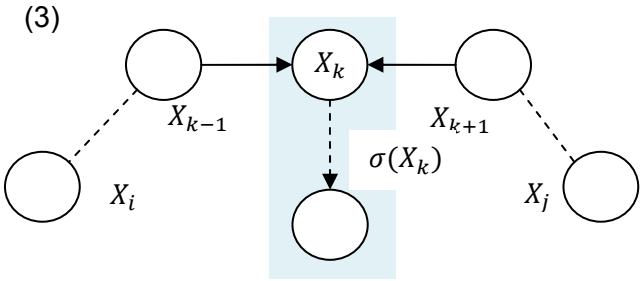


Figure 3.4 An illustration of d-separation. The connection between X_i and X_j is converging and blocked since neither X_k or its descendants are observed (shaded in blue).

3.3 Example of a BBN

This section presents a simple example of a BBN and illustrates the way in which a BBN can be used. The network presented in Figure 3.5 represents a medical diagnosis example where the two top nodes are "predispositions" which influence the likelihood of the illnesses following below. In the bottom of the network are the disease symptoms.

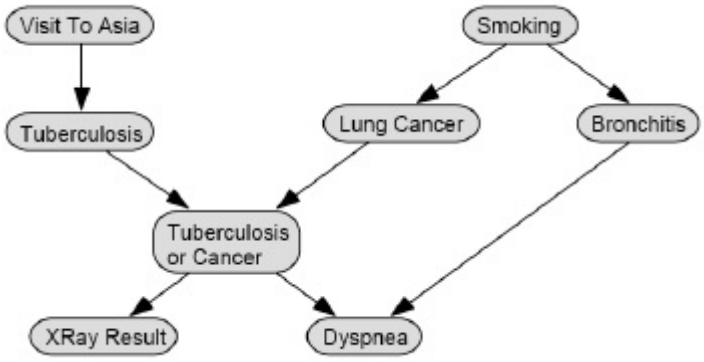


Figure 3.5. Network representing medical diagnosis example

Each of the nodes in the network has different states e.g. smoker/non-smoker for the node "Smoking" and every node state has a default probability. This information is summarized in a so called conditional probability table for the node. The conditional probability tables increases in size as the number of parent nodes and states increases. Default probabilities can be based either on statistics (for the "Visit to Asia" and "Smoking" nodes) or be based on the status of the parent nodes (every other node). For instance, the probability of "Lung Cancer" depends on if the patient is a smoker or not. This information is represented in a conditional probability table for the node lunch cancer. Table 3.2 shows such a conditional probability table.

Table 3.2. Conditional probability table

Parent node(s)	**Child node: Lung cancer**	
Smoking	Lung cancer	No Lung cancer
Smoker	10%	90%
Non-smoker	1%	99%

After belief update has been performed, the network will display the state probabilities for every node (as percentages). The starting point in this example is before any observations have been made (generic case). The "Visit to Asia" and "Smoking" nodes include prior beliefs based on which the state probabilities in the other nodes are calculated. Figure 3.6 shows the generic case of the network.

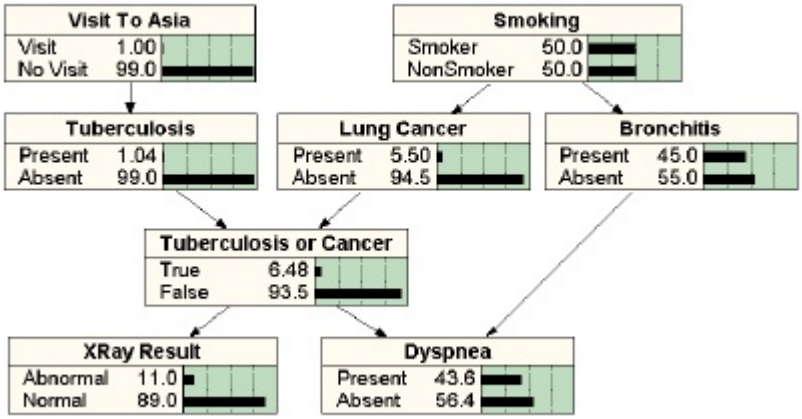


Figure 3.6. Generic case of the network

Now the network is applied to a specific case in which observations are made and entered as findings (input) in the observable nodes. The observations consist of questions that are asked to the patient as well as of medical examinations. In our example the diagnosis involves a smoker who has not visited Asia and whose medical examination have shown normal X-ray results but suggests presence of dyspnoea. Based on this information, after belief updating, the network shows that a high probability of bronchitis prevails.

Figure 3.7 shows the network for this case. As seen from the figure, the combined information from the observables indicates the presence of bronchitis, with a next to negligible probability of either tuberculosis or lung cancer.

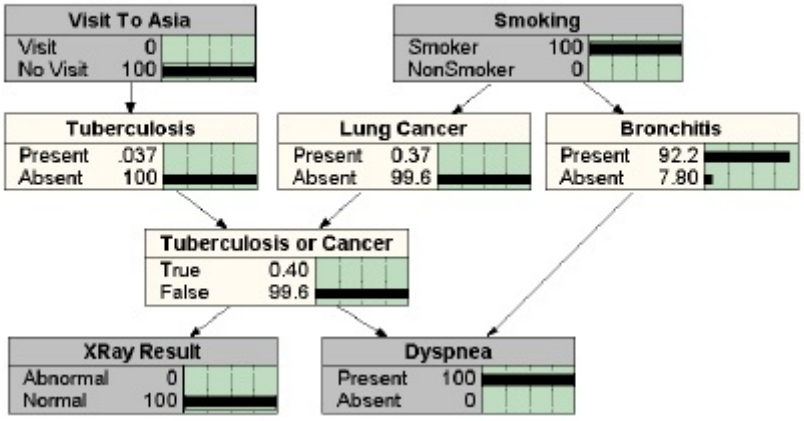


Figure 3.7. Specific case of the network

4. LINKING RASTEP WITH A FAST RUNNING DETERMINISTIC CODE FOR SOURCE TERM DETERMINATION

4.1 Introduction

This chapter aims at investigating potential approaches for improvement of the source term predictions. During project phase 1 one approach for enhancing the source term was identified [12]. The main objective here is to follow that approach and explore the integration of a fast running deterministic source term module within RASTEP, making the tool more dynamic and the source term predictions more realistic.

The most feasible alternatives for the source term module will be analysed in this report and conceptual and technical issues will be discussed. The purpose is to address the challenge of combining deterministic safety analysis (DSA) and PSA with particular focus on the use of these approaches in Severe Accident Management as well as to address the issue of implementing a fast running deterministic code with RASTEP.

4.2 Problem statement

Probabilistic methods currently used for safety analyses are mostly based on static logic models. On the contrary, deterministic models are capable of resolving time dependent interactions between physical phenomena. Integrated Deterministic-Probabilistic Safety Assessment (IDPSA) is one of the proposed acronyms used to group new approaches on combined DSA and PSA. IDPSA methods aim at making safety studies more flexible and realistic with the goal of meeting increasingly stringent safety requirements, licensing rules and to open the way for new reactor designs [13].

The BBN model of RASTEP is based on logical relationships between various key plant parameters, represented by a graphical network. The network is plant specific and is derived from the system description and plant documentation. The PSA functional structure is also included in the network as well as thermal-hydraulic considerations when it comes to specific systems. The outcome of RASTEP is typically defined by various plant end-states ranked according to probability. Each plant state is then associated to environmental source terms which give the quantity, characteristic and timing of the radioactive release. Source terms are currently determined by MAAP calculations during PSA level 2 studies.

The analysis performed with RASTEP aims at addressing the actual accidental scenarios. This differs from the way the analysis is performed during PSA studies, where a set of representative scenarios are investigated. Taking this distinctive difference into consideration, the possibility of updating the source term information as the accidental scenario progresses would give the opportunity to have more realistic results for the environmental release in the very current situation.

The objective of this part of the project is to evaluate how the source term predictions can be updated, and understand what the requirements are to implement this modification. A deeper understanding of the concepts behind the integration of deterministic and probabilistic methods is the starting point to achieve this goal.

4.3 Tools for source term assessment

The tools that model severe accidents progression can be divided into three main types according to their use and capabilities, i.e., mechanistic codes, PSA codes, and parametric codes.

Mechanistic codes model the physical phenomena during a severe accident in a very detailed manner. Typically these codes need a comparatively long computational time (i.e., some days), and are used in severe accident research providing valuable data for use within benchmark activities. Most of these codes are based on finite element for structural dynamics modelling and are able to handle non-linearities. There are also the simple parametric codes which are intended for very specific PSA applications. An example of application for this type of codes is the source term estimation. These codes are based on simple parametric models where interpolations between fixed points are made to determine the values of parameters. The fixed points used for the interpolations are pre-calculated with a more complex code. Parametric codes generally need a lot of effort in initial testing and calibration by the use of more detailed calculations or experimental data.

Multiple codes, each dealing with a specific aspect of the accident behaviour, are sometimes coupled. A more modular and integrated approach has been adopted in later generations of severe accident codes, e.g., MAAP and MELCOR. A short description of these codes is given below.

4.3.1 MAAP – Modular Accident Analysis Program

MAAP is an integral computer code owned and licensed by EPRI and developed by Fauske and Associates (FAI). It simulates accident progression in light water and heavy water nuclear power plants. Several parallel versions of this code are specifically made for other types of reactors (e.g., CANDU and VVER reactors). MAAP simulates Loss-Of-Coolant Accident (LOCA) and non-LOCA transients mainly for PSA applications. It can also simulate severe accident sequences, including actions taken as part of the Severe Accident Management Guidelines (SAMGs). When a set of initiating events and operator actions are given, MAAP predicts the plant's response as the accident progresses. MAAP, and specifically version 4 of the code, is used for the following:

- Prediction of the timing of key events (e.g., core damage, core uncover, core relocation to the lower plenum, and vessel failure);
- Evaluation of the influence of mitigation systems, including the timing and the impact of operation of such systems;
- Evaluation of the impact of operator actions;
- Prediction of the magnitude and timing of fission product releases (source terms);
- Investigation of uncertainties in severe accident phenomena.

MAAP requires two files as input: a parameter file which contains plant-specific information, specifications of the output and user-controlled phenomenological parameters; and a sequence input file which specifies the accident initiators, operator actions, and sequence control times. After the information in the two files has been processed, the code is able to predict the sequence of events and the corresponding plant conditions. The output also includes a summary of the sequences and events, tables of time-dependent results, and tabulated results. These results provide all the details of the plant's status for the selected times and are suitable for plotting.

MAAP results are primarily used for the determination of level 1 and 2 PSA success criteria and for accident timing for use in for human reliability analyses. These results are also used for equipment qualification applications, determination of fission product release frequencies,

emergency planning and training, simulator verification, analyses to support plant modifications, and generic plant issue assessments. MAAP, as integrated code, treats the full spectrum of important phenomena that can occur during an accident and, at the same time, creates the model for thermal-hydraulics and fission products assessment. It is also capable of modelling the primary system and the containment and reactor/auxiliary building [15].

4.3.2 MELCOR – Methods for Estimation of Leakages and Consequences of Releases

MELCOR is a fully integrated computer code that models the progression of severe accidents in LWR nuclear power plants. MELCOR is developed at Sandia National Laboratories under a contract with the U.S. Nuclear Regulatory Commission (NRC). A large series of severe accident phenomena is treated in MELCOR in a unified framework (e.g., thermal-hydraulic response in the RCS, cavity, containment, and buildings; core heat up, degradation, and relocation; etc.). The current uses of MELCOR also include estimation of severe accident source terms with sensitivity and uncertainty analysis [16]. MELCOR is capable of modelling all phases of severe accident progression. Thermodynamic state properties are treated in the spatial geometry and volumes of an NPP. Two-phase models and non-equilibrium thermodynamics are also implemented (e.g., different temperatures and phase conditions) as well as the opportunity to consider volumes occupied by other material.

The code comprises of a driver module and a number of model packages which are executed depending on the problem that has to be solved. The use of MELCOR in combination with the model packages offers the same types of assessments that is included in MAAP (see list in section 4.3.1). Conversely to MAAP, MELCOR produces a number of output files which include a diagnostic file displaying errors and warnings to assist the user in debugging the input, an output text file, a plot file and a message file. A different package computes off-site consequences of radioactive releases based on source terms previously calculated [17].

4.4 On-line accident diagnostic tools

In this section, two on-line accident diagnostic tools, MARS and ADAM, are presented. Both these codes were indicated in a previous M.Sc. thesis work [12] as good candidates for use within RASTEP. Especially MARS (Modular Accident Response System) was seen as a promising candidate software in this respect. Accordingly, particular focus will be given to the MARS software and its features. A number of software tools other than MARS or ADAM are already available (e.g., ASTRID, SABINE, SESAME, CRISALIDE, CAMS, etc.). However these tools do not look feasible for implementation with the Bayesian Network and the calculation speeds are not compatible with the intended RASTEP functionality.

4.4.1 MARS – Modular Accident Response System

4.4.1.1 Introduction and main features

MARS is an integrated software suite that provides a complete engineering simulation of an operating nuclear power plant based on actual conditions within the plant. MARS models the plant response during all modes of operation such as shutdown, refuelling, normal and abnormal, and accident conditions. The MARS suite includes:

- A simulation of the nuclear power plant using MAAP and other SA codes;
- A user-friendly Graphical User Interface (GUI) that represents the nuclear power plant;
- The ability to use on-line plant data for engineering simulations;
- Modules for performing alternative and/or redundant instrumentation readings;

– Models for emergency response activities, e.g., on-site and off-site radiation dose assessment.

The MARS software suite uses MAAP along with customized and industry standard software to support the MARS applications. As described in section 4.3.1, MAAP is able to simulate the thermal-hydraulic and fission product plant response of the nuclear power plant during normal, accident and severe accident sequences, including actions taken as part of normal operating procedures and of emergency operating procedures (EOP).

The MARS GUI provides the user with an easy-to-use method for interacting with the simulation providing the user with information that is easy to understand and as familiar as possible. By using dynamic sets of on-line data, MARS is able to obtain insights into the current and potential future status of the plant during many modes of operation. The MARS' *tracker* function uses a limited set of dynamic plant signals to initialize the MAAP code and then follow the evolving plant response. The MARS *instrumentation* module provides an alternative means to validate and/or obtain plant instrumentation readings. The MARS *predictor* module provides the user with the ability to perform much faster-than-real-time evaluations, including modelling operator actions. For example, the MARS user can model the loss of a given safety system to determine its effect on the overall accident progression. The *predictors* provide ways to look into future plant states.

Figure 4.1 shows a screenshot of the graphical user interface of the version of MARS used by CSN in Spain (MARS-CSN) [19].

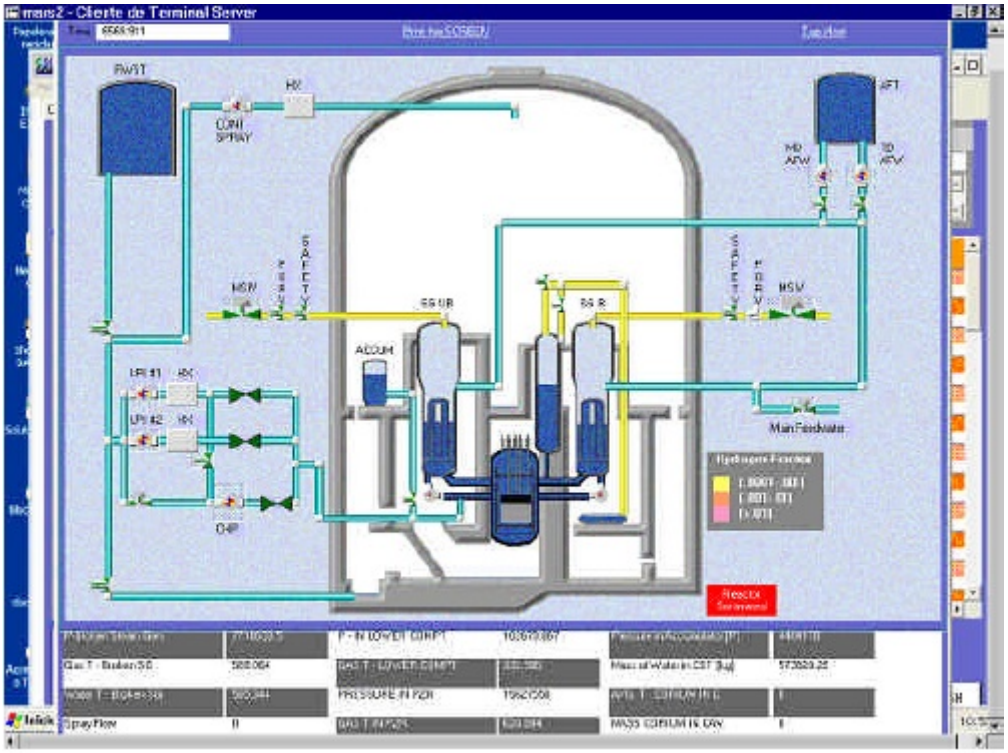


Figure 4.1 Example of MARS Graphical User Interface

4.4.1.2 **Input to MARS**

MARS normally uses around 75 signals as input data, on-line transmitted from the plant computer or entered manually. This generates the information for initiating MAAP accident

simulations [18]. The incoming plant data will go through two steps of processing, i.e., verification and conversion. As shown in Figure 4.2, the MARS software processes the data either when they come directly from the control room computer of an NPP, or when they are simulated with other tools. In the verification process, signals of good quality are accepted and those that are determined to be of poor quality are discarded. In this case, the user will be notified, and another method for determining a representative value will be used. If some parameter of the primary system needed by MARS is not included in the input, the remaining parameters will be used to estimate a representative value. The user also has the possibility to manually input a value based on information that is available off-line or derived from some other sources, e.g., a BBN in the case of RASTEP.

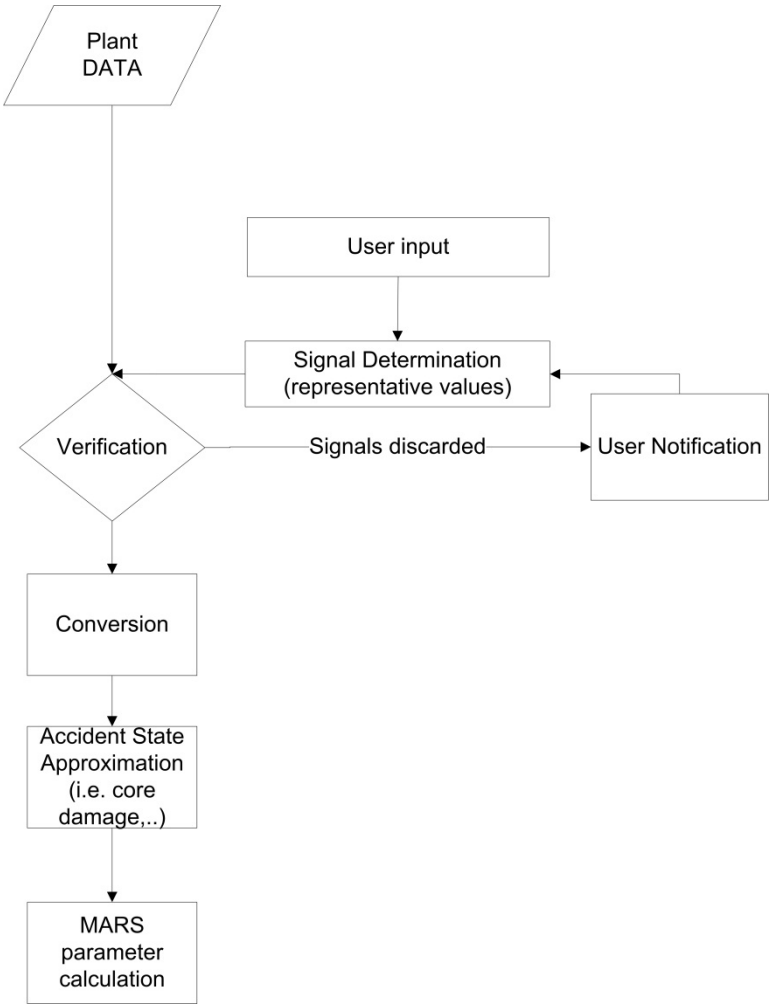


Figure 4.2 Overview of the MARS processing structure

In the MARS initialization routine, the processed plant data are used to estimate the approximate accident state (e.g., core uncovered, core damaged, reactor pressure vessel failure, etc.). The identification of the time intervals for such events is very important. This is a fundamental feature that makes it possible to follow the accident progression in its initial phases. When the accident state has been approximated, several thousand parameters that are required by MARS to be fully initiated are computed based on the plant data.

4.4.1.3 The tracking module

The MARS *tracking* function is a code module used to check that the simulation follows the actual plant behaviour. The *tracking* function is the basis for the *predictor* function to be actuated. Initially, the *tracking* function performs an assessment of the first symptoms of the accident to determine the plant status and the types of accident initiators. Once the plant status and the accident initiators are identified, and the evolving set of plant data is available, the *tracker* performs the calculations to guide the MAAP code towards the simulation of the plant behaviour.

A comparison of the plant data available with the *tracker* information is performed over multiple time intervals in order to assess how the simulation follows the actual plant status. If differences between the simulation and the actual plant status are found, and certain parameters are not in the range specified by the user, the *tracker* corrector logic is applied, i.e., the *tracker* simulation is modified considering the differences between the simulated values and the actual plant data. The *tracking* module is also capable of evaluating the potential root causes of the accident. For instance, if the *tracker* identifies a LOCA, the information generated by the *tracker* about the break size (of the pipe) and the elevation may be used to determine possible locations of the break.

4.4.1.4

4.4.1.5 The predictor module

The *predictor* module provides indications and predictions of future plant states when only the actual scenario is available. For instance, the *predictors* can be used to estimate the time before a major change in the accident scenario occurs (e.g., core uncovery, core damage, reactor pressure vessel failure or containment failure) and the efficiency of accident management strategies.

When the *tracker* has gathered enough information on the plant status, the *predictors* are initialized. The calculations of the *predictors* can be performed assuming a wide variety of actions by the operators of the plant. Usually, in the first execution, the assumption that the operators follow entirely the EOP is made. The second *predictor* analyses the behaviour of the plant in both the short and long term assuming no additional actions by the operators. Other *predictors* analyse the effects when following the accident management guidelines (SAMGs) or other unexpected operator actions.

The *predictors* provide information such as minimum injection flow rates for successful mitigation of the accident, timing of vessel and containment failures, effects of the operator actions, source term assessment, and future accessibility to all plant areas based in predicted radiation levels.

4.4.1.6 Instrumentation

The availability of signals and parameters coming from the plant instrumentation is crucial to manage an accident scenario. One important requirement is to obtain, at least, approximate values for the relevant parameters in the plant. It is also fundamental to discard instrumentation values which appear to be of poor quality. The capability to analyse the available plant signals and to determine the validity of the simulated data is a key feature of MARS. Confidence levels are also provided for some variables. In some situations, the user is notified of poor quality data and countermeasures will be taken by MARS to replace those data with representative values.

4.4.2 ADAM – Accident Diagnosis Analysis and Management

The Accident Diagnostic, Analysis and Management (ADAM) computer code is a tool for online accident diagnostics, management and training developed by ERI (Energy Resources International, Inc.) and financed mainly by the Swiss Federal Nuclear Safety Inspectorate

(ENSI). The capabilities of ADAM incorporate a balance of mechanistic and phenomenological models with a simple parametric approach. All the typical analyses and characteristics typical of severe accident codes (like MAAP and MELCOR) are also included in the ADAM model.

The model in ADAM is defined by a coarse spatial nodalisation of the reactor coolant system and containment, where the definition of timing is explicit. This model configuration enables ADAM to give results 100 to 1000 times faster than real-time. Figure 4.3 shows a screenshot of the graphical user interface of the ADAM code.

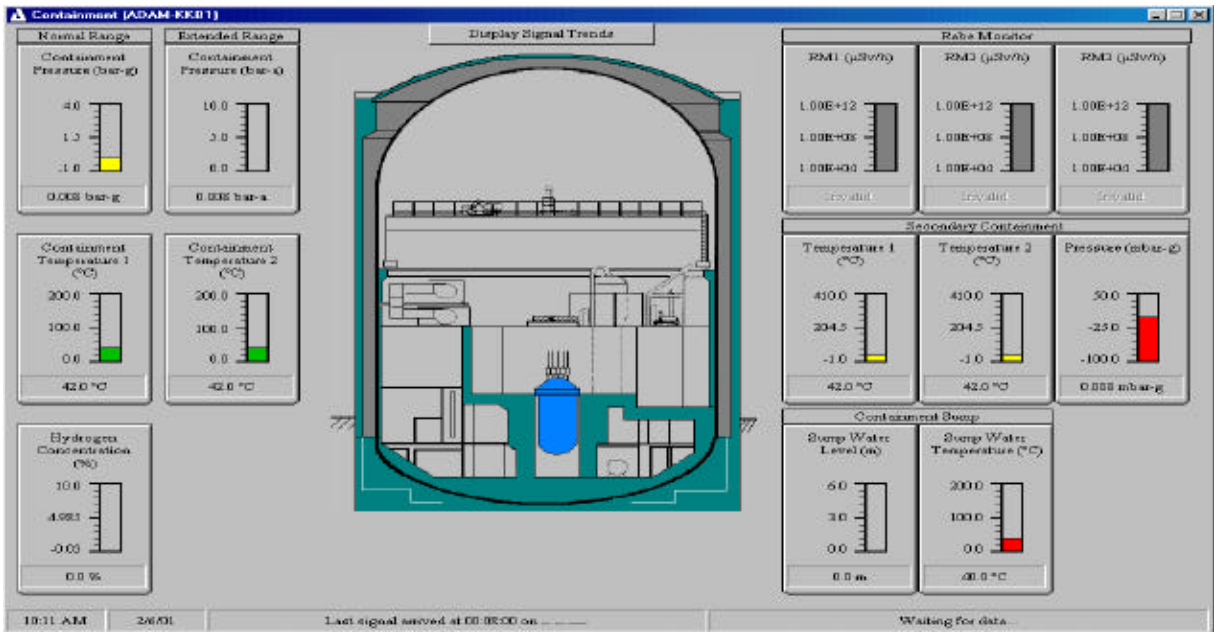


Figure 4.3 Example of ADAM Graphical User Interface

ADAM is designed to operate in two modes referring to two different modules the On-Line Diagnostics mode, and the Accident Management and Analysis mode.

In the On-Line Diagnostics mode (D), the parameters available from on-line measurements at the plant are used as input for ADAM. These values are fed into ADAM at specified time intervals (e.g., every 2 minutes) and used to assess safety margins. A number of alarms denote these safety margins in the graphical interface (e.g., margin to containment failure, margin to venting system actuation, etc.). Furthermore, the state of the reactor, of the containment, and of all auxiliary buildings is monitored and a deterministic logic is the basis for the calculation of the expected plant states.

In the Accident Management and Analysis mode (A), ADAM is used to simulate multiple accident scenarios and to determine the potential consequences and implications of a series of Severe Accident Management (SAM) actions. Consequently, ADAM offers efficient means for training, drills, accident analysis and emergency planning. The source term assessment is also included in this mode of operation as well as the evaluation of PSA success criteria.

The ADAM system is designed to provide support and meet the needs of the experts at the accident response centre (regulator) where only limited on-line information about the plant status is usually available. The philosophy of this tool is to provide support for the emergency

preparedness and, at the same time, to avoid complicated calculations in order to have faster than real-time predictions [21].

4.5 Linking the BBN to a fast running deterministic code

4.5.1 Basic considerations regarding the use of MARS

A good understanding of the functionality of the Bayesian Belief Network (BBN) and the information that the network uses for the calculations is the basis for the evaluation of the connection with the fast running source term code. The physical quantities and, in general, the observables play a major role in directing the BBN towards the final most probable outcome.

An important issue to address is at what point to start the deterministic code simulation after the BBN based analysis has been initiated and how many readings/findings that are required before starting. The data that can be extracted from the output states of a RASTEP BBN are not sufficient to initiate MAAP or other commonly used severe accident codes. However, RASTEP information can be used as input for the simulations with the MARS software. MARS is designed to use less than 100 parameters as input to give relevant simulations. Early versions of MARS were provided with a very limited set of plant data (around 15 parameters) and the results in the early age of this software tool were already satisfactory according to the experience of CSN [20]. At the same time, in the event of lack of information, the MARS' *tracker* uses a limited set of dynamic plant signals to initialize the MAAP code and then to follow the evolving plant response. The ability of creating an input model for a deterministic code, using less on-line signals than commonly needed, represents an interesting feature in MARS. Even if MARS could serve solely for this purpose that would be a very effective feature to add into RASTEP.

The MARS's *tracker* gets initialized using plant data (or manually inputting data – if necessary) and tracks the evolving plant behaviour. While MARS is tracking the evolving plant response, either after automatically or manually input, the user can initiate a set of *predictors* to perform much faster than real time simulations of alternative (hypothetical) ways the accident could evolve in the future. The fast *predictors* can also run “what-If” evaluations in order to extrapolate deviations from normal operation in the plant and bound the uncertainty of not having solid sets of plant data [22]. The *predictors* do include source term information which is typically coupled to off-site radiological codes such as MAAP4-DOSE, LENA, ARGOS, etc. Therefore, the MARS software performs mechanistic thermal-hydraulic and source term evaluations. One of the big advantages of the MARS *predictors* is that they make it possible to initiate a set of calculations to explore many possible plant states [20][22].

4.5.2 Preparing the input for MARS

To connect the Bayesian network (i.e. plant states) to the input deck of MARS, a description of the available set of BBN nodes has to be given. An example of the available information is shown in Table 4.1. Specifically, the table shows some of the BBN nodes for a generic Swedish BWR. As one can see, the results from the BBN refer mainly to system availabilities and plant states. Each of the entries in Table 4.1 corresponds to a node in the network. After the BBN infers the solutions, each state in the node will end up in a certain state of occurrence with its own probability. This information together with plant data transmitted on-line or manually introduced into an input deck is what MARS will use to start the simulations.

As mentioned above, MARS has to be fed with characteristic parameters from the plant. This means that an accurate selection of parameters has to be made to take into account either

the importance of findings from the BBN with related probabilities, or the use of on-line plant data. It is not straightforward if one has to prefer the use of predicted events (e.g., steam explosion, release from fuel) rather than signals from the measurement instruments at the plant to start the deterministic code. It is certain that the outcomes from the BBN can provide, at least, insights and support for the preparation of the deterministic code input deck. This would perfectly fit the scope of RASTEP and its usability before validation studies and tests will assess good performances in other configurations.

Table 4.1 Example of nodes in the BBN (BWR)

AFW availability	Steam explosion
Containment depressurization line closure (rupture disc)	Reactor building exhaust system filtration
Containment sprays	Reactor Building mode
Containment long-term pressure development	Need for rupture disc depressurization
Containment isolation	Independent RHR system available
AFW sufficiency	Residual Heat Removal sufficient
Core cooling sufficient	RPV failure mode
Corium coolability in lower drywell	Status of bypass to turbine building
Ex-vessel melt coolability	LDW filling system successful
Containment hydrogen combustion	Manual activation of independent RHR system
Containment status	Main feedwater availability
Status of venting	Status of the pressure suppression function
Containment threat	Rupture disc function
ECCS availability	Restart of core cooling
Mode of fuel release	Availability of filtered venting system
Containment rupture due to phenomena	Turbine condenser
Direct containment heating (DCH)	SRV LOCA (unclosed valves)
Recriticality	Availability of depressurization system

Two alternatives for connecting a fast running deterministic code to the BBN are described in the following two sections. Integrated use of the deterministic code within RASTEP represents a case where the MARS code is fed either with data from the plant or with relevant status information predicted by the BBN. Iterative represents a case where the MARS code is initially fed with plant data and information from the BBN. In this case, the scenarios simulated by the deterministic code will subsequently be used to drive new predictions of the BBN and then iteratively fed to MARS.

4.5.3 Integrated use of deterministic code

Figure 4.4 illustrates the integrated use of the fast running deterministic code MARS. According to the block diagram, the input data for both the BBN and the deterministic code could come either as on-line transmitted data from the NPP or it can be entered manually by the user. Signals acquired are analysed and validated, both of the BBN input and the MARS software.

Consequently, the validated input data is fed to the BBN, and the solution for the future plant status is inferred. The results from the BBN are then fed into MARS. This is a crucial step in the proposed RASTEP process because the selection of the final states of the BBN has to be done in accordance with the MARS input specifications.

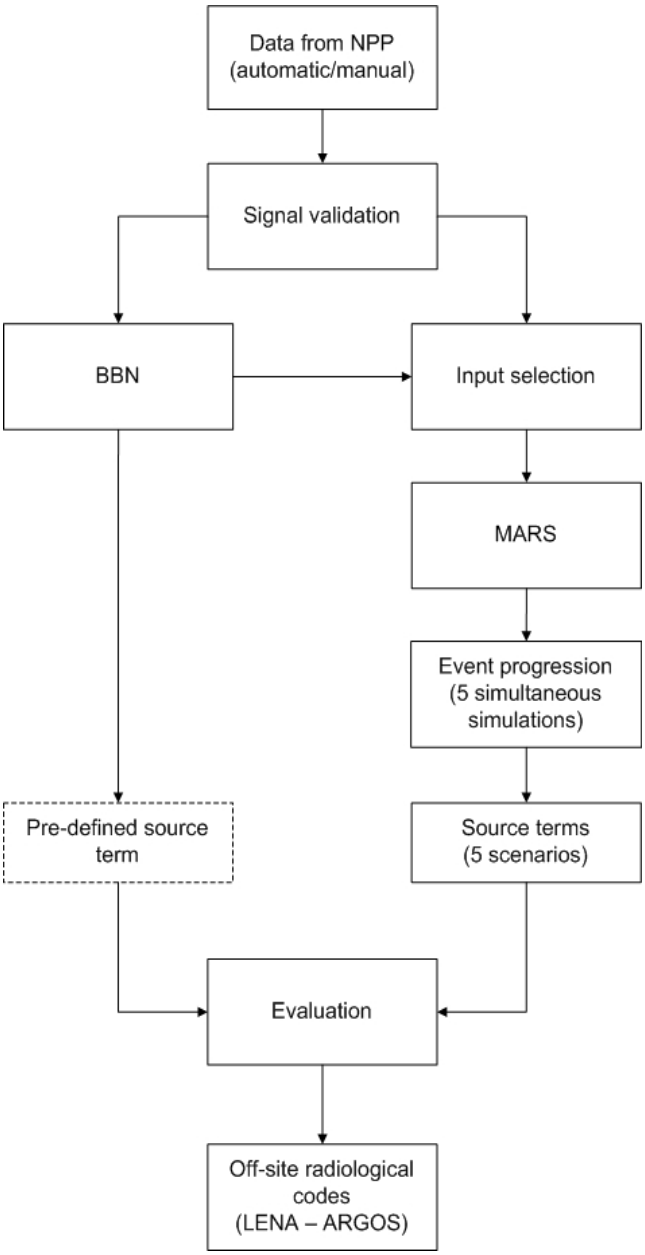


Figure 4.4 MARS linked to a BBN – Integrated mode

It is in the block called *input selection* in Figure 4.4 that the choice between data from the plant physical parameters and from the BBN output has to be made. The ambition is to depend on plant data as little as possible. This means that, if only data for observables are available, no input selection needs to be performed since the only information needed is already available from the BBN inferred results. In accordance with the information provided by Fauske and Associates [22] and CSN [20], the outcomes from the BBN is consistent with the MARS input definition.

Multiple MARS simulations can be performed, up to a maximum number of five, each related to five different plant statuses. This implies that, once the source terms for the predicted accidental scenarios are calculated, an accurate choice between the various alternatives has to be made. For this purpose, the pre-calculated source terms can be used to make comparisons and, according to the release categories defined in the BBN model, selected in

order to give the best estimation. To make the new source terms as realistic as possible a less coarse division into release categories is probably necessary. This important conceptual issue will be described further in section 4.5.5.

4.5.4 Iterative use of deterministic code

An alternative way of connecting the BBN to MARS implies iterative use of MARS, illustrated by Figure 4.5. Thus, the data from the plant are fed to the block *signal validation*. The use of this block serves as screening for either the data coming directly from the plant or manually entered by the operator/user into the BBN. Here the information is captured also from the event progression analysis of MARS and its predicted plant states.

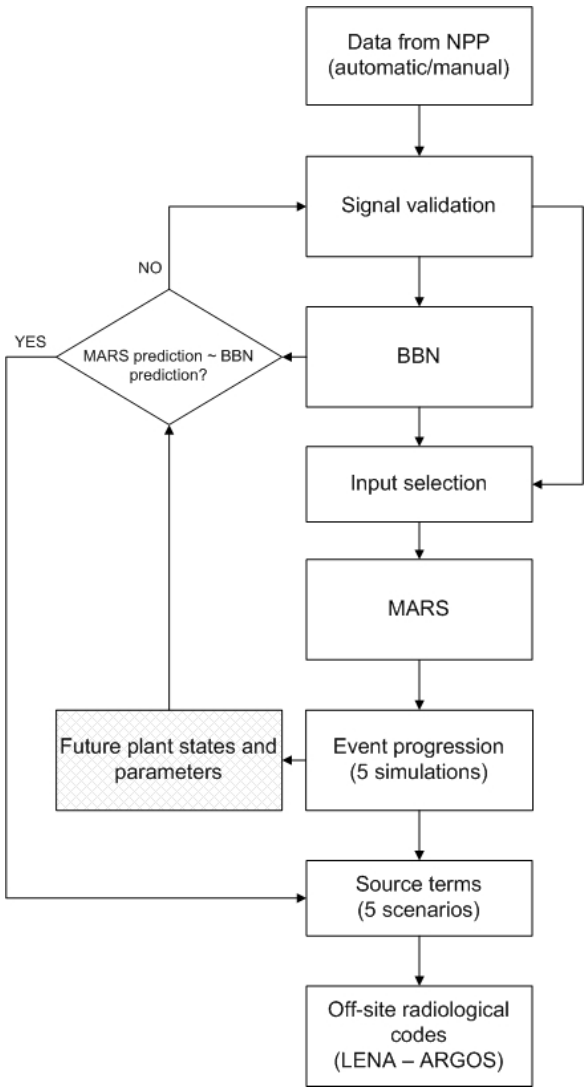


Figure 4.5 MARS linked to the BBN – Iterative mode

Similarly to the integrated use, the data just selected and validated is fed into the BBN, which will infer the solution of the most probable accidental end states. Up to five alternative plant states can be selected from the BBN outcomes, and will subsequent be fed into MARS to run five simultaneous simulations for different accidental scenarios. The parameters from the BBN together with the plant data are then selected to prepare the input for MARS.

After MARS has computed the event progression simulations for the different scenarios, the predicted plant states are sent back to the *signal validation* through a *decision block*. The new plant states have to be extracted from the MAAP simulations and there is still a need to clarify by which means the selection of data to extract has to be made. The decision on how to make this extraction and which type of information to take depends mainly on the quality of the predictions and the difference between the parameters previously selected in the BBN. The available parameters of the plant, at the time of the iteration, also play an important role.

The *decision block* is where the decision whether to iterate the process is made. At this stage the data previously extracted from the deterministic analysis is compared with the BBN outcome. If the most probable end states predicted by the probabilistic module are in accordance or, at least, comparable to those simulated by MARS, the iteration can be stopped. If this is the case the source term referring to the last MARS simulation, contained in the MAAP output, can be accepted and selected for the next step in the scheme. If on the other hand, the MARS predicted plant states are very different from those predicted by the BBN, then an iteration is made, i.e., the information coming from the deterministic simulation is redirected to the *signal validation* again.

In the *signal validation* block, the information is collected directly from the plant (automatic/ manual) or via the *decision block*. These data are then chosen and sent again to the BBN to start the probabilistic prediction. In section 4.5.5 below, the problems related to this integration mode and other questioning issues will be further discussed.

4.5.5 Managing the conceptual framework

As described above for the iterative mode, one of the first issues to address when linking the BBN to MARS concerns the selection of the plant data used by both codes. In this perspective, it is important to understand how the probabilistic prediction in the BBN is meant to drive the deterministic analysis. Furthermore, it is important to decide which information has to be kept and what probability data are relevant in the BBN outcomes that, later on in the coupled configuration, will serve to initiate MARS.

When the deterministic code is driven by the parameters selected from the BBN end states, the results from the simulation need to be ranked in accordance with the initial predicted scenario (probabilistically inferred). This is the way in which RASTEP is supposed to address the prediction of an accidental scenario as described in 4.5.4. The deterministic results are in the form of predicted future plant states and physical parameters.

The approach to select parameters and rank the output of the deterministic code is a challenge. However, it is normal practice when doing MAAP calculations to indicate time intervals for the simulation. The time factor in the deterministic simulations plays an important role both for the analysis of the evolution of single parameters and for the time intervals in which the source terms have to be evaluated. This appears in total accordance with the way in which the selection of the parameter should be made.

It seems probable that characterizing the solutions of MARS also in terms of system damage and quantity of substances in specific volumes of the containment could help the selection of parameters. This would help when it is more complicated to rank the output of the deterministic code. Applying this approach would enable the user to make an easier choice for the comparison with the output of the BBN.

The source term data are still the most important information one should retrieve by using RASTEP. In both analysis approaches defined above (integrated and iterative), the relevant source terms come from the fast running deterministic code. Another problem arising from

this approach is the matching between the source terms extracted from on-the-fly deterministic simulations and those currently available in RASTEP. A way to solve this problem is to identify less coarse release categories both in the BBN model and its pre-calculated source terms and in the output of the deterministic code. This will facilitate the comparisons of the release data for both the codes avoiding wrong interpretations.

Moreover, the predictions of the BBN should be used as a basis in the structure of the entire analysis. If there was no probabilistic usage in RASTEP such as the BBN inferring the solutions, this would merely result in using MARS as a “standalone” code. The power of the BBN model has to be kept and its capabilities better understood with the ambition of creating a tool that is reliable and fast enough to give insights truly reliable during severe accident progression at NPPs.

4.5.6 Suggested approach for source terms definition in RASTEP

The interest in enhancing the source term module of RASTEP is clearly justifiable when looking at how the source term prediction is presently carried out. The basis for the RASTEP source term module is mainly the PSA level 2 and its related MAAP calculations. Severe accident sequences have been modelled in accordance with state-of-the-art practice to assess the releases connected with such sequences.

In the level 2 PSA, release categories are defined and subsequently associated with plant end-states. Specifically, in PSA level 2, each release category is related (through a binning process) to a set of sequences having similar features with respect to event progression, release path, initiating events, and actions put in place to mitigate the consequences. However, the sequences selected from PSA do not necessarily refer to the same plant damage state (PDS) in the actual scenario. For instance, *immediate* and *early* failures occurring in the event of an accident may be considered to be the same with respect to the associated releases. This is one of the reasons why release categories are generally assessed conservatively, resulting in a static behaviour when utilised in RASTEP [23]. Typically, the number of release categories considered in a level 2 PSA study is around 15-20. All release categories are related to specific release paths in the containment volume, the reactor building and in the turbine hall. Each of them refers to specific accidental modes of the plant. These modes are closely connected to the PSA sequence analysis.

The predefined source terms currently modelled in the Bayesian network are not fully accurate and timing aspects of the release are not sufficiently realistic. The ambition is to have source terms which are more related to the accident scenarios in terms of timing and specific volumes of the plants. As suggested previously, the capabilities of the fast running deterministic code are available in a way that the information can match more detailed release data in the sub-volumes of the plant. Thus, more detailed release data in specific volumes of the containment can be assessed using MARS.

The representative nodes (source terms nodes) in the BBN model are currently associated with the level 2 PSA release categories. In order to understand how to capture the new release information given by the linked deterministic code it is necessary to consider some available techniques that may be used as a complement.

4.5.6.1 **PSA sequence analysis**

The PSA for a nuclear power plant can catch all the vulnerability on specific systems and the over-all logical model is capable of representing all the critical sequences leading to unacceptable releases from the plant. The ambition is to investigate those sequences which are not necessarily leading to major releases, but that can still give information on minor releases in specific regions and volumes of the containment. The set of sequences can be

extended and possibly integrate more scenarios related to those consequences which were considered of minor importance. The problem that the analyst will face when attempting this approach is that such sequences are not available according to the ranking and selection normally adopted. Nevertheless, the PSA model offers the opportunity to make this adjustment and it can include as many sequences as desired, incorporating more specific scenarios [24]. The choice of the additional scenarios to include is strictly dependent on how RASTEP is modelled. This implies a modification in the BBN either in the case of the integrated or iterative use (section 4.5.3 and 4.5.4). A redefinition of the predefined source terms would also give the ability to do better comparisons with the source terms provided by MARS. Many advantages could derive from this approach even if no fast running deterministic code is linked to RASTEP.

Another important factor is the time aspects in the release data. The source terms currently extracted from the MAAP calculations behave statically. Introducing new sequences will also help in capturing more relevant parameters and time intervals which were not considered before. According to this description, more MAAP calculations are needed for evaluating cases previously neglected. The computational time required to run all the MAAP simulations would not impact the RASTEP usability because the deterministic calculations are always performed in advance.

4.5.6.2 **MARS source terms**

According to the linking procedure introduced in section 4.5, the MARS software should be able to provide on-the-fly source terms and event progression analysis related to the most probable plant states indicated by the BBN. At this point, the problem is about the way the source terms data have to be included into specific release categories and, later on, how to back-track this information within the linked configuration.

As a first consideration, the information contained in the MARS output has to be referred to specific volumes of the plant. These volumes have to be previously indicated and evaluated by means of other studies. The detailed deterministic analysis in MARS can meet the use of new release categories for very specific volumes of the plant. For instance, if one is interested in evaluating the leakage from the containment in the event of a LOCA, restricting the analysis to a few regions close to the containment wall and indicating a release category for that specific area of the containment would yield a more accurate source terms in many cases [20].

In case new release categories are introduced when MARS is linked, back-tracking this information to the BBN would still be a major concern, as the end-states in the BBN are currently not modelled to match a new release grouping. This means that new techniques to interpret this deterministic information are needed, involving the addition of new nodes to the BBN.

4.5.6.3 **Remodelling the BBN – Adding new nodes and/or redefining CPTs**

In order to include new source terms data, the BBN model has to be reorganized and remodelled in accordance with the grouping of the release categories. The release information could come either from new sequences in the PSA or from a fast running deterministic code. The nodes currently modelled in the BBN are possibly not in a sufficient number to accommodate all the information needed.

When source terms accounting for volumes in the plant not previously considered in the network are present, a new node and/or a new set of nodes should be created in the BBN. The end-states in the BBN need to match the release data coming from the deterministic code and, at the same time, give the correct outcome with the related probability. If MARS or any other deterministic code were to be linked to RASTEP, the data extracted from the

deterministic simulations have to be integrated in the model assuring that the proper information is contained in the parent nodes, all the connections are established for all the updated source terms and CPTs well described.

In case new sequences are evaluated in the PSA studies, they have to be mapped into the BBN model and possibly specific new nodes for other systems in the plant have to be created. However, it is very probable that only the CPTs have to be redefined and no additional nodes have to be created in the network.

4.6 Discussion

4.6.1 Review of the approaches

The review activity together with the linking strategy performed within this study has generated a set of criteria associated with the definition of a dynamic source term module in RASTEP. The new source term module should be:

1. *Realistic* – i.e. conservatism needs to be reduced
2. *Accurate* – i.e. on-line plant data as basis for the predictions
3. *Adaptable* – i.e. use probabilistic predictions as basis to perform new calculations

The use of RASTEP in its current configuration (i.e., without any integration with a fast running deterministic code) has demonstrated that the predicted most probable plant statuses are usually indicating quite accurately the future accidental scenarios. This is mainly due to the strength of the PSA modelling, the accurate mapping of the key plant parameters, and the systems interrelations modelled in the BBN. However, using a limited set of predefined source terms make the current approach not realistic enough. In particular, the overall logic of the tool is still based on static plant models which constitute the foundation for the calculations. The first criterion in the list above is, then, not fulfilled by using the actual tool if no modifications are made in the model.

Using a fast running deterministic code such as MARS could provide great advantages in performing on-the-fly calculations. Furthermore, the same live plant data necessary for the BBN can be used with MARS. The ability of MARS to perform what-if-analyses (e.g., when a major loss of signals from the control room occurs) is also highly attractive.

If RASTEP would be linked with MARS, the same end states currently resulting from the probabilistic inference appear to be a good fit for the input of a deterministic code. Even if the end states of the BBN are grouped too coarsely to be used as input parameters, MARS can still adapt to use only those data and probably still give reasonable results [22]. However, the information that can be captured from the BBN is not necessarily the final states of the predictions. More valuable information can be extracted from the hidden nodes within the probabilistic model giving better understanding of the accidental progression in the deterministic simulation.

As described in section 4.5.2, understanding which approach is favourable for generating the input for MARS is still a big challenge. The easiest way would be feeding the deterministic software directly with live plant data. Moreover, the use of the BBN to infer the most likely plant state is a very powerful feature. Even in the case on-line data transmission is implemented, it could still be very beneficial to use the BBN module to feed MARS. The boundary conditions related to the accidental scenario could be assessed rapidly in an early phase of the accident.

When it comes to addressing the feasibility of the implementation of MARS in the RASTEP tool, it can be seen that MARS has the considerable advantage of being based on MAAP models. MAAP models for all Swedish nuclear power plants already exist and, therefore, a large body of the modelling competence is available [25]. In contrast, other deterministic tools such as ADAM would need that plant specific models have to be created.

4.6.2 Questions to be answered

Two ways of using a fast running deterministic code (i.e., MARS) with RASTEP were indicated and analysed, i.e., integrated use and iterative use. In both approaches, the BBN will be giving insights into the plant status during an early stage of the accident and the deterministic code will be functioning better, later on, when more reliable information is available.

In iterative use the strength of the BBN will be used more directly. The BBN prediction will be important when the progression of the accidental scenario resulting from MARS is assessed and a cross comparison of plant parameters has to be made (section 4.5.4). The iterative use seems to be more adequate and functional than the one where no iteration is included. The possibility to verify that the on-the-fly predictions given by the deterministic code are in line with BBN prediction represents an advantage in order to assess that the simulations follow the real accidental scenarios. If one is interested only in adjusting the pre-calculated source terms currently used in the source term module, before attempting the actual implementation of a deterministic code into RASTEP, studies should be carried out to determine what characteristic times the user should investigate in the MAAP output table.

In view of the above, some key questions have to be answered during the preparation of a coupled system:

- At what stage in the accident scenario should the MARS' source term be considered reliable?
- How to combine the information (on-line plant data and BBN prediction of plant status) in order to generate a reliable source term?
- What are the most important parameters in the BBN predictions to be used in MARS?
- How to technically redefine the release categories definition both in the PSA and BBN model?
- How many source terms are needed in order to have realistic predictions?
- Is the automatic selection of the source term the best way to retrieve the results?
- What are the relevant accident sequences to be considered if a benchmarking of the coupled system has to be performed?
- How to technically link the MARS software with RASTEP?
- Does it really make sense to have more detailed and realistic source term data in all situations?
- How to determine set points for the deterministic code to start its calculations (before/after the BBN suggests a release path)?
- Can the information from MARS totally replace the predefined source terms or not?
- Is it reasonable to consider a "stand alone" use of MARS?

Moreover, the mathematical framework as well as the software infrastructure necessary for execution of the linking needs to be specified and tested.

5. EVALUATION OF CODES - COMPARISON OF MAAP AND MELCOR AS DETERMINISTIC TOOLS WITHIN RASTEP

5.1 Introduction

This chapter is based on the outcome of a pilot project [26] on comparison between MAAP and MELCOR, and of a M.Sc. Thesis project initiated in order to address some issues identified in the pilot [27]; the M.Sc. Thesis work is currently being performed and will be finalised towards the end of 2013. Therefore, results presented are preliminary and subject to completion and confirmation.

The aim of this sub-project is to evaluate whether MELCOR could function as a complement to MAAP in the field of severe accident analysis for nuclear power plants in Sweden, i.e. in the framework of RASTEP. This involves addressing the following:

- Solving the issues and problems identified in pilot phase of comparison between MAAP and MELCOR [26].
- Extending the basis with results from several different scenarios for comparison between MAAP and MELCOR.
- Answering the main question, i.e. whether other codes, e.g. MELCOR, may be used as a complement to MAAP in the RASTEP framework.

As stated, the main task is to solve the known issues, and straighten out some question regarding MELCOR raised in the pilot application. To do so the code itself (MELCOR) has to be understood. It will also be important to obtain knowledge about preceding work with MELCOR. As the latest version of MELCOR (version 2.1) is currently not fully accepted by the industry, an older version (1.8.6) was used. There is also a model of the O3 reactor available for version 1.8.6, i.e. the same as previously used in the pilot application.

The existing MELCOR model will be used to run scenarios and to compare to results from previous simulations made with MAAP. MAAP results are seen as reference, but the results will be concretized and discussed, with the possibility in mind that there might be more accurate interpretations. However, anomalies found in the MELCOR results will be traced and explained, and if possible (and necessary), also corrected. After simulations, comparisons and interpretations, an attempt will be made to answer whether other codes, e.g. MELCOR, may be used as a complement to MAAP in the RASTEP framework.

5.2 Background

RASTEP itself is not designed to calculate quantities of a release, and this is derived from simulation programs especially designed for the task. The output, or source term, (i.e. magnitude and composition of the release) from a simulation is then transferred, and used by, RASTEP for a certain chain of events. The present solution is to have a library containing pre-calculated source terms for different scenarios.

It is of paramount importance that severe accident analyses are made in the best possible manner. Comparisons between different codes could function as a quality check of the methods used in the safety work around NPPs. It is likely that different codes will have different strengths and weaknesses. In this way, a deeper knowledge and awareness of the capabilities of alternative tools is obtained. One possible outcome is that the currently used code (MAAP) is the best alternative for simulations of certain sequences, whilst another code shows better performance for other types of scenarios. Different codes could also prove themselves the strongest alternative within different parts of the same simulation, e.g. one of

the codes shows best performance in the initial part of the simulation, and another code performs better towards the middle, or the end, of the sequence. In such cases the best result would be given if the codes were ran in parallel. A comparison between different codes could result in two, or more, codes being used in parallel. A clear benefit from this is the effect of a second opinion (also stated in [30]).

Since RASTEP is supposed to make predictions in real time, it must also be very flexible in terms of how a particular scenario evolves. This sets high demands in terms of flexibility, and thus a sufficiently large library of pre-calculated source terms is required, since no realistic option for real time calculation of the source term is available at present. To build a large library of source terms, the use of some alternative to MAAP could prove to be a valuable asset. This would, indirectly, also increase the credibility for RASTEP, i.e. since its level of accuracy highly depends on external codes. Finally, it is important to perform comparisons between different codes for severe accident analyses, e.g. to identify weaknesses in order to make improvements. Thus it is also important to obtain knowledge about how to compare different codes in the best possible way.

5.3 Objectives

Two codes have been selected for comparisons between different scenarios, i.e., MELCOR developed by Sandia National Laboratories International for the Nuclear Regulatory Commission (NRC), and MAAP developed by Fauske & Associates LCC. The reasons for these choices, apart from the fact that these codes are widely used state of the art tools for SA analysis, are that simulation results from MAAP are available and that access to MELCOR could be provided via SSM. MAAP also functions a kind of reference tool since it is the standard code for these kinds of simulations in Sweden today.

MAAP simulation results will be used for comparison with calculations done by MELCOR. The objective will be to seek out differences, explain them or correct them, in order to determine the validity of MELCOR and if it could function alongside or as an alternative for MAAP. A model of Oskarshamn 3 developed for MELCOR 1.8.6, has been adapted for severe accident analysis by the Royal Institute of Technology in Stockholm (KTH) [30]. The project was funded by SSM and supported by OKG (the operator of the O3 reactor).

In the recent pilot comparison between MAAP and MELCOR [26], the 1.8.6 model was used along with data from a previously made simulation in MAAP. A scenario previously simulated in MAAP was also introduced to MELCOR. The analysis case is a station blackout (SBO where all manual actions regarding recovery and consequence mitigating systems (RAMA systems) fail. However, all automatic functions of the RAMA systems are actuated. Several differences between MELCOR and MAAP were identified from the results of the simulation, and some issues and questions regarding MELCOR were identified [26]. These are specified below:

- Reactor vessel melt through occurs earlier in MAAP (4.0 h) than in MELCOR (7.0 h). The debris is also located at the bottom of lower plenum for a longer period of time in MELCOR.
- The containment pressure relief system (362) is activated earlier in MAAP (4.6 h) than in MELCOR (9.9 h), i.e. the pressure increase is more rapid in MAAP.
- The temperature difference between the debris and the surrounding water in lower drywell is about 600 °C in MELCOR after 24 h. At that time the corresponding temperature difference in MAAP was only 6 °C. Thus the debris is not coolable in the MELCOR simulation.
- No molten core concrete interaction (MCCI) could be observed in either of the two simulations (i.e. both MELCOR and MAAP show the same results).

- The decontamination factor was only 7 in MELCOR, which is far below the requirement of 100. Decontamination factors up to 500 have been observed in the verification program of the venturi scrubbers.
- MELCOR needed 13.3 h to complete the calculus, whereas MAAP needed 1 h. MELCOR is known to be very mechanistic. MAAP on the other hand uses several simplifications in its calculations. It is however likely that there are other explanations to this big difference in calculation time.
- The radioactive release could not be plotted in a satisfactory way by MELCOR (i.e. problem with the plot routine).

A total number of additional scenarios are planned to be analysed as part of the on-going project, with various statuses of systems related to lower drywell filling, pressure suppression in wetwell, independent containment spray system, and containment filtered venting.

5.4 Preliminary results

This part of the project was carried out through simulation of an analysis case for comparison to MAAP results. Since the MAAP model is the one officially used in PSA etc, it was used as reference for these simulations. Both MELCOR and MAAP provide the possibility to plot the simulation results in MS Excel, making comparisons fairly straightforward. Anomalies found in the simulation results, were to be explained and traced to their origin within the O3 model. Although the MAAP simulations were seen as the reference, the MELCOR interpretations might of course also be the correct ones in cases where discrepancies were identified. Some differences could not be traced back to the model, i.e. they are likely to be caused by different quantification models being used within the two codes.

Simulation time
The time to complete a simulation in MELCOR was several times longer than the time demanded for MAAP to complete the same sequence. It was found at an early stage that MELCOR could be set to take longer time steps. In this way the quantification time was reduced to 1/3. In a later stage, the Multi-Venturi Scrubber System (MVSS) model was modified, which improved the MVSS technical characteristics, but also reduced the quantification time to approximately 1/10 (to about 1 hour) compared to the initial case.

The activation of system 362
The containment pressure relief system, i.e. system 362, was set to activate at 5.0 bar(a) in the MELCOR model, while the threshold for this system was set to 5.5 bar(a) in the MAAP model. This can be spotted in a comparison between the pressure curves, i.e., the peak of the pressure curve in MELCOR is lower compared to the one from the MAAP simulation. Thus, the MELCOR model was adjusted accordingly. A consequence from this was that the time before system 362 initiated increased. Thus the timing differences between MAAP and MELCOR increased slightly.

Quenching of the debris
One of the issues, pointed out in [26] was that the temperature difference between the internals of the debris and the surrounding water was too large. This in turn indicated that the debris did not break up into smaller pieces forming a pile of gravel-like particles at the bottom of lower drywell. Instead it remained intact, forming one large mass of unquenchable debris. According to KHT experiments, this would evidently not be the case in reality. The simulations displayed a much slower cooling inside the debris from the MELCOR results compared with the results from MAAP. The temperature of the debris also stabilized at a much higher temperature in the MELCOR results.

The final solution to this issue was to make use of the parameters for regulation of the heat transportation coefficients for the debris built into MELCOR, i.e. heat conduction-, heat transfer- and boiling coefficients [28]. Each of the coefficients was altered separately at first, since it was unclear how the built in model handled them, i.e. it was possible that some of them were a limiting factor. For instance the built in model is supposed to have a complete set of boiling curves [29], thus increment of only the heat conduction- and transfer coefficients was thought to be enough. However, it was found that also the boiling parameter had to be altered for sufficient heat transportation from the debris to the surrounding water.

The increment of these coefficients would also increase the likelihood of MCCI interactions. However, these were continuously absent in the simulation results. Due to more efficient transfer of the decay heat from the debris to the water pool, the debris bed was quenched.

The time to vessel melt through and pressure relief of the containment
No matter how much the heat coefficients were increased, the time before system 362 was activated could not be reduced enough. However, during the investigation of this issue, a new anomaly was discovered, i.e., that the decay heat was much lower in MELCOR than in MAAP. It was due to a nonstandard model (ORIGEN) for the decay heat having been implemented into the O3 model. Instead of ORIGEN the decay heat model was changed to the standard ANS model, and adjusted (amplified by a factor of 1.17), for resemblance with the decay heat in MAAP. Due to a higher decay heat, the vessel melt through now occurred at an earlier stage. The pressure increase rate was more rapid and, ultimately, the containment pressure relief system (362) was initiated sooner, resulting in an earlier radioactive release into the environment, i.e. since this happens as 362 gets activated.

Although there still remained differences, between the results from MAAP and MELCOR, they have now been reduced considerably. However the melt-through still occurred earlier in MAAP, and the pressure increase in the containment was still more rapid compared to MELCOR.

Pressure fluctuations
The pressure curve of the containment fluctuates after the pressure relief through system 362. The reason for this behaviour follows from a thermo hydraulic competition between the decay heat in the debris and the cold water that flows into drywell from the wetwell. The water in the drywell is heated and evaporated by the decay heat, resulting in a pressure increase. At the same time, pressurized vapour flows through the PS (pressure suppression) system, from the upper drywell into the wetwell, causing "cold" water to flow into the lower drywell. Thus, the evaporation, inside the drywell, stops and the pressure decreases until the water is heated sufficiently for a new cycle.

Exactly the same phenomenon could be spotted in earlier MAAP simulations. However, due to previous debates on whether this cyclic behaviour is actually real or simply a mathematical artefact it was decided to suppress this within MAAP simulations. The reason for this is that the removal of these fluctuations, real or not, is more conservative than to have them still, i.e. since the fluctuations would provide a pump-like effect inside the containment, enabling the safety systems to work more effectively.

The mitigation of the fluctuations was done by narrowing the path way of system 358, the system that enables water to flow from the wetwell into the lower drywell, so that cold water would flow in a more controlled manner. Thus the same approach was brought to the MELCOR model.

Modification of the MVSS model

During the simulations, an anomaly in the radioactive release curves was discovered in the MELCOR results. It could be traced to the implementation of system 362, or more specifically the MVSS, within the O3-model. The MVSS is modelled as a pool of water into which the radioactive gases are released. The decontamination factor consists only of water scrubbing. Therefore, as the water starts to boil, the scrubbing effect is reduced considerably. This is seen in the release curve as a large step where the level of release is increased many times.

In reality there is no such reduction in efficiency of the MVSS. During its development, this system was put through extensive and thorough testing. For extra conservatism, water scrubbing-effects were never taken into account, i.e. boiling was assumed to occur, in the scrubber, soon after activation. Instead the evaluation solely considered the inertial scrubbing created by the venturi nozzles, i.e. high velocity collisions, with high frequency, between water and contaminated gas. The efficiency of this system is insensitive to boiling. Furthermore the decontamination factor was proven at least five times higher than the required value of 100 [31].

Since the function could be seen as, more or less, constant during the simulation. It was considered a better option to use a simpler model of the MVSS. Therefore, the original model was exchanged to an aerosol filter with a constant decontamination factor, equal to the value used in the MAAP simulation. The result from this modification is visualized in Figure 5.1. Furthermore, a filter is also a less demanding element to simulate than a separate volume with water boiling and scrubbing. As a result, the simulation time was decreased by 1/3.

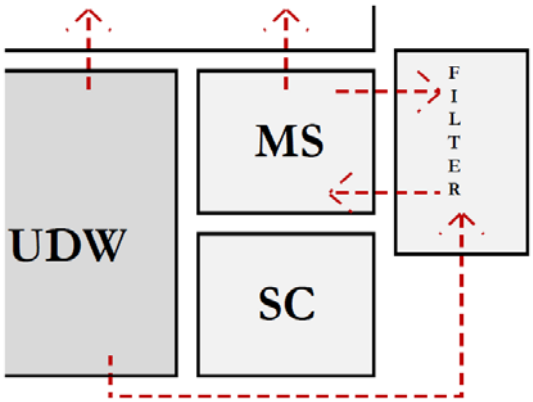


Figure 5.1: The scrubber is replaced by a filter in the O3 model.

6. DEVELOPMENT OF A METHOD FOR CONSISTENT AND ROBUST DETERMINATION OF CONDITIONAL PROBABILITY TABLES

6.1 Introduction

This part of the project aims at developing a general method where beliefs can be included in a systematic way to define CPTs in the BBN. This semi-quantitative method should have qualities which enable the BBN constructor to find uncertainties in the network and study the importance of different parts of the BBN along with determining CPTs. Therefore, sensitivity analysis regarding parameters and model structure of the BBN must be included in the method. This analysis should also show how parameters and model structure influence the prediction of source terms.

The intention of the general method is to give a consistent approach of combining the different parts in building a reliable BBN, where the main parts are:

- *Network structure* – provide a graphical representation of the relationship between events.
- *Probability estimation* – define probabilities in the CPTs with a focus on methods for expert elicitation.
- *Sensitivity analysis* – identify how observations and probability uncertainties affect the network's output.
- *Verification and validation* – make sure that the network's output is reliable and makes sense.

Priority is given to the *Probability estimation* and *Sensitivity analysis* parts.

RASTEP is designed to support decisions in case of a severe accident. Thus, it is crucial that the predictions are as credible and reliable as possible, which in turn increases the requirements on validity, robustness, and transparency of the BBN. Therefore it is important to analyse how the choice of model parameters and model structure affects the network's output.

The determination of the CPTs depends on the characteristics of the node to which a CPT belongs. Some nodes require special attention since there is no or very little data to derive the probabilities from and hence the CPTs have to be based on expert judgement. Developing CPTs using experts can be time consuming since the number of probabilities that need to be estimated is usually very large. Furthermore, using expert judgment may induce large uncertainties in the network and it is desirable to know how the uncertainties affect the reliability of the output.

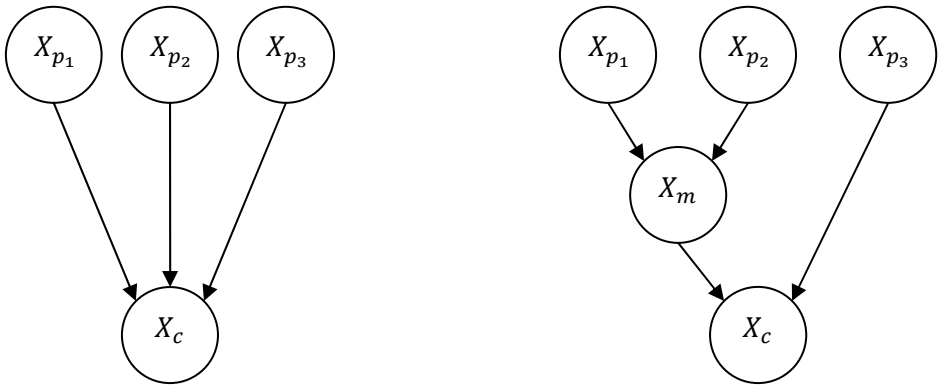
The aim therefore is to propose a method for the definition of a relevant and defendable set of conditional probabilities in a BBN that also considers uncertainties in the network.

6.2 Network Structure

The qualitative part, i.e. the nodes and arcs, forms the basis of the network and is the first part to be determined when building a BBN. How to set up the network structure will not be discussed in detail. However, a way of modifying the structure of the BBN to make the probabilities in CPTs easier to assess is *node divorcing* which is presented briefly below.

The technique of node divorcing reduces the number of combination of parent states for a node by introducing a mediating node. This node becomes the parent of the original child

node and the child node of the divorced parents. In Figure 6.1 the nodes X_{p_1} and X_{p_2} have been divorced from the child node, X_c, by the mediating node X_m. If the nodes in Figure 6.1 all have four states the CPT for X_c requires 256 probabilities to be assessed. However, if X_m is introduced, also having four states, the CPTs for both X_m and X_c contain 64 probabilities reducing the total number of probabilities to be assessed to 128. Node divorcing only works for discrete nodes and it is only effective if the number of combinations of states from X_{p_1} and X_{p_2} is greater than the number of states in X_m. [32] [33]



a) Original b) Divorced

Figure 6.1 Node divorcing. X_m divorces the nodes X_{p_1} and X_{p_2} from X_c

A property of a BBN is that the immediate successors of a node should be conditionally independent given that node. If the BBN does not satisfy this property it indicates that a hidden node should be introduced [34]. The introduction of a hidden node is done using the same technique as for node divorcing.

6.3 Probability estimation

When the network structure has been set up, the quantitative part of the network needs to be defined, i.e. the probabilities in the CPTs. If no data is available for the determination of probabilities, the source of probabilistic information is expert judgement [35]. Using experts' beliefs to assign conditional probabilities is called expert elicitation.

The main steps of the elicitation process are:

- determine what the experts need to elicit,
- select experts,
- expert elicitation, where the experts may use an elicitation method to assign probabilities,
- if there are several experts, combine their assessments, and
- document the process and the result. [36]

Due to the structure of a BBN the number of probabilities that populate a CPT grows rapidly with the number of parent nodes related to that CPT, e.g. a child node with i states and N parents each having k states demands $i \cdot k^N$ probabilities to populate its CPT. Thus, the elicitation can be very time consuming.

For an expert it can be particularly difficult to assign probabilities for events that are very rare. Therefore, different methods to help the expert and to systematise the elicitation have been developed. There are two types of elicitation methods; *Elicitation of a single probability* and *Elicitation of a full conditional probability table*. The methods presented below are findings from literature studies. Methods for eliciting single probabilities were more frequently used than elicitation of full CPTs.

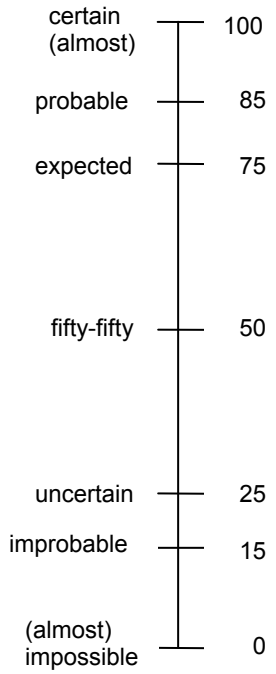
6.3.1 Elicitation of a single probability

This type of elicitation can further be divided into two groups; direct methods and indirect methods. In direct methods experts should give their degree of belief as a number directly, e.g. a probability, whilst for indirect methods the expert will make a decision from which his belief is inferred. [37]

6.3.1.1 **Probability scale**

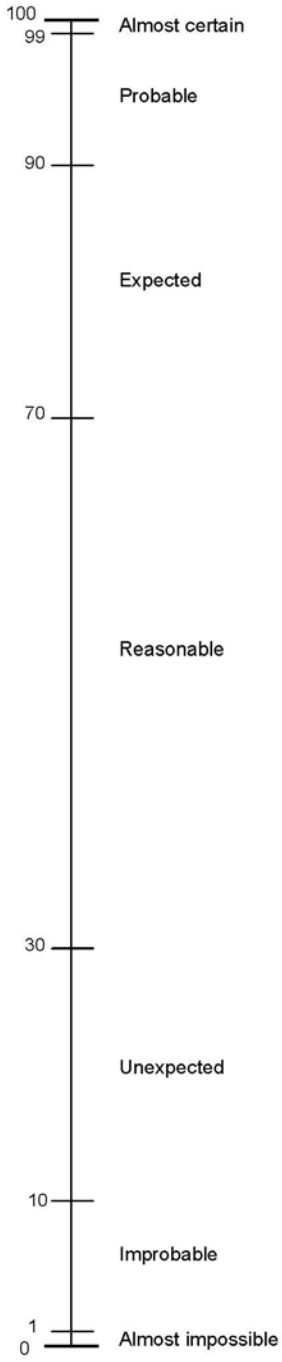
A numerical probability scale is a well-known direct method. It is a horizontal or vertical line showing the probability either in the interval 0-1or 0-100%. The line is divided into several non-overlapping intervals, where 5-7 is the optimal number of intervals. For each probability that the expert assess he puts a mark, which corresponds to his belief, on the scale and the probability can then be determined by measuring the distance between the mark and the 0.

Since not all experts are familiar with probabilities and are more comfortable expressing their beliefs with words there are scales with both verbal and numerical intervals, see Figure 6.2. As a result of an application of the suggested methodology to the O3 BBN model, a further refinement of the probability scale was developed, see Figure 6.3.



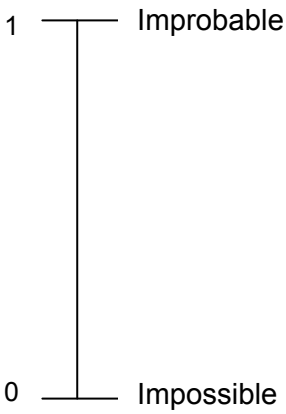
Verbal description of probability	Probability interval
high	0.7 - 1.0
moderate	0.3 - 0.7
low	0.05 - 0.3
very low	0.001 - 0.05
extremely low	10^{-6} - 0.001
negligible	0 - 10^{-6}

Figure 6.2 Verbal and numerical probability scale, represented as a scale and as a table.

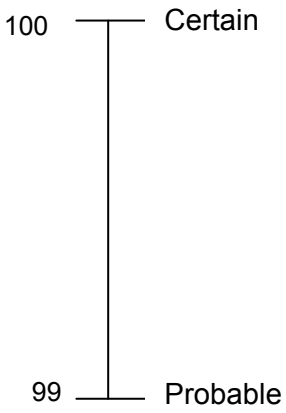


Verbal description of the possibility of the event	Probability interval
Almost certain	99 – 100%
Probable	90 – 99%
Expected	70 – 90%
Reasonable	30 – 70%
Unexpected	10 – 30%
Improbable	1 – 10%
Almost impossible	0 – 1%

Probability intervals for the probability scale.



Scale of the interval "Almost impossible".



Scale of the interval "Almost certain".

Figure 6.3 Suggested probability scale for the O3 network, table with probability intervals and subscales for the intervals "Almost impossible" and "Almost certain".

The use of probability scale is advantageous since it is easy for the expert to express his beliefs in a fast and distinct way. However, when using a probability scale the assessments are prone to have scaling biases, such as centring and spacing effects. Centring is the tendency of experts to use the centre of the scale and spacing is the tendency of experts to divide their responses more or less evenly over the scale. [37] [38] [39]

6.3.1.2 Gamble-like methods

When an expert finds it hard to express his belief of an event as a number, a gamble-like method can be used. The basic idea of gamble-like methods is to infer the expert's assessment of a probability from his behaviour in a controlled situation. Gamble-like methods are indirect methods that are designed to represent controlled situations. Two types of gamble-like methods can be distinguished.

In the *certain-equivalent* gamble an expert chooses between an exact reward, *x*, and a probabilistic alternative where the reward, *r,* depends on the probability, *p*, to be assessed. The exact reward is then altered until the expert is indifferent between the two choices. The probability, *p*, can be calculated as *p=x/r.*

In the *lottery-equivalent* gamble the expert chooses between a lottery where the outcome depends on a given probability and a lottery that depends on the probability, *p*, to be assessed. Altering *p* until the expert is indifferent between the two lotteries give the value of *p.*

Using gamble-like methods can suppress biases as centring and spacing but they are instead influenced by risk attitudes, especially certain-equivalent gamble. Two other drawbacks of gamble-like methods are that they are complicated for an expert to learn and are very time-consuming. [37] [38].

6.3.1.3

6.3.1.4 Probability wheel

The probability wheel is an indirect method that is not influenced by risk attitudes. It is usually a circle divided into two sections. The sections are altered until the expert believes he can spin a pointer and the probability that it will stop in a section is equivalent to the probability being assessed. Probability wheel is very similar to a direct method but a drawback is that it cannot elicit small or large probabilities. [37] [38]

6.3.2 Elicitation of a full conditional probability table

In this section three methods of elicitation of a full CPT, using a reduced number of assessments, will be discussed. Their aim is to generate a full CPT without the expert assigning each probability individually. In the literature other methods exist with this purpose but the advantage of the *Likelihood method*, the *EBBN method* and the *Weighted sum algorithm* is that they are not restricted to binary nodes.

To illustrate how the methods work in practice; guidelines are created as a result from testing the methods on several nodes. Each method will then be used on a node, *Battery voltage*, from the network *Car Diagnosis 2* available in Netica, see Table 6.1 [40]. As a measure of accuracy the mean, m_{abs}, of the absolute difference between the original CPT given in Netica and the CPT generated by an elicitation method is calculated. The methods are used on additional nodes and the mean will be used to compare the methods.

The approach for testing the methods is that, in the absence of a real expert, the original CPT has been used as the knowledge of an expert. Consequently, if a method requires probabilities as assignments, these have been taken from the original CPT. If weights are required they are based on the characteristics of the original CPT.

Table 6.1 The original CPT for the node Battery voltage from the network Car Diagnosis 2 in Netica.

		Battery voltage		
Charging system	Battery age	strong	weak	dead
okay	new	0.950	0.040	0.010
okay	old	0.800	0.150	0.050
okay	very old	0.600	0.300	0.100
faulty	new	0.008	0.300	0.692
faulty	old	0.004	0.200	0.796
faulty	very old	0.002	0.100	0.898

6.3.2.1 **Likelihood method**

This method takes its starting point in Bayes' theorem, which can be written as

$$P\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right) \propto L\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right) P(x_c),$$

where $P\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right)$ is the conditional probability and $L\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right)$ the likelihood of a child state, x_c, given the states for each of the parent nodes, x_{p_i}. The prior probability for x_c is $P(x_c)$ and in this method $P(x_c)$ will be derived from a typical distribution, T_{x_c}. A typical distribution describes the normal state of affairs and its specific form is not of importance. As an example the typical distribution for a node with three discrete states might be [0.25 0.50 0.25] but it could also take the form of a continuous distribution. By focusing on the likelihood, instead of the conditional probability, the elicitation procedure becomes easier since the expert does not have to remember the relative frequency of different states of the child node and hence the method is called the likelihood method.

Using the log likelihood, instead of the likelihood, is convenient since the log likelihood covers a smaller range of values. Assuming that the parent nodes are independent the log likelihood can be expressed as

$$\log_b L\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right) = \sum_{i=1}^{N} \gamma_{x_c x_{p_i}}$$

$$\Leftrightarrow$$

$$L\left(x_c \middle| x_{p_1}, x_{p_2}, \dots, x_{p_N}\right) = b^{\left(\gamma_{x_c x_{p_1}} + \gamma_{x_c x_{p_2}} + \dots + \gamma_{x_c x_{p_N}}\right)}$$

and a more convenient variant is to define $\gamma_{x_c x_{p_i}} = \beta_{x_c} \alpha_{x_{p_i}}$.

The elicitation procedure is then accomplished by letting the expert provide the following information:

1. a typical distribution, T_{x_c},
2. the base, b,
3. a weighting factor for each state of the child node, β_{x_c},
4. a weighting factor for each state of the parent nodes, $\alpha_{x_{p_i}}$.

Given this information the CPT can be calculated using the equations above. It can be seen that if the parameter $\gamma_{x_c x_{p_i}}$ is positive, the likelihood for that combination of x_c and x_{p_i} will

increase, and if the parameter is negative the likelihood will decrease. The value of the likelihood determines how the probability of the child node, $P(x_c|x_{p_1}, x_{p_2}, \dots, x_{p_N})$, shifts in relation to T_{x_c}. The base b can be set so that the values of $\gamma_{x_c x_{p_i}}$ are of desired magnitude. Advantages of the likelihood method are that only one value for each state of each parent node needs to be specified and that the expert is asked to give influence weights instead of probabilities. [41] [42]

Some other advantages of this method are that it works even if a node only has one parent and the algorithm that generates the CPT is easy to implement. If the child node has more than three states this method will be very complex and therefore only guidelines for nodes with two and three states are stated.

Two states

b A suitable start for b is 10. If a smaller/larger range of the probabilities is desired b should be smaller/larger.

T_{x_c} If possible, use a typical distribution for the states in the child node. If there is none use a uniform distribution.

β_{x_c} Let the weights of the child states have opposite signs, e.g. [-1 1].

$\alpha_{x_{p_i}}$ Start with one of the parents and decide for which of its states each child state should have a high probability. The parent state that results in a high probability of a child state should have the same sign as that child state's weight. A suitable assignment for the first parent is [-1…0…1]. The other parents' weights are then assigned in the same way except that the sign and magnitude of the first parent weights must be considered as well. A reasonable interval for the weights are [-3…3].

Three states

b A suitable start for b is 10. If a smaller/larger range of the probabilities is desired b should be smaller/larger.

T_{x_c} If it is known that one of the child states, independent of the combination of parents states, always will have approximately the same probability, first assign T_{x_c} so that this state have the maximum value of the probability for all of the parent combinations. Then distribute the probabilities for the two other states uniformly. If all three states can have high probabilities assign a high probability to one of the states and let the two others be uniformly distributed, e.g. $T_{x_c} =$ [0.9 0.05 0.05].

β_{x_c} If a weight is equal to 0 it does not shift $P(x_c|x_{p_1}, x_{p_2}, \dots, x_{p_N})$ away from T_{x_c}. Hence, assign the state that was first assigned in T_{x_c} with 0 and then assign the other states with -1 and 1.

$\alpha_{x_{p_i}}$ Use the same reasoning as for two states but the assignments are only based on the weights of the child states that are not equal to 0.

When using this method to generate the CPT for the node *Battery Voltage* the following assignments were made:

$b = 10$
$T_{x_c} =$ [1/3 1/3 1/3]
$\beta_{x_c} =$ [-1 0 1]
$\alpha_{x_{p_1}} =$ [-1 1]
$\alpha_{x_{p_2}} =$ [-0.5 0 0.5].

These assignments resulted in the CPT in Table 6.2 and the mean, m_{abs}, was equal to 0.054.

Table 6.2 Likelihood method used on the node *Battery Voltage.*

		Battery voltage		
Charging system	Battery age	strong	weak	dead
okay	new	0.968	0.031	0.001
okay	old	0.901	0.090	0.009
okay	very old	0.706	0.223	0.071
faulty	new	0.071	0.223	0.706
faulty	old	0.009	0.090	0.901
faulty	very old	0.001	0.031	0.968

6.3.2.2 **EBBN method**

The EBBN method (an elicitation method for BBNs) requires only a limited amount of elicited probabilities from an expert to derive a node's CPT [35]. It uses piecewise linear interpolation, based on the ranks of the parent nodes' states, to determine the CPT. The method requires that the states of the child node, X_c, and the states of the parent nodes, $pa(X_c)$, can be ordered on the form low to high. Before the actual method is described, some definitions are introduced.

X_c The node whose CPT is to be determined, where x_c^{min} and x_c^{max} are the lowest and highest ordered state of X_c, respectively.

$pa(X_c)$ The set of parent nodes of X_c.

a An assignment of states of the parents, $pa(X_c)$.

a_{neg} The assignment in which all the parent nodes are in their most favourable state for low ordered states of X_c.

a_{pos} The assignment in which all the parent nodes are in their most favourable state for high ordered states of X_c.

a_{neg,p_i^+} The assignment of $pa(X_c)$ in which $X_{p_i} \in pa(X_c)$ is in its most favourable state for high ordered states of X_c and all $X_q \in pa(X_c) \backslash X_{p_i}$ are in their least favourable state for higher ordered states of X_c.

$S^+(X_p, X_c)$ Represents that $X_p \in pa(X_c)$ has a positive influence on X_c, which means that observing a higher ordered state for X_p does not decrease the likelihood of higher ordered states of X_c, regardless of the states of the other nodes $pa(X_c) \backslash X_p$.

$S^-(X_p, X_c)$ Represents that $X_p \in pa(X_c)$ has a negative influence on X_c, which means that observing a lower ordered state for X_p does not decrease the likelihood of lower ordered states of X_c, regardless of the states of the other nodes $pa(X_c) \backslash X_p$.

I Is a function of parent states that expresses the positiveness (or negativeness) of the joint influence of $pa(X_c)$ on X_c. The function can take on values in the range 0 to 1, i.e. $I \in (0,1)$ with $I(a_{neg}) = 0$ and $I(a_{pos}) = 1$. The method makes use of two types of influence factors, *individual influence factor* I_{ind} and *joint influence factor* I_{joint}. The two factors are determined as follows:

$$I_{ind}\left(x_{p_i}^{j}\right) = \begin{cases} \dfrac{\mathrm{rank}\left(x_{p_i}^{j}\right) - 1}{\mathrm{rank}\left(x_{p_i}^{max}\right) - 1}, & \text{if } S^{+}(X_p, X_c) \\ \dfrac{\mathrm{rank}\left(x_{p_i}^{max}\right) - \mathrm{rank}\left(x_{p_i}^{j}\right)}{\mathrm{rank}\left(x_{p_i}^{max}\right) - 1}, & \text{if } S^{-}(X_p, X_c) \end{cases}$$

$$I_{joint}(a) = \frac{\sum_{\left\{i:X_{p_i} \in pa(X_c)\right\}} I_{ind}\left(x_{p_i}^{j}\right) \cdot \left(\mathrm{rank}\left(x_{p_i}^{j}\right) - 1\right)}{\sum_{\left\{i:X_{p_i} \in pa(X_c)\right\}} \left(\mathrm{rank}\left(x_{p_i}^{max}\right) - 1\right)},$$

where the rank of the lowest ordered state is set to 1.

The EBBN method requires that the states of $pa(X_c)$ are ordered such that each of these nodes have either a positive or negative influence on X_c. Before the determination of a CPT can be done an expert must make the following assessments:

1. For each of the parent nodes $X_p \in pa(X_c)$, order the states of X_p such that X_p has either a negative or a positive influence on X_c.
2. For each of the states x_c of X_c:
   - Determine the assignment $pa(X_c) = a_{x_c}$ such that the probability $P(X_c = x_c | a_{x_c})$ is as large as possible.
   - Assess the probabilities $P(X_c | a_{x_c})$.
3. For each of the parent nodes $X_{p_i} \in pa(X_c)$, assess $P(X_c = x_c^{max} | a_{neg,p_i^+})$ and $P(X_c = x_c^{min} | a_{neg,p_i^+})$.

The determination of a CPT can be divided in to two steps. In the first step $P(X_c = x_c)$ is estimated as a function of I_{joint}, $f_{x_c}(I_{joint})$, for each state x_c of X_c. This is done by constructing a piecewise linear function $f_{x_c}: [0,1] \rightarrow [0,1]$ through the points $\left[I_{joint}(a_{x_c}), P(X_c = x_c | a_{x_c})\right]$. The estimated probabilities $P(X_c = x_c)$ for all x_c sum up to 1. In the second step the conditional probabilities $P(X_c | pa(X_c) = a)$ of the CPT are derived using

$$P(X_c = x_c^m | pa(X_c) = a) = \sum_{i:X_{p_i} | pa(X_c)} w_i \cdot \frac{\int_{I_{min}(i,j,a)}^{I_{max}(i,j,a)} f_{x_c^m}\left(I_{joint}(a)\right) dI_{joint}(a)}{I_{max}(i,j,a) - I_{min}(i,j,a)}$$

where

$I_{max}(i,j,a) = \max\left(I_{ind}\left(x_{p_i}^{j}\right), I_{joint}(a)\right)$,

$I_{min}(i,j,a) = \min\left(I_{ind}\left(x_{p_i}^{j}\right), I_{joint}(a)\right)$,

and w_i is the weight for each parent $X_{p_i} \in pa(X_c)$ calculated as

$$w_i = \frac{1}{2}\frac{\delta_i^+}{\sum_{n:X_{p_n} \in pa(X_c)} \delta_n^+} + \frac{1}{2}\frac{\delta_i^-}{\sum_{n:X_{p_n} \in pa(X_c)} \delta_n^-}$$

where

$$\delta_i^+ = P\left(X_c = x_c^{max} \middle| a_{neg,p_i^+}\right) - P\left(X_c = x_c^{max} \middle| a_{neg}\right),$$

$$\delta_i^- = P\left(X_c = x_c^{min} \middle| a_{neg}\right) - P\left(X_c = x_c^{min} \middle| a_{neg,p_i^+}\right).$$

The EBBN method only requires the expert to assign as many rows of the CPT as there are child states and one weight for each parent node. This method is based on linear interpolation and therefore it has difficulties to produce a large difference between two adjacent probabilities of a state in the CPT. According to [35], the EBBN method includes the expression *dominance*, which can be a characteristic of a parent node. Applying dominance comes down to, if the expert is certain of a probability, replacing a probability calculated by the method with the expert's belief. This can be done for all three methods and is therefore not discussed in this section.

Since the EBBN method only requires assignments of probabilities the method only needs two guidelines. First, order the states both for the child node and for the parent nodes. Second, if the child node has few parents, each having few states, consider letting the expert set the parent weights, w, direct instead of calculating them as in the method. This could also be done if the node has many parents and the expert knows how the parent nodes are weighted.

Assignments that were made when using the EBBN method on the node *Battery voltage* were:

$a_{x_c^{strong}}$ = [okay new] $a_{x_c^{weak}}$ = [faulty new] $a_{x_c^{dead}}$ = [faulty very old]
x_c^{max} = dead x_c^{min} = strong
a_{neg} = [okay new] a_{neg,p_1^+} = [faulty new] a_{neg,p_2^+} = [okay very old]

$$P\left(X_c \middle| a_{x_c^{strong}}\right) = [0.95 \;\; 0.04 \;\; 0.01]$$
$$P\left(X_c \middle| a_{x_c^{weak}}\right) = [0.008 \;\; 0.3 \;\; 0.692]$$
$$P\left(X_c \middle| a_{x_c^{dead}}\right) = [0.002 \;\; 0.1 \;\; 0.898]$$

$$\left.\begin{aligned} P\left(X_c = x_c^{max} \middle| a_{neg}\right) &= 0.01 \\ P\left(X_c = x_c^{min} \middle| a_{neg}\right) &= 0.95 \\ P\left(X_c = x_c^{max} \middle| a_{neg,p_1^+}\right) &= 0.692 \\ P\left(X_c = x_c^{min} \middle| a_{neg,p_1^+}\right) &= 0.008 \\ P\left(X_c = x_c^{max} \middle| a_{neg,p_2^+}\right) &= 0.1 \\ P\left(X_c = x_c^{min} \middle| a_{neg,p_2^+}\right) &= 0.6 \end{aligned}\right\}$$ Assignments required for the weights, w_1 and w_2.

This resulted in the CPT in Table 6.3, where the assigned rows of probabilities have replaced the generated, and the mean, m_{abs}, was 0.085.

Table 6.3 EBBN method used on the node *Battery Voltage*. Bold numbers indicate assigned probabilities.

		Battery voltage		
Charging system	Battery age	strong	weak	dead
okay	new	**0.950**	**0.040**	**0.010**
okay	old	0.600	0.134	0.266
okay	very old	0.197	0.198	0.605
faulty	new	**0.008**	**0.300**	**0.692**
faulty	old	0.004	0.160	0.836
faulty	very old	**0.002**	**0.100**	**0.898**

6.3.2.3 **Weighted sum algorithm**

This method consists of an algorithm that estimates the $k_1 \times \cdots \times k_N$ conditional probabilities, $P\left(X_c = x_c^m | X_{p_1} = x_{p_1}^{j_1}, X_{p_2} = x_{p_2}^{j_2}, \ldots, X_{p_N} = x_{p_N}^{j_N}\right)$, that populate a CPT. With X_c as the child node with l states and $\{X_{p_i}\}_{i=1}^{N}$ as the parent nodes with k_i states each the algorithm takes the following form

$$P\left(x_c^m | x_{p_1}^{j_1}, x_{p_2}^{j_2}, \ldots, x_{p_N}^{j_N}\right) = \sum_{i=1}^{N} w_i \, P\left(x_c^m | \left\{Comp\left(X_{p_i} = x_{p_i}^{j_i}\right)\right\}\right)$$

where $m = 1,2,\ldots,l$ and $j_i = 1,2,\ldots,k_i$. The method requires the expert to elicit two things;

1. the relative weights $w_1, \ldots, w_N$ for the parent nodes, where $0 \leq w_i \leq 1$ and $\sum_{i=1}^{N} w_i = 1$
2. the $k_1 + \cdots + k_N$ probability distributions, $P\left(x_c^m | \left\{Comp\left(X_{p_i} = x_{p_i}^{j_i}\right)\right\}\right)$, over X for compatible parental configurations.

Compatible parental configurations refer to the term $\left\{Comp\left(X_{p_i} = x_{p_i}^{j_i}\right)\right\}$ which has the following definition.

Definition 5.1. The state $X_{p_n} = x_{p_n}^{j_n}$, for the parent X_{p_n}, is compatible with the state $X_{p_i} = x_{p_i}^{j_i}$, if according to the expert's mental model the state $X_{p_n} = x_{p_n}^{j_n}$ is most likely to coexist with the state $X_{p_i} = x_{p_i}^{j_i}$. Then $\left\{Comp\left(X_{p_i} = x_{p_i}^{j_i}\right)\right\}$ denotes the compatible parental configuration where X_{p_i} is in the state $x_{p_i}^{j_i}$ and the rest of the parents are in states compatible with $X_{p_i} = x_{p_i}^{j_i}$.

Using the weighted sum algorithm will make the number of assessments of a CPT linear instead of exponential. [43]

The weighted sum algorithm is a simple elicitation method that doesn't need any specific guidelines before it is used. The expert only has to assign as many rows of the CPT as there are states in the parent nodes. However, the method is based on the concept of compatible parental configuration, which can be hard for an expert to determine.

The following assignments were made when using this method on the node *Battery Voltage*,

$w = [0.9 \;\; 0.1]$
$P\left(x_c^m | \{Comp(X_{p_1} = x_{p_1}^{okay})\}\right) = P\left(x_c^m | x_{p_1}^{okay}, x_{p_2}^{new}\right) = [0.95 \;\; 0.04 \;\; 0.01]$

$P\left(x_c^m|\{Comp\left(X_{p_1} = x_{p_1}^{faulty}\right)\}\right) = P\left(x_c^m|x_{p_1}^{faulty}, x_{p_2}^{very\ old}\right)$ = [0.002 0.1 0.898]
$P\left(x_c^m|\{Comp\left(X_{p_2} = x_{p_2}^{new}\right)\}\right) = P\left(x_c^m|x_{p_1}^{okay}, x_{p_2}^{new}\right)$ = [0.95 0.04 0.01]
$P\left(x_c^m|\{Comp\left(X_{p_2} = x_{p_2}^{old}\right)\}\right) = P\left(x_c^m|x_{p_1}^{faulty}, x_{p_2}^{old}\right)$ = [0.004 0.2 0.796]
$P\left(x_c^m|\{Comp\left(X_{p_2} = x_{p_2}^{very\ old}\right)\}\right) = P\left(x_c^m|x_{p_1}^{faulty}, x_{p_2}^{very\ old}\right)$ = [0.002 0.1 0.898].

The assignments for the weighted sum algorithm gave the CPT in Table 6.4, where the assigned rows of probabilities have replaced the generated, with a mean, m_{abs}, calculated to 0.062.

Table 6.4 Weighted sum algorithm used on the node *Battery Voltage*. Bold numbers indicate assigned probabilities.

		Battery voltage		
Charging system	Battery age	strong	weak	dead
okay	new	**0.950**	**0.040**	**0.010**
okay	old	0.855	0.056	0.089
okay	very old	0.855	0.046	0.099
faulty	new	0.097	0.094	0.809
faulty	old	**0.004**	**0.200**	**0.796**
faulty	very old	**0.002**	**0.100**	**0.898**

6.3.3 More examples of elicitation of a conditional probability table

The three methods were also used on two other nodes, both found in Netica; *Voltage at plug* from the network *Car Diagnosis 2*, see Table 6.5, and *Cardiac output* from the network *ALARM*, see Table 6.7 [40]. For the node *Voltage at plug* the methods were only used for the upper half of the CPT, i.e. when the parent node *Main fuse* is in state *okay*. This is done because the probabilities in the rows where *Main fuse* is in state *blown* are all 0 or 1 and therefore these distributions are considered to be known. The result from the three methods for each of the nodes can be seen in Table 6.6 and Table 6.8, where the assigned rows of probabilities have replaced the generated. The mean values, m_{abs}, for all three methods and nodes are presented in Table 6.9.

Table 6.5 The original CPT for the node *Voltage at plug* from the network *Car diagnosis 2*.

Main fuse	Distributer	Battery voltage	Voltage at plug: strong	weak	none
okay	okay	strong	0.90	0.05	0.05
okay	okay	weak	0.00	0.90	0.10
okay	okay	dead	0.00	0.00	1.00
okay	faulty	strong	0.10	0.10	0.80
okay	faulty	weak	0.00	0.10	0.90
okay	faulty	dead	0.00	0.00	1.00
blown	okay	strong	0.00	0.00	1.00
blown	okay	weak	0.00	0.00	1.00
blown	okay	dead	0.00	0.00	1.00
blown	faulty	strong	0.00	0.00	1.00
blown	faulty	weak	0.00	0.00	1.00
blown	faulty	dead	0.00	0.00	1.00

Table 6.6 The resulting CPTs from the three methods used on the node *Voltage at plug.* Bold numbers indicate assigned probabilities.

Likelihood			EBBN			Weighted sum		
strong	weak	none	strong	weak	none	strong	weak	none
0.992	0.008	0.000	**0.900**	**0.050**	**0.050**	**0.900**	**0.050**	**0.050**
0.100	0.800	0.100	**0.000**	**0.900**	**0.100**	**0.000**	**0.900**	**0.100**
0.000	0.008	0.992	0.169	0.422	0.409	0.450	0.025	0.525
0.003	0.201	0.796	0.319	0.430	0.251	0.450	0.025	0.525
0.000	0.000	1.000	0.000	0.375	0.625	0.000	0.450	0.550
0.000	0.000	1.000	**0.000**	**0.000**	**1.000**	**0.000**	**0.000**	**1.000**
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000
0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000

Table 6.7 The original CPT for the node Cardiac output from the network *ALARM.*

		Cardiac output		
Heart rate	Stroke volume	low	normal	high
low	low	0.98	0.01	0.01
low	normal	0.95	0.04	0.01
low	high	0.30	0.69	0.01
normal	low	0.95	0.04	0.01
normal	normal	0.04	0.95	0.01
normal	high	0.01	0.30	0.69
high	low	0.80	0.19	0.01
high	normal	0.01	0.04	0.95
high	high	0.01	0.01	0.98

Table 6.8 The resulting CPTs from the three methods used on the node *Cardiac output.* Bold numbers indicate assigned probabilities.

Likelihood			**EBBN**			**Weighted sum**		
Low	Normal	High	Low	Normal	High	Low	Normal	High
0.999	0.001	0.000	**0.980**	**0.010**	**0.010**	0.755	0.235	0.010
0.945	0.055	0.000	0.304	0.611	0.085	0.118	0.872	0.010
0.342	0.636	0.022	0.096	0.417	0.487	**0.300**	**0.690**	**0.010**
0.871	0.129	0.000	0.556	0.402	0.042	**0.950**	**0.040**	**0.010**
0.113	0.662	0.225	**0.040**	**0.950**	**0.010**	**0.040**	**0.950**	**0.010**
0.000	0.003	0.997	0.020	0.339	0.641	**0.010**	**0.300**	**0.690**
0.681	0.318	0.001	0.197	0.543	0.260	**0.800**	**0.190**	**0.010**
0.012	0.225	0.763	0.025	0.464	0.511	0.268	0.722	0.010
0.000	0.000	1.000	**0.010**	**0.010**	**0.980**	0.450	0.540	0.010

Table 6.9 The mean of the absolute difference between the node's original CPT and the method's generated CPT.

	Likelihood	**EBBN**	**Weighted sum**
Battery voltage	0.054	0.085	0.062
Voltage at plug	0.045	0.157	0.131
Cardiac output	0.082	0.193	0.220

6.3.4 Comparison of elicitation approaches

Of the three methods used for elicitation of a single probability the probability scale is considered the best method. This method helps with systemising the elicitation process and the scale can be designed according to the type of probabilities in the network. Although it may be difficult to design a suitable scale for some networks, the other two methods have even greater disadvantages; the gamble-like methods because they are hard to learn for an expert and the probability wheel because of its lack of eliciting high and low probabilities.

A general conclusion for the three methods that elicit full CPTs is that they generate more correct CPTs if the probabilities in each child state do not shift too much between high and low probabilities. Another finding for these methods is that an expert must check to see if there are any obvious errors in the generated CPTs and if there are, change these according to his opinion. This replacement also has to take place if the generated probabilities are not the same as the assigned probabilities. For small CPTs these methods might not shorten the elicitation procedure but they can still be useful if:

- the expert is uncertain of assessing specific probabilities and feels more confident in assessing weights for the child and parent states, then he could use the likelihood method.
- the expert is certain of some probabilities and uncertain of others, then the EBBN method or the weighted sum algorithm can be used.

A last conclusion is that the person that performs the elicitation, with the help of an expert, must be well versed in the methods he chooses to use in the elicitation process.

To determine which of the three methods that is best at generating full CPTs the mean of the absolute difference is compared and each generated probability is compared with the original probability. In Table 6.9 it can be seen that the likelihood method produced the lowest mean for all three nodes, which implies that it is the best one to use. Another advantage of the likelihood method is that it handles situations where probabilities in each child state shift between high and low probabilities better than the other two methods. The EBBN method's inability to handle these situations can clearly be seen when comparing the probabilities in Table 6.1 and Table 6.3. A comparison between the EBBN method and the weighted sum algorithm shows that the latter gives lower mean. This might be due to the fact that the expert sometimes has to assign more probabilities in the weighted sum algorithm and therefore more generated probabilities have been replaced resulting in a lower mean.

6.4 Sensitivity analysis

Sensitivity analysis is a technique to investigate how much a mathematical model's inputs, when varied, affects the model's output. In a BBN, the inputs can be seen either as states of observable nodes or as the probabilities in the conditional probability tables and the output is the probability of the state of interest. Consequently, one can distinguish between two different sensitivities:

- sensitivities oriented to findings, i.e. observations,
- sensitivities oriented to network parameters, i.e. the probabilities in the CPTs.

By studying these two kinds of sensitivities in a BBN the constructor of the network will be aware of which observable nodes and parameters that are critical for the network. The two sensitivity analyses are often done separately and when investigating the network's reliability both need to be considered. [44]

6.4.1 Sensitivity to findings

Based on the concept of d-separation, see Section 3.2, it can be determined by only looking at the structure of the network if an observable node, given evidence, has any influence on a node of interest. Together with an expert one can check if this information is consistent with his intuition of how the network should work.

If an observable node, X_o, is not d-separated from the node of interest, X_r, the influence can be quantified through *entropy reduction.* [4] Entropy, $H(X_r)$, is a measure of how scattered the probability mass is between the states in X_r and is calculated as

$$H(X_r) = -\sum_{x_r \in X_r} P(x_r)\log_2 P(x_r)$$

[45]. $H(X_r)$ is the entropy of X_r before any new findings and the effect of an observation in X_o can be measured through entropy reduction, I,

$$I = H(X_r) - H(X_r|X_o) = \sum_{x_r}\sum_{x_o} \frac{P(x_r, x_o)\log_2 P(x_r, x_o)}{P(x_r)P(x_o)},$$

where $H(X_r|X_o)$ is the entropy of X_r given new findings from X_o [46]. Entropy reduction can be calculated in Netica which also ranks the nodes based on their value of I. By performing this analysis it is possible to identify if a node is either too sensitive or insensitive to other nodes given different sets of observations. The result of the analysis could assure the constructor that the network's structure is correct or highlight errors in the network structure or CPTs. [4] It may also be worth considering the usefulness of nodes with less important inputs, even contemplating if they should be rejected from the BBN [44].

6.4.2 Sensitivity to parameters

Using experts' beliefs to define probabilities will inevitably result in inaccurate assessments and in a BBN these inaccuracies affect the reliability of the network's output. Sensitivity to parameters can be used to identify how uncertainties in parameters influence the output probabilities, i.e. the output robustness. [11]

If uncertainty in a parameter has large impact on the network this represents a probability where the accuracy is important. Consequently, rough estimates of all probabilities in the CPTs are sufficient as first assignments and sensitivity analysis will reveal probabilities that need to be assigned more accurate. [47]

Depending on the number of inputs varied, a sensitivity analysis to parameters can be either a one-way analysis or an n-way analysis. A one-way analysis implies that only one of the model inputs is varied whilst the others are fixed. In an n-way analysis the effect of varying n inputs is considered. [48]

When performing a one-way sensitivity analysis on the network's parameters one can study how a probability of interest (output value) is affected by varying a parameter. How sensitive an output is to uncertainties in a parameter depends on the certain case, i.e. current states of all observable nodes. [44] This means that in order to make a full scale sensitivity analysis for each output probability of interest; for every possible case the sensitivity is calculated for each parameter in the network. However, this straightforward sensitivity analysis is very time-consuming. One way to reduce the computations is to first ask experts in the field which parameters that are expected to be influential and focus primarily on these parameters. To

further limit the computations, for each specific case, only the nodes in the *sensitivity set*, Section 6.2.1, are studied. [9]

For each parameter belonging to a node in the sensitivity set the probability of interest is related as a quotient, called *sensitivity function*, of two linear functions to the parameter under study, Section 6.4.2.2[11]. By using the sensitivity function the influence a parameter has on a probability of interest can be quantified by calculating the *sensitivity value* and the *vertex proximity*, Section 0 and 6.4.2.4 [48].

6.4.2.1 **Sensitivity set**

A sensitivity set consists of those, and only those, nodes where variation in the parameters may influence the network's probability of interest. The definition of a sensitivity set is based on the concept of d-separation, see Section 3.2, and is stated below.

Definition 6.1. Let $B = (G, P)$ be a BBN. Let $X_r \in X(G)$ be the network's node of interest and let $O \subseteq X(G)$ be the set of observed nodes. Now, let G^* be the digraph that is constructed from G by adding an auxiliary predecessor Z_i to every node $X_i \in X(G)$. Then, the *sensitivity set* for X_r given O, denoted $Sen(X_r, O)$, is the set of all nodes X_i for which $\neg\langle \{Z_i\} \mid O \mid X_r \rangle_{G^*}^{d}$.

The definition of sensitivity set may need some clarification. The auxiliary predecessor Z_i is used to represent inaccuracy in X_i's assessment. If Z_i is not d-separated from the node of interest, X_r, then X_i belongs to the sensitivity set for that specific case of observed nodes. [11] Figure 6.4 shows an example when X_1 and X_2 don't belong to the sensitivity set for the node of interest, X_r, since their auxiliary predecessors Z_1 and Z_2 are d-separated from X_r.

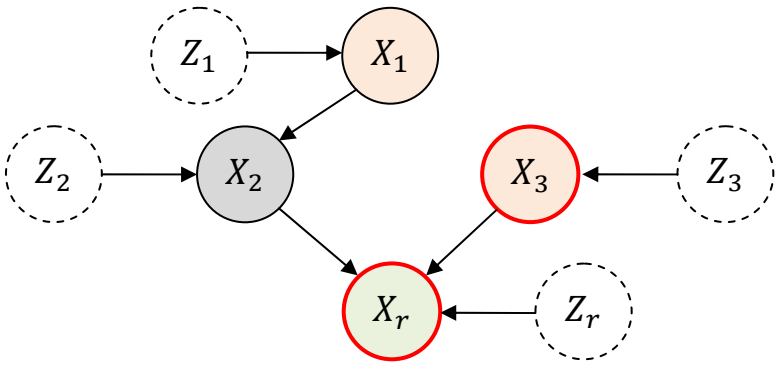


Figure 6.4 A BBN where X_2 is observed (grey), auxiliary predecessors are added to the network and X_r is the node of interest. A red circle represents a node that belongs to the sensitivity set.

6.4.2.2 **Sensitivity functions**

A sensitivity function provides a way of studying how variation in a parameter, y, affects a probability of interest, p. The parameter under study is $y = P(x^i|pa(X) = a)$, where x^i is a state of a node X and a is a combination of states of the parents of X. The probability of interest is $p = P(x_r|O)$, where x_r is a specific state of the node of interest and O is the set of observations.

When varying a parameter y the other parameters of the same distribution, $P(x^j|a), j \neq i$, must be co-varied so that the parameters sum to one. Keeping the sum to one is

accomplished by, if y with an original value of y_0 is varied to the value y_1, letting each of the other parameters $P(x^j|a)$ vary as follows

$$\hat{P}(x^j|a) = P(x^j|a) \cdot \frac{1-y_1}{1-y_0},$$

where $\hat{P}(x^j|a)$ are the co-varied parameters and this is called proportional co-variation.

If the parameters are proportionally co-varied the sensitivity function, $f_p(y)$, which describes how p varies with y takes the form

$$f_P(y) = \frac{c_1 y + c_2}{c_3 y + c_4} = \frac{d_1 y + d_2}{y + d_3}.$$

[48]

The sensitivity function reduces to a *linear* function, i.e. $c_3 = 0$, if the parameter under study belongs to a node that is an ancestor of the node of interest and the parameter's node has no observed descendants. Also if there are no observations; the probability of interest is linearly related to all parameters in the sensitivity set. The constants in a linear sensitivity function can be determined by computing the probability of interest for two different values of the parameter y and then solving a simple linear equation system.

If the sensitivity function is not linear it can be observed as a fragment of a *rectangular hyperbola*, see Figure 6.5. The constants in a hyperbolic sensitivity function can be determined by computing the probability of interest for three different values of y. A rectangular hyperbola takes the general form

$$f(y) = \frac{r}{y-s} + t$$

where for a hyperbolic sensitivity function, the constants are

$$s = -d_3,\ t = d_1 \text{ and } r = d_2 + s \cdot t.$$

[11]

6.4.2.3 **Sensitivity value**

The derivative of the sensitivity function can be used as a measure of how infinitesimally small shifts in the parameter under study affects the probability of interest. This measure is called the *sensitivity value* of y and p and it is defined as the absolute value of the first derivative of the sensitivity function at the original value, y_0, of y.

$$SV = |f'(y_0)| = \frac{d_1 d_3 - d_2}{(y_0 + d_3)^2}$$

is the sensitivity value of a hyperbolic sensitivity function and

$$SV = |f'(y_0)| = \frac{c_1}{c_4}$$

is the sensitivity value of a linear sensitivity function. According to the literature; parameters with a sensitivity value larger than one need further attention and the accuracy of these parameters are of importance for the network [49].

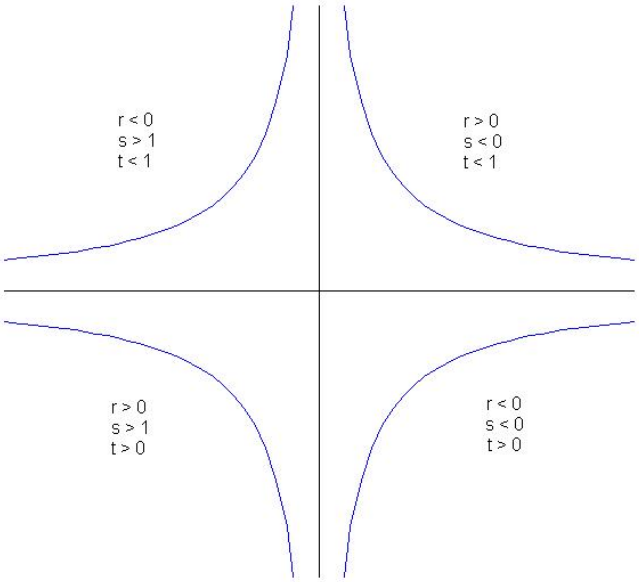


Figure 6.5 A rectangular hyperbola where in each quadrant a sensitivity function is illustrated.

When studying a probability of interest the sensitivity value can be calculated for all the parameters in the sensitivity set for all possible cases. However, this can be very time consuming and thus an upper bound on the sensitivity value can be used to reduce the number of computations. The upper bound for y_0 and the original value of the output probability, p_0, is

$$|f'(y_0)| \leq \frac{p_0(1-p_0)}{y_0(1-y_0)}$$

and it is based on the so called bounding functions, see [48] for more details. In Figure 6.6 the upper bound as a function of y_0 and p_0 is plotted. Figure 6.6 can be used for identification of which combinations of y_0 and p_0 that could result in high sensitivity values. Hence, it is also possible to identify parameters that will have small sensitivity values for all cases, i.e. all possible values of p_0. This means that it is not necessary to calculate the actual sensitivity value of these parameters and thus the number of computations is reduced. [48]

In Figure 6.7 the upper bound as a function of y_0 and p_0 can be seen from above and the black parts indicate functional values lower than 1.01. This plot implies that parameters close to 0.5 will have a maximum sensitivity value smaller than 1.01 for all possible output probabilities p_0. The black shape in the in Figure 6.7 would be even more narrow for a lower limit than 1.01 and a limit lower than one would induce a gap in the middle of the black shape.

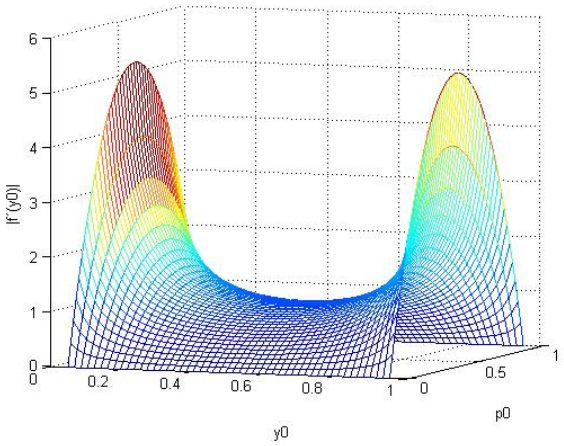


Figure 6.6 The upper bound as a function of x_0 and p_0 plotted for $0.05 < y_0 < 0.95$.

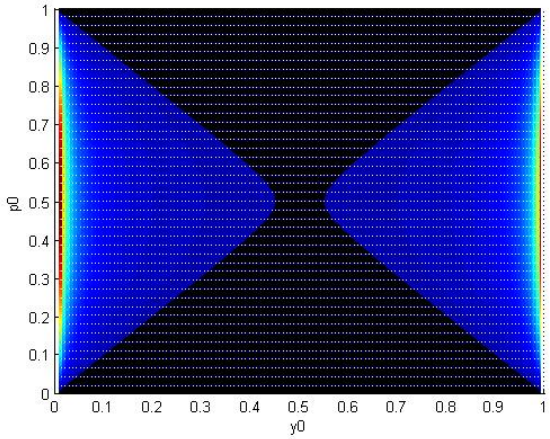


Figure 6.7 The upper bound as a function of y_0 and p_0 seen from above. The black parts indicate functional values lower than 1.01.

If the sensitivity function is linear, the maximum sensitivity value is one [48]. Worth noting is that for a linear sensitivity function, if $SV = 1$, it directly reflects the uncertainty of the parameter in the output. This means that independently of the size of the shift of a parameter the output probability is shifted the same amount, i.e. if the parameter is shifted 0.2 so is the output. For a linear function, a sensitivity value equal to one also implies that either $y_0 = p_0$ or $y_0 = 1 - p_0$.

6.4.2.4 **Vertex proximity**

The assessed probabilities in a CPT may be very uncertain and to study only infinitesimally small shifts in the parameters is then not enough. If the relation between the output and the parameter is a linear function the sensitivity value remains the same for larger parameter shifts. But if the sensitivity function is hyperbolic the sensitivity value could change significantly for larger shifts. Thus, another measure, which looks at larger shifts of parameters, is needed to quantify the sensitivity of the output probabilities. A technique for this is to find the y for which the sensitivity value is equal to one and use that as a measure of the shift from large sensitivity values to small ones and vice versa. The point is called the *vertex* of the hyperbola branch under study and can easily be computed from the constants of the sensitivity function using

$$y_v = \begin{cases} s + \sqrt{|r|}, if\ s < 0 \\ s - \sqrt{|r|}, if\ s > 1 \end{cases}.$$

The original value of the parameter, y_0, and y_v can be compared and the conclusion that can be made depends on the type of hyperbolic function that the sensitivity function represents.

In Figure 6.5 one can see that the absolute value of the derivative of the functions in the second- and third-quadrant, i.e. s >1, is increasing. Which means that if y_0 is smaller than y_v but quite close to it, it can be indicative of possibly significant effects of variation of the parameter to larger values. With the same reasoning, for functions where $s < 0$, and if y_0 is larger than y_v but quite close to it, it can be an indication that, if the parameter is shifted to a smaller value, it affects the output probability significantly. [48]

Further, it is not interesting to calculate the vertex if the sensitivity value is larger than one, since the output is already considered to be sensitive to the value of the parameter.

6.4.3 An illustrative example of sensitivity analysis

A quite thorough sensitivity analysis is done on a BBN, available through Netica [40], called *Car Diagnosis 2* and the network can be seen in Figure 6.8. This example is meant to show how a sensitivity analysis can be done and the usefulness of the analysis. First, rough estimates are assigned in the CPTs. Throughout the sensitivity analysis some of the estimates are replaced by the probabilities given by the original network, which are assumed to be the correct probabilities. After a replacement, the parameter is excluded from the rest of the sensitivity analysis. This procedure, showing how a sensitivity analysis works, is inspired by Coupé et al [47].

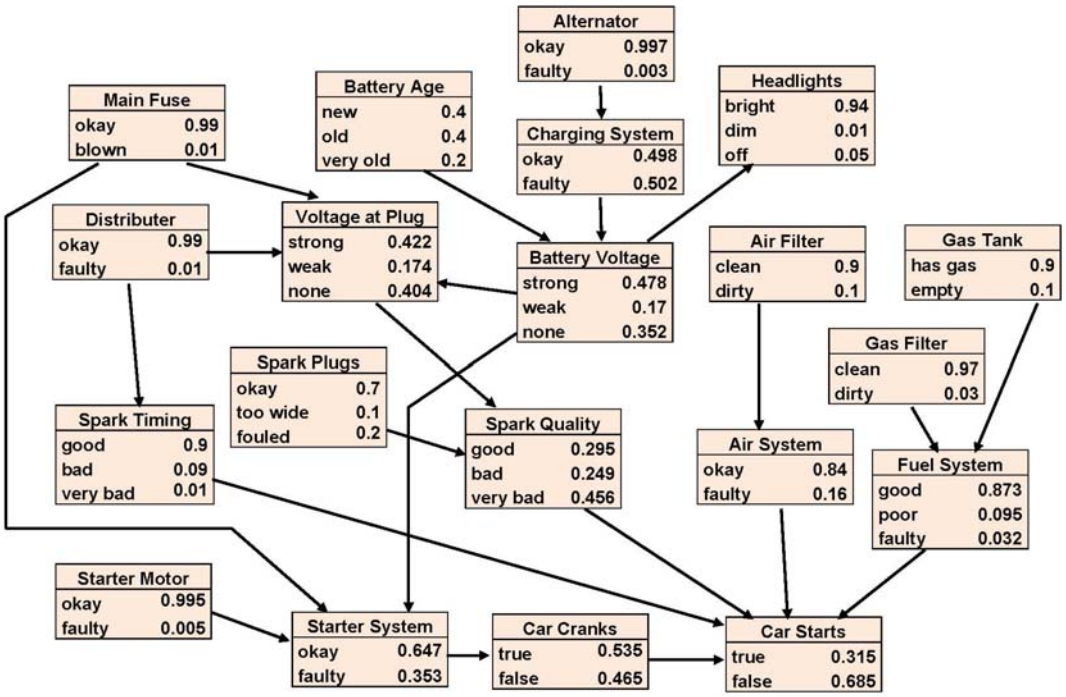


Figure 6.8 The network *Car Diagnosis 2.*

6.4.3.1 **Problem set up**

The likelihood method is considered to be the best method of the ones studied in Section 6.3.2 and it is used to generate rough estimates of the parameters in the two nodes *Voltage at Plug* and *Battery Voltage.* For the rest of the nodes, a random number is generated between 0 and 0.2 for each row in each original CPT. The first parameter in a row is then changed as much as the generated number and the parameter is increased if the original value is below 0.5 and decreased otherwise. Remaining parameters in the row are decreased respectively increased uniformly.

If the original parameters in a row are all either 0 or 1, this is considered to represent a situation when the expert is certain of a probability distribution. These probabilities are kept and the row is excluded from the sensitivity analysis. For the same reason all parameters with a value of 0 are kept and excluded from the analysis.

The state probabilities in three nodes; *Car Starts*, *Starter System* and *Voltage at Plug*, are used as output of interest for this sensitivity analysis. *Car Starts* is chosen on the basis that it's the final output of the network and the main interest. The other two nodes are chosen since they both connect different parts of the network. The selection of the nodes has been inspired by the nodes of interest in the BBN developed within RASTEP.

6.4.3.2 **Sensitivity to findings**

Sensitivity to findings is done in Netica and the result, for all three nodes of interest, using no observations is that the *entropy reduction* is equal to zero for the nodes *Headlights* and *Gas Tank*. Studying the CPTs of those nodes and the CPT of the node *Fuel System* confirms that *Headlights* and *Gas Tank* have no impact on the rest of the network. Therefore these two nodes are removed from the network. If an expert is at hand when performing sensitivity to findings a more thorough analysis can be done.

6.4.3.3 **Sensitivity to parameters**

At the beginning of this analysis the network consists of 57 rows with estimated parameters making it a total of 127 parameters. If a row has two parameters they will result in the same sensitivity value since the parameters will be varied with the same amount and therefore only half of the parameters in the binary nodes are studied. There is no symmetry in the sensitivity values for nodes with more than two states and all parameters in these nodes need to be included in the sensitivity analysis. This results in a sensitivity analysis where 80 parameters are studied.

The analysis is done in three stages, one for each node of interest, and in each stage five cases with three observed states and one case with no observations are studied. The cases are both randomly generated and chosen with the aim to produce interesting types of sensitivity sets. The node *Spark quality* is excluded from the analysis since it is a deterministic node.

For each case with observed nodes the number of studied probabilities is reduced to the number of parameters in the nodes in the sensitivity set. The number of parameters is further reduced if a node in the sensitivity set has an observed parent since only the parameters belonging to a parental combination with the observed state need to be studied.

In a first stage of sensitivity analysis to parameters the probability of the state *true* in node *Car Starts* is used as an output probability of interest. For each case the sensitivity set is found and the sensitivity values, SV, and vertexes, x_v, are computed. After all six cases have been analysed the maximum sensitivity value of each parameter is found. Very few sensitivity values are larger than one and therefore it is decided that all parameters with $SV \geq 0.3$ should be replaced with the original probabilities. Since the sum of a row in a CPT must

sum to one the entire row of a parameter with $SV \geq 0.3$ must be replaced and the total number of changed parameters is 13, see Table 6.10.

Table 6.10 Result of the first stage of sensitivity analysis to parameters.

Number of parameters with $SV \geq 0.3$	7
Number of rows changed	6
Total number of changed parameters	13

In the second stage the state *okay* in *Starter system* was used as output probability of interest. Sensitivity values larger than 0.3 are again considered to represent parameters that should be replaced and another 14 parameters are changed, see Table 6.11.

Table 6.11 Result of the second stage of sensitivity analysis to parameters.

Number of parameters with $SV \geq 0.3$	8
Number of rows changed	6
Total number of changed parameters	14

In the third stage all the three states in the node *Voltage at plug* are used as output probabilities and studied at the same time. Demonstrations of two different cases, Case 1 and Case 2, can be seen in Figure 6.9 and Figure 6.10. In the figures the node of interest, the observations and the sensitivity sets are shown in different colours. The sensitivity function with the output probability $p = P(\text{Voltage at plug} = \text{strong})$ as a function of the parameter
$y = P(\text{Voltage at plug} = \text{strong} \mid \text{Main fuse} = \text{okay}, \text{Distributer} = \text{okay}, \text{Battery voltage} = \text{strong})$ is plotted for both cases in Figure 6.11. In Case 1 the sensitivity function is linear since the node *Voltage at plug* has no observed descendants. Since *Car starts* is observed in Case 2 the sensitivity function for the same output and parameter becomes hyperbolic.

For hyperbolic sensitivity functions the vertexes are calculated in order to study what happens if larger parameter shifts are made. Throughout the sensitivity analysis of this network only a handful calculated vertexes are in the range $0 \leq y_v \leq 1$ which are the only ones of interest since a parameter only can take values in that range. In Case 2 when the parameter $y = P(\text{Distributer} = \text{okay})$ is studied together with the output probability $p = P(\text{Voltage at plug} = \text{none})$ the vertex is $y_v = 0.88$. In Figure 6.12 the hyperbolic sensitivity function can be seen where the red dot is $y_0 = 0.81$ and the green line represents the vertex. In this plot it can be seen that if the original value of the parameter would increase the sensitivity value, i.e. the derivative of the sensitivity function, would be larger and for a parameter value larger than the vertex the sensitivity value becomes larger than 1. The conclusion is that, although SV = 0.85 for y_0, the sensitivity value for this parameter and output probability should be considered to be larger than 1.

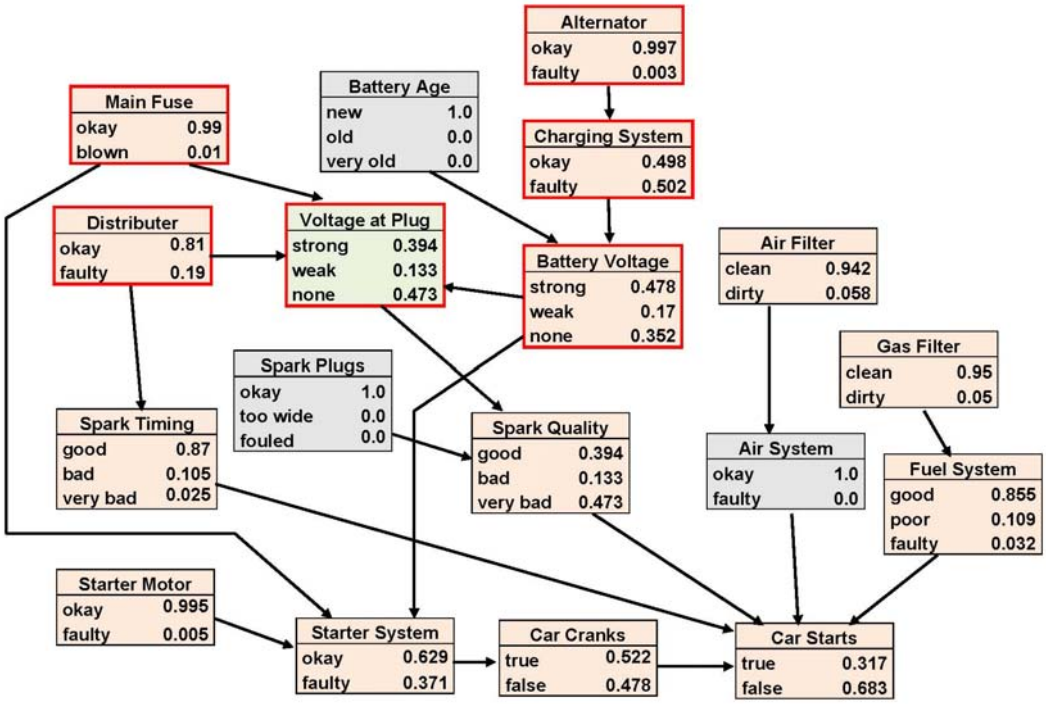


Figure 6.9 The network *Car Diagnosis 2* - Case 1. Green represents the node of interest, grey represents an observed node and a red border indicates a node that belongs to the sensitivity set.

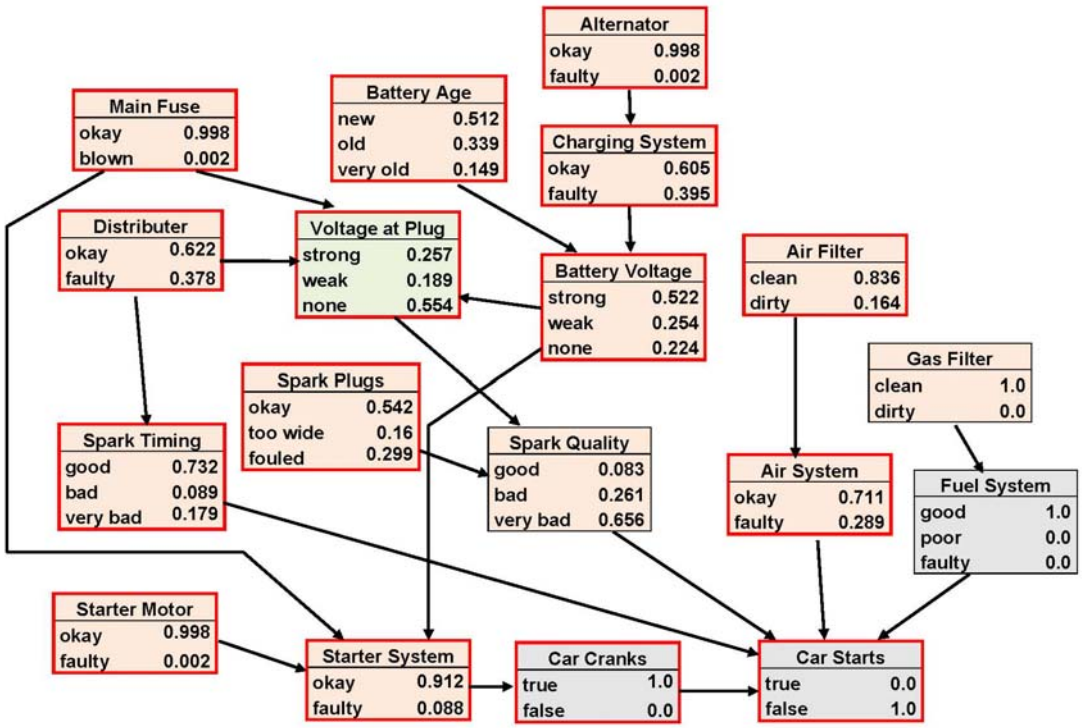


Figure 6.10 The network *Car Diagnosis 2* - Case 2. Green represents the node of interest, grey represents an observed node and a red border indicates a node that belongs to the sensitivity set.

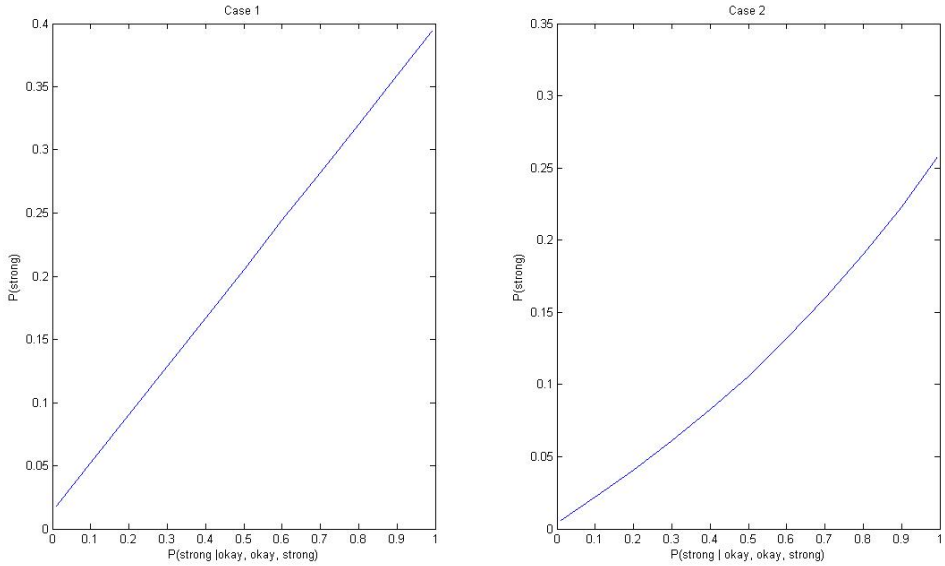


Figure 6.11 The sensitivity function with the output probability $p = P(\text{Voltage at plug} = \text{strong})$ as a function of the parameter $y = P(\text{Voltage at plug} = \text{strong} \mid \text{Main fuse} = \text{okay}, \text{Distributer} = \text{okay}, \text{Battery voltage} = \text{strong})$. Case 1 to the left and Case 2 to the right.

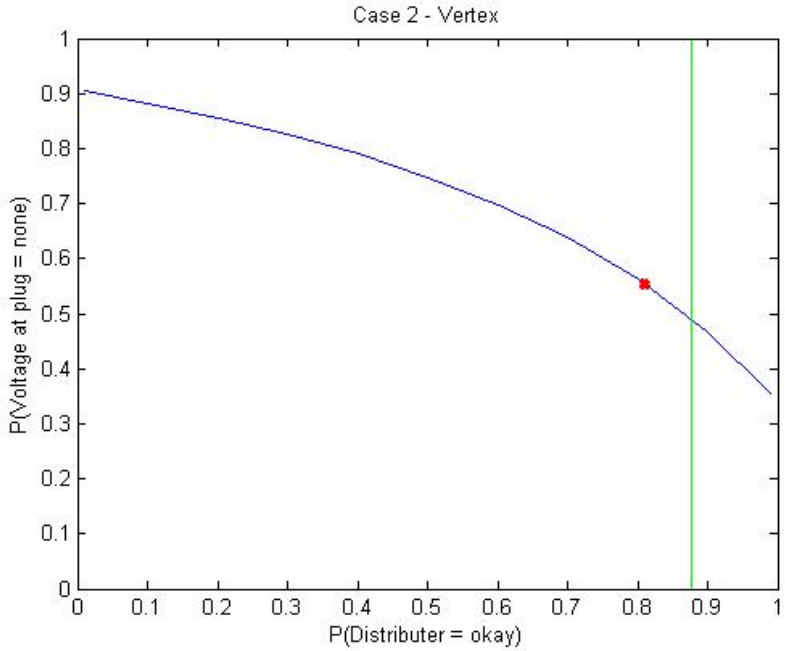


Figure 6.12 The sensitivity function with the output probability $p = P(\text{Voltage at plug} = \text{none})$ as a function of the parameter $y = P(\text{Distributer} = \text{okay})$. Red dot is $y_0 = 0.81$ and the green line represents the vertex $y_v = 0.88$.

The resulting changes after sensitivity analysis with respect to the output probabilities in the node *Voltage at plug* can be seen in Table 6.12. It should be noted that the maximum sensitivity value of all output probabilities is considered.

Table 6.12 Result of the third stage of sensitivity analysis to parameters.

Number of parameters with $SV \geq 0.3$	10
Number of rows changed	6
Total number of changed parameters	14

Table 6.13 shows the resulting output probabilities after each stage. The results after stage three are compared with the original values and one can see that there is still room for improvement. Therefore a fourth stage of sensitivity analysis is carried out with *Car Starts* as node of interest since it is considered to be the most interesting node in this network. Three cases are studied in this stage and as expected the sensitivity values are relatively lower in this stage compared to the other stages. The reason for this is that many parameters have already been changed and are not included in the further analysis. Although only the six parameters with a $SV \geq 0.1$ are replaced the resulting output probabilities are closer to the original, see the result of stage three and four in Table 6.14.

Table 6.13 Result of the fourth stage of sensitivity analysis to parameters.

Number of parameters with $SV \geq 0.1$	6
Number of rows changed	6
Total number of changed parameters	14

A total of 24 rows with estimated parameters are replaced with the original probabilities in the sensitivity analysis. This means that even though more than half of the rows have rough estimates of the parameters the output probabilities are fairly close to the original.

Table 6.14 Output probabilities of the nodes of interest with different amounts of estimated parameters. Values in parenthesis indicates the number of estimated rows in the CPTs of the network.

Output probability	Only rough estimates (57)	Result of stage 1 (51)	Result of stage 2 (45)	Result of stage 3 (39)	Result of stage 4 (33)	**Original (0)**
$P(\text{Car Starts} = \text{True})$	0.168	0.192	0.229	0.265	0.298	**0.315**
$P(\text{Starter system} = \text{okay})$	0.490	0.491	0.580	0.580	0.607	**0.647**
$P(\text{Voltage at plug} = \text{strong})$	0.415	0.358	0.368	0.410	0.420	**0.422**

6.4.3.4 **Some comments**

Parameters with a sensitivity value larger than 1 need further attention. In the sensitivity analysis of the network *Car Diagnosis 2* very few sensitivity values were larger than 1 and therefore a lower limit was used to identify parameters that had significant impact on the output. Another way of determining the parameters that need further attention is to simply replace a specific number of probabilities in each stage. It is difficult to give an exact limit or a specific number since it depends on the network under study and the magnitude of sensitivity values generated. Regardless of the method to determine the parameters whose accuracy is important, the total number of re-estimated parameters comes down to weigh the time and money to be invested against the benefits of higher accuracy [9].

6.5 Verification and validation

Before a BBN can be used its ability to model a system and give a reliable output must be evaluated. Errors in some parts of the network can be identified both when the structure is reviewed and during the sensitivity analysis. However, the network should be further verified and validated, preferably with the help of several experts in the field.

The network can be verified by testing if the BBN predictions are in line with results from other available sources of information [50]. Experts can help to verify if the behaviour of the network is as expected when different observations are given as inputs to the network.

During sensitivity analysis parameters with high influence on the network are identified and the probabilities should be more accurately estimated. If for some of these parameters it is only possible to give rough estimates it should be evaluated if it is accepted or if the structure of the network needs to be revised.

6.6 Proposed method for robust determination of CPT:s

The proposed method for including experts' beliefs in the CPTs of a BBN can be seen in Figure 6.13. The aim has been to create a method that results in a reliable network even though expert beliefs, which are uncertain sources, have been used. First the structure of the network should be studied to see if it can be modified in order to make the assessments of the probabilities easier. Estimation of probabilities can be divided into three groups and regardless of which method that is used, rough estimates of the probabilities are sufficient as a first assignment. Rough estimates are good enough since running a sensitivity analysis on the network will reveal which parameters have large affect on output probabilities of interest and thus need to be more accurately estimated. Sensitivity analysis may also lead to modification of the network's structure. Verification and validation of the network is the last part of the method. The four parts of the method are performed iteratively until the network is robust and reliable.

6.6.1 Network structure

After the structure of a BBN has been constructed it can be studied if some modifications can be made to ease the assignment of probabilities. The network can be modified through the technique of node divorcing, which implies that a node is introduced between a child node and some of its parents, resulting in fewer probabilities to assign.

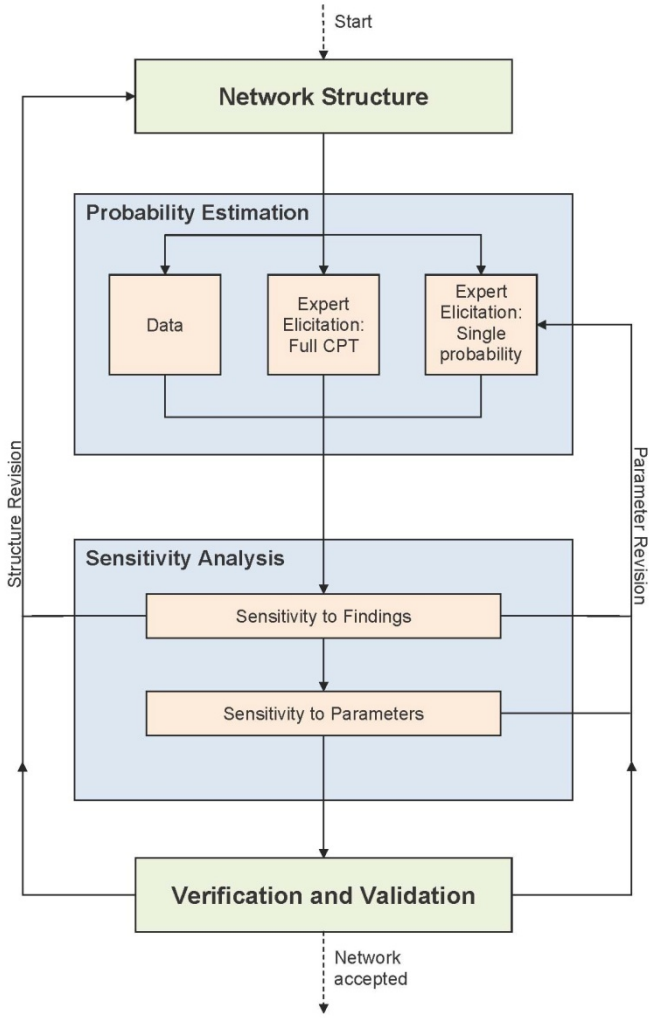


Figure 6.13 Proposed method.

6.6.2 Probability estimation

In the second part of the method probabilities in the network's CPTs are estimated. These assignments can be done in three different ways, as is shown in Figure 6.13. The probabilities can be estimated from data or if there are no known data experts' beliefs are used to assess the probabilities.

Assessing probabilities with the help of experts can be done through elicitation and for the proposed method there are two kinds of elicitation procedures; elicitation of a single probability and elicitation of a full CPT. Procedures for elicitation of a single probability are foremost ways of systemising the assessment of probabilities and helping the expert to express his beliefs in probabilities. Single probabilities can be elicited either by letting the expert assign the probabilities directly or by an indirect method where the expert make a decision from which his belief is inferred, see Section 5.1.

Elicitation methods for generating a full CPT are the other kind of elicitation procedures and their purpose is to use fewer assignments to assess a full CPT, thus making it more time efficient. Three different elicitation methods that are not restricted to binary nodes are likelihood method, EBBN method and weighted sum algorithm and have been discussed in Section 5.2. These elicitation methods require different types of assignments. The likelihood method requires the expert to assess weights for the states in the child node of interest and

the states of its parent nodes based on a typical probability distribution of the child states. The other two methods require the expert to assess probabilities for some of the rows in the CPT and weights for the parent nodes. The elicitation methods of a full CPT can be used if the expert feels more confident determining these types of different assignments instead of single probabilities although the time of the elicitation procedure won't be significantly reduced.

The choice of elicitation method depends on the structure of the CPT, e.g. how many parent states, and on the expert's ability to assign probabilities or weights. Guidelines for when and how to use each method is found in Section 5.2.1-5.2.3.

6.6.3 Sensitivity analysis

In this part of the proposed method a sensitivity analysis is performed. This is a technique to study how states of observable nodes and probabilities in CPTs affect the probability of a state of interest. The analysis can be distinguished between two different sensitivities; sensitivity to findings and sensitivity to parameters.

Sensitivity to findings is based on the concept of d-separation, see Section 2.3.1, and can determine if a node, given an observation, has influence on a node of interest. If it has, entropy reduction can be used to quantify the influence. The result of this analysis may require the structure of the network to be reconsidered and imply that single probabilities should be re-estimated, see Section 6.1.

Sensitivity to parameters is used to identify how uncertainties in the parameters, i.e. the probabilities in a CPT, affect an output probability. The influence a parameter has on an output probability can be quantified by using the sensitivity function to calculate the sensitivity value, see Section 6.2. A high sensitivity value indicates that the parameter has a high influence on the output probability and thus it is of importance that the parameter is assessed with high accuracy. The demand of high certainty in the parameters may lead to re-estimation of some parameters after sensitivity to parameters has been performed. After re-estimating parameters a new sensitivity analysis should be performed to see if there are other parameters that result in high sensitivity values. The determination of when the parameters are sufficiently accurate is a weigh between the time and money to be invested and the benefits of higher accuracy.

By studying these two kinds of sensitivities in a BBN the constructor of the network will be aware of which observable nodes and parameters are critical for the network. The two sensitivity analyses are often done separately but when investigating the network's reliability both should be considered.

6.6.4 Verification and validation

Before a BBN can be used its ability to model a system and give a reliable output must be evaluated. The network can be verified either by comparison with data or by the help of experts. If the result is not satisfying a revision of either the network structure or the probability estimation may be required.

6.7 Discussion

The iterative method that is proposed has the ability to produce a relevant and defendable set of conditional probabilities in a BBN even though expert judgement is included in the assessments.

The use of the elicitation methods described, either elicitation of single probabilities or of full CPTs, gives a systematic way to include expert beliefs in the network. If the expert is uncomfortable in assigning specific probabilities a method that generates a full CPT is recommended. The advantage is that it better reflects the expert's knowledge of the causalities in the network and therefore is more defendable to use. A drawback with the full CPT methods is that the complexity of the methods requires the constructor to be well versed in their implementation.

The gain of performing sensitivity analysis, although it may be time consuming, is that in a large part of the parameters rough estimates are acceptable to use. Since sensitivity analysis puts focus on the most essential parts of the network, where accuracy in the probabilities is important, the resulting BBN is reliable although rough estimates are included. If the sensitivity analysis indicates that parameters, for which only rough estimates are possible to assess, have high sensitivity values the constructor has to consider the required reliability of the network. The constructor may accept these rough estimates and otherwise the structure of the network has to be revised.

It has been a challenge to reach a conclusion of which limit for the sensitivity values that determines the parameters that needs to be accurately assessed. Parameters with sensitivity values larger than one have to be re-estimated if possible. However, it depends on the network under study if parameters with lower sensitivity values also should be considered as probabilities with high impact on the network. The number of parameters and the range of the sensitivity values in the network are of great importance in this decision.

The benefit of following this method is that along the way errors in the network can be detected and corrected and in that way prevent the use of an incorrect network. At the same time it can be confirmed that the network is correct and strengthen the reliability of the network.

For the O3 network it is concluded that the elicitation methods for a full CPT may be applied and a probability scale suitable for the O3 network is proposed. Performing sensitivity analysis on the O3 network gives information about which parameters that influence the prediction of source terms and also the ability of the model to predict initiating events. Since only a small percentage of the parameters in O3 have sensitivity values larger than one it seems reasonable to consider the accuracy of more parameters. Further it is concluded that only two of the most important output nodes are needed to be studied. This since performing sensitivity analysis on the other output nodes doesn't give any new valuable information.

When investigating a network's reliability both sensitivity to findings and to parameters need to be considered. Sensitive to findings have not been performed on the O3 network and it needs to be done to reveal how observations in the observable nodes affect the network's output. In the short sensitivity to parameters analysis performed, 10% of the parameters in the O3 network were identified as parameters where the accuracy is important. However, it is recommended to continue the sensitivity analysis of the network with more cases, which may reveal more parameters with significant impact on the network.

For the BBN in RASTEP it should be possible to implement interfaces for the probability estimation part. For the probability scale the interface should allow the expert to make his marks on the scale and then calculate the probabilities. For the elicitation methods of a full CPT an interface lets the expert give his assignments as inputs and it quickly generates the CPT so that the expert can confirm the result.

In the real case scenario the user of the BBN in RASTEP may be uncertain about the observables that are used as inputs to the network. This situation might call for an interface that enables the user to receive information about the effect of the input that he is about to give the network. Another suggestion is to let the user assign a probability for an observation, i.e. his certainty for this observation being true.

It would be valuable to connect an interface to Netica in order to simplify the sensitivity analysis. Given the output of interest the interface would have the ability to calculate the sensitivity values for all parameters in the sensitivity set.

7. PARTICIPATION IN INITIATION OF OECD/NEA WGAMA TASK ON COMPARISON OF FAST-RUNNING TOOLS FOR SOURCE TERM DETERMINATION

7.1 Background and outline of WGAMA project

During 2012, the Working Group Accident Management (WGAMA) of the OECD/NEA initiated an international benchmarking project on fast-running software tools used to model fission product releases during accidents at nuclear power plants [51].

The objective of the WGAMA activity is to benchmark software tools used to estimate accidental radioactive material releases inside and outside of the containment boundary during accident conditions or emergencies in nuclear facilities such as power reactors, research reactors, fuel reprocessing facilities, etc. The benchmarking is intended to identify the strengths and weaknesses of the tools used for source term prediction and identify the knowledge gaps, to propose improvements to modelling capabilities. The proposed activity and the follow-up are expected to augment the predictive capability of national regulatory authorities to rapidly respond to short-term protective measures effectively to nuclear emergencies.

The software tools evaluated within the WGAMA project will be assessed based on their ability to meet the following criteria:

- Estimate the fission product source terms and provide an estimate of core damage state and the condition of the physical barrier
- Predict doses resulting from fission product releases
- Capability to run with small number of input parameters (at the start of a nuclear accident only limited information will be available for use)
- Incorporate additional details as more information becomes available and improve the predicted results
- Versatility in dealing with different reactor technologies
- Speed of calculation
- Accuracy and confidence in the results
- Output the results in a clear, user-friendly and logical manner that can be useful in taking necessary actions

The benchmarking study aims to assess and improve the current state of knowledge regarding fast-running modelling software, as well as to assess and improve the software. It will result in increased understanding of the available tools and will identify gaps in their performance. This will give users a better understanding on how much one can rely on the results of the simulations in an emergency and what modelling capabilities can be improved to enhance safety and emergency response.

7.2 NKS project contribution to the WGAMA project

This project was initiated after the NKS project had been started. Participation was discussed with the NKS programme managers, and it was agreed that although the OECD/NEA project will carry on after the finalisation of the NKS project, it would still be beneficial to include RASTEP in the initiation of the WGAMA project.

During 2013, the following has been achieved:

- Contribution to WGAMA project start-up by reviewing and commenting project questionnaire
- Addition of Asea-Atom BWR 75 to the standard designs to be covered within the project
- Filling in of project questionnaire, and performance of first (tentative) evaluation of analysis cases.
- Presentation of the RASTEP approach and of the preliminary evaluation results at the second project meeting in October 2013.
- RASTEP will be applied to three scenarios, i.e.:
    - Case 1 Oskarshamn 3 (ASEA-Atom BWR); RASTEP model exists
      Scenario: Scram due to transient TSxD (turbine trip with dump blocking) with complete loss of RHR and severe accident management systems operating
    - Case 2 Peach Bottom (GE BWR, Mark I); to be based on revised model for Oskarshamn 2 BWR
      Scenario: Unmitigated long-term station blackout
    - Case 3 Surry (W 3-loop PWR); to be based on revised model for Ringhals 3 PWR (W, 3-loop)
      Scenario: Unmitigated long-term station blackout

The work performed so far has shown that RASTEP is well suited for inclusion in the WGAMA project. RASTEP specifically addresses several of the criteria defined for the project. It is expected that participation in the project will contribute both to the WGAMA project by demonstrating an innovative approach to source term prediction, and to RASTEP as a project by providing a benchmark with several alternative softwares and approaches and by providing a possibility for a certain degree of verification and validation of analysis results.

8. CONCLUSIONS

8.1 Tools for accident progression analysis and source term determination within RASTEP

The use of RASTEP in its current configuration (i.e., without any integration with a fast running deterministic code) has demonstrated that the predicted most probable plant statuses usually are quite accurate in predicting the potential future accidental scenarios. This is mainly due to the strength of the PSA modelling, the accurate mapping of the key plant parameters, and the systems interrelations modelled in the BBN. However, the use of a limited set of predefined source terms makes the current approach not realistic enough, and reduces flexibility in the use of the tool.

Using a fast running deterministic code such as MARS could provide great advantages in performing on-the-fly calculations. Furthermore, the same on-line plant data necessary for the probabilistic (BBN) module can be used with the MARS software.

The fact that MARS is based on MAAP and all Swedish NPPs already use MAAP models for deterministic analyses, makes this approach even more attractive. The requirements in terms of costs and time for the implementation of such a system is however uncertain. According to the CSN experience in Spain, more resources were used in creating the MAAP models at the nuclear sites than for the validation and tests of the MARS system itself.

As described in section 4.5.2, understanding which approach is favourable for generating the input for MARS is still a big challenge. The easiest way would be feeding the deterministic software directly with live plant data. Moreover, the use of the BBN to infer the most likely plant state is a very powerful feature and, even in the case systems for on-line plant data transmission are implemented, it could still be very beneficial to use the probabilistic (BBN) module to feed MARS. The boundary conditions related to the accidental scenario could be assessed rapidly in an early phase of the accident.

However, due to the difficulties and unanswered questions regarding MARS, the more realistic choice for now is judged to be expanding today's library of pre-calculated source terms to achieve higher scenario resolution and better precision, i.e. more realistic data, compared to the present, simplified data. In line with this objective, and to guarantee high quality and validity, the use of a variety of deterministic codes is assumed to be valuable.

Being a tool for real-time analysis, RASTEP will require this source term library to be highly flexible and operable. Sequences entirely or partly generated by several different tools, to be used in series or in parallel, will require a good deal of know-how and skills on how to co- and cross-interpret results. A study on the possible interplay and compatibility of different codes, i.e. MAAP and MALCOR, has been initiated; the preliminary results imply that obstacles are to a large extent surmountable.

8.2 Enhancement of method for determination of CPT:s

One mayor task of this project has been to develop a general method where expert beliefs can be included in a systematic way when defining the CPTs in the BBN. The iterative method that is proposed has the ability to produce a relevant and defendable set of conditional probabilities in a BBN even though expert judgement is included in the assessments. The proposed method consists of four parts with corresponding analysis steps, dealing with Network structure, Probability estimation, Sensitivity analysis, and Verification and validation. These steps are performed iteratively until the network is robust and reliable.

The use of either elicitation of single probabilities or of full CPTs, gives a systematic way to include expert beliefs in the network. If the expert is uncomfortable in assigning specific probabilities a method that generates a full CPT is recommended. The advantage is that it better reflects the expert's knowledge of the causalities in the network and therefore is more defendable to use. A drawback with the full CPT methods is that the complexity of the methods requires the constructor to be well versed in their implementation.

The gain of performing sensitivity analysis, although it may be time consuming, is that it will allow rough estimates to be used in a large part of the parameters. Since sensitivity analysis helps to put focus on the most essential parts of the network, where accuracy in the probabilities is important, the resulting BBN will be reliable in spite of the rough estimates used initially. If the sensitivity analysis indicates that parameters, for which only rough estimates are possible to assess, have high sensitivity values the constructor has to consider the required reliability of the network. The constructor may accept these rough estimates and otherwise the structure of the network has to be revised.

The benefit of following this method is that along the way errors in the network can be detected and corrected, or the network is confirmed to be reliable.

9. REFERENCES

1. Grindon, E., Ang. M. L, Kulig, M. Slootman, M., Löffler, H., Horvath, G., Bujan, A., Frid, W., Cholewa, W. and Khatib-Rahbar, M., "A rapid response source term indicator based on plant status for use in emergency response (STERPS)" Proceedings of FISA 2003, November 2003, Luxemburg.
2. Frid, W., Knochenhauer, M., Bednarski, M., "Development of a Bayesian belief network for a boiling water reactor during fault conditions"; CAMES Computer Assisted Mechanics and Engineering Sciences, Vol. 12, No.1 2005.
3. Pourret, O., Naïm, P. & Marcot, B. (Eds.), 2008, Bayesian networks: a practical guide to applications, 73, West Sussex: Wiley.
4. Woodberry, O., Nicholson, A.E., Korb, K.B. & Pollino, C., 2005, Parameterising Bayesian Networks: A Case Study in Ecological Risk Assessment, AI 2004: Advances in Artificial Intelligence, 1101-1107, Berlin Heidelberg: Springer.
5. Charles River Analytics, 2008, About Bayesian Belief Networks, Cambridge: Charles River Analytics, Inc.
6. Langseth, H. & Portinale, L., 2007, Bayesian networks in reliability, Reliability Engineering and System Safety, 92, 92-108.
7. Mihajlovic, V. & Petkovic, M., 2001, Dynamic Bayesian Networks: A State of the Art, Europhysics Letters.
8. Norsys, 1997, NETICA Application for Belief Networks and Influence Diagrams- User's guide, Vancouver: Norsys Software Corp.
9. Coupé, V.H.M., Gaag, L.C. van der & Habbema, J.D.F., 2000, Sensitivity Analysis: an aid for belief-network quantification, The Knowledge Engineering Review, 15, 215-232.
10. Jensen, F.V., 1996, An introduction to Bayesian Networks, London: UCL Press.
11. Coupé, V.H.M. & Gaag, L.C. van der, 2002, Properties of sensitivity analysis of Bayesian belief networks, Annals of Mathematics and Artificial Intelligence, 36, 323-356.
12. Alfheim, P. (2012). Definition and evaluation of a dynamic source term module for use within RASTEP: A feasibility study. MSc Thesis, Uppsala Universitet.
13. Adolfsson, Y., Holmberg, J. E., & al. (2011). Proceedings of the deterministic/probabilistic safety analysis workshop. Espoo.
14. International Atomic Energy Agency. (1995). Procedures for Conducting Probabilistic Safety Assessments of Nuclear Power Plants (Level 2). Vienna: IAEA.
15. EPRI. (2006). MAAP4 Applications Guidance. Palo Alto, CA.
16. Sandia National Laboratories. (2001). MELCOR Computer Code Manuals - Vol. 3: Demonstration Problems.
17. Nilsson, L. (2006). Development of an Input Model to MELCOR 1.8.5 for Oskarshamn 3 BWR. Research, Swedish Nuclear Power Inspectorate (SKI).
18. Raines, J., Hammersley, R., Henry, R., & Blaisdel, J. B. (1993). MARS - An accident management tool. Specialist meeting on operator aids for severe accidents management and training. Halden.
19. Alonso, J. R., Santiago, A., & Alonso, M. a. (2005). Consejo de Seguridad Nuclear use and experience with the MARS software. The 11th International Topical Meeting on Nuclear Reactor Thermal-Hydraulics (NURETH-11). Avignon.

20. Alonso, J. R. (October-December, 2012). Nuclear Installations Coordinator, Consejo de Seguridad Nuclear (CSN), Personal communication .

21. Zavisca, M., Khatib-Rahbar, M., & Esmaili, H. (2002). ADAM: An Accident Diagnostic, Analysis and Management System - Application to Severe Accident Simulation and Management. 10th International Conference on Nuclear Engineering. Arlington, VA, USA.

22. Raines, J. (October-December, 2012). Manager, Fauske & Associates LLC, Personal communication .

23. Knochenhauer, M., & Hedtjärn, V. S. (2012). Using Bayesian Belief Network (BBN) Modelling For Rapid Source Term Prediction – RASTEP Phase 1. Internal report, Scandpower AB.

24. Krčál, P. (January 2013). Software Developer,Scandpower AB, Personal communication .

25. Enerholm, A. (January, 2013). Principal consultant, Scandpower AB, Personal communication.

26. Enerholm, A. (2013). Melcor vs MAAP calculation for Oskarshamn 3 station blackout. Stockholm: Scandpower Report 210829-01/R1, September 2013.

27. Sunnevik, K. (2014), Tools for severe accident analysis and source term calculation - Evaluation of MELCOR as an alternative to MAAP in Swedish nuclear safety assessments. M.Sc. thesis report at Uppsala University, to be published early 2014.

28. R. O. Gauntt, J. C. (2005). MELCOR Computer Code Manuals Vol. 1 Primer and User's Guide Version 1.8.6. Sandia National Laboratories.

29. R. O. Gauntt, J. C. (2005). Melcor Computer Code Manuals Vol.2 Reference Manuals Version 1.8.6. Sandia National Laboratories.

30. Ma, W., & Dinh, T.-N. (2008). MELCOR Analysis for Severe Accident Scenarios of the Oskarshamn-3 BWR Related to Power Update. Stockholm: KTH for Swedish Nuclear Power Inspectrate (SKI).

31. Hamnér, B. (1988). Ringhals 1-4, Oskarshamn 1-3, Forsmark 1-3 - Verifiering av vattenskrubber. Ringhalsverket, OKG & Forsmark.

32. Krieg, M.L., 2001, A Tutorial on Bayesian Belief Networks, Edinburgh: DSTO Electronics and Surveillance Research Laboratory.

33. Rajabally, E., Sen, P., Whittle, S. & Dalton, J., 2004, Aids to Bayesian Belief Network Construction, Second IEEE International Conference on Intelligent Systems.

34. Henrion, M., 1987, Practical issues in constructing a Bayes' belief network, Proceedings of Third Workshop on Uncertainty in AI, AAAI, Seattle, Wash, 132-139.

35. Wisse, B. W., Van Gosliga, S. P., Van Elst, N. P. & Barros, A. I., 2008, Relieving the elicitation burden of Bayesian Belief Networks, Sixth Bayesian Modelling Applications Workshop on UAI, 26.

36. Cooke, R.M. & Goossens, H.J., 2000, Procedures guide for structured expert judgement in accident consequence modelling, Radiation Protection Dosimetry, 90, 303-309, Nuclear Technology Publishing.

37. Renooij, S., 2001, Probability elicitation for belief networks: issues to consider, The Knowledge Engineering Review, 16, 255-269.

38. Burgman, M., Fidler, F., McBride, M., Walshe, T. & Wintle, B., 2006, ACERA Project 0611: Eliciting Expert Judgment, Melbourne: Australian Centre for Excellence in Risk

Analysis.

39. Gaag, L.C. van der, Renooij, S., Witteman, C.L.M., Aleman, B. & Taal, B., 1999, How to elicit many probabilities, Uncertainty in artificial intelligence, 647-654, San Francisco, CA: Kaufman.
40. NETICA, Bayesian network development software, Norsys Software Corp. www.norsys.com
41. Kemp-Benedict, E., 2008, Elicitation Techniques for Bayesian Network Models, Working Paper WP-US-0804, Stockholm: Stockholm Environment Institute.
42. Kemp-Benedict, E., Bharwani, S., de la Rosa, E., Krittasudthacheewa, C. & Matin, N., 2009, Assessing Water-related Poverty Using the Sustainable Livelihoods Framework, Stockholm Environment Institute Working Paper, Stockholm: Stockholm Environment Institute.
43. Das, B., 2004, Generating conditional probabilities for Bayesian networks: Easing the knowledge acquisition problem, Edinburgh: DSTO Command and Control Division.
44. Bednarski, M., Cholewa, W. & Frid, W., 2004, Identification of sensitivities in Bayesian networks, Engineering Applications of Artificial Intelligence, 17, 327-335.
45. Jensen, F.V., 2001, Bayesian Networks and Decision Graphs, New York: Springer.
46. Marcot, B.G., 2012, Metrics for evaluating performance and uncertainty of Bayesian network models, Ecological Modelling, 230, 50-62.
47. Coupé, V.M.H., Peek, N., Ottenkamp, J. & Habbema, J.D.F., 1999, Using sensitivity analysis for efficient quantification of a belief network, Artificial Intelligence in Medicine, 17, 223-247.
48. Gaag, L.C. van der, Renooij, S. & Coupé, V.M.H., 2007, Sensitivity Analysis of Probabilistic Networks, Advances in Probabilistic Graphical Models: Studies in Fuzziness and Soft Computing, 213, 103-124, Berlin: Springer.
49. Gaag, L.C. van der, 2001, Analysing Sensitivity Data from Probabilistic Networks, Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence, 530-537.
50. Hedtjärn Swaling, V., Frid, W., Knochenhauer, M. & Lundtofte, C., 2011, Using PSA to Develop a Tool for Rapid Source Term Prediction Based on Bayesian Belief Networks, Nordic PSA Conference "Castle Meeting 2011".
51. OECD/NEA WGAMA. International benchmarking project on fast-running software tools used to model fission product releases during accidents at nuclear power plants (FASTRUN). Task GAMA (2012)2

Title	Using Bayesian Belief Network (BBN) Modelling for Rapid Source Term Prediction – Final Report
Author(s)	Michael Knochenhauer, Vidar Hedtjärn Swaling, Francesco Di Dedda, Frida Hansson, Stina Sjökvist and Klas Sunnegård
Affiliation(s)	Lloyd's Register Consulting AB, SWEDEN
ISBN	978-87-7893-369-0
Date	October 2013
Project	NKS-R and B / RASTEP
No. of pages	78
No. of tables	17
No. of illustrations	29
No. of references	51

Abstract

The project presented in this report deals with a number of complex issues related to the development of a tool for rapid source term prediction (RASTEP), based on a plant model represented as a Bayesian belief network (BBN) and a source term module which is used for assigning relevant source terms to BBN end states. Thus, RASTEP uses a BBN to model severe accident progression in a nuclear power plant in combination with pre-calculated source terms (i.e., amount, composition, timing, and release path of released radio-nuclides). The output is a set of possible source terms with associated probabilities. One major issue has been associated with the integration of probabilistic and deterministic analyses are addressed, dealing with the challenge of making the source term determination flexible enough to give reliable and valid output throughout the accident scenario. The potential for connecting RASTEP to a fast running source term prediction code has been explored, as well as alternative ways of improving the deterministic connections of the tool. As part of the investigation, a comparison of two deterministic severe accident analysis codes has been performed. A second important task has been to develop a general method where experts' beliefs can be included in a systematic way when defining the conditional probability tables (CPTs) in the BBN. The proposed method includes expert judgement in a systematic way when defining the CPTs of a BBN. Using this iterative method results in a reliable BBN even though expert judgements, with their associated uncertainties, have been used. It also simplifies verification and validation of the considerable amounts of quantitative data included in a BBN.

Key words BBN, Bayesian Belief Network, Severe Accidents, Source Terms, Level 2 PSA, CPT, Conditional Probability Tables

Available on request from the NKS Secretariat, P.O.Box 49, DK-4000 Roskilde, Denmark.
Phone (+45) 4677 4041, e-mail nks@nks.org, www.nks.org