Risk Management in Intelligent Agents

Xun Wang

A Thesis submitted for the degree of Doctor of Philosophy

Faculty of Engineering and Information Technology

University of Technology, Sydney

2012

Certificate of Authorship and Originality

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Student

Acknowledgements

Working on my doctoral research has been one of the most memorable experiences of my life. I would like to thank all people who have helped and inspired me during the past five years of my study.

First and foremost, I would like to thank my supervisor, Professor Mary-Anne Williams. Without your inspiration and encouragement, I would not even contemplate working on my Ph.D research project. Throughout my Ph.D career, you have shown me what would be possible and steer me towards my research goals. You have given me great support in developing my main research ideas; challenged me on many of my immature thoughts. You have pushed me to reach a level of scientific standards that I would not be able to achieve alone. In addition to my main research project, you have also encouraged me to explore new technologies and supported me on fun and challenging projects such as RoboCup. These "side" projects have kept me sane when my main research topic overwhelmed me at times. "Research should be fun" is what I have learned from you. You are a great mentor and a friend. I am deeply grateful for what you have done for me.

Second, I would like to thank many of my colleagues for reviewing my thesis and collaborating with me on many projects. Birger Johansson was the first person who read my thesis draft while it was still an incoherent piece of text. Your suggestions were very useful. I enjoyed many insightful discussions with you while you were visiting our laboratory. I also had the great opportunity working with Chris Stanton on the RoboCup project which I enjoyed enormously. Thanks for correcting many of my grammatical errors in my thesis. Your comments on how to structure my work has greatly helped me. Sajjad Haider taught me a great deal on machine learning and Bayesian inference techniques. Your inputs on my thesis have further improved some of my research ideas. I would also like to thank Benjamin Johnston for his great effort going through my final thesis draft in details. Many of your comments were invaluable. I am also in debt to Shan Chen for keeping my spirit up when I was exhausted at times. I also had pleasure working with Jebrin Al-Sharawneh. Our discussions on what is risk were challenging and insightful.

Third, I would like to thank Thierry Denoeux for our discussion on Belief Functions and Transferable Belief Model during KSEM 2010 conference and many anonymous reviewers for their comments and suggestions on the ideas presented in my research papers. My thesis examiners Didier Dubois and Pavlos Peppas have given me valuable critiques and insightful suggestions. I am forever grateful for your helps! I would also like to thank IBM for their financial support during the fourth year of my Ph.D career.

Finally, I would like to thank my parents, my wife Lily and my daughter Alison for putting up with me for such a long time. Without your support and understanding, this thesis would not be possible.

Table of Contents

Table of Contents iv

Abstract xiv

1 Introduction 1

2 An Analysis of Risk Related Concepts 8
- 2.1 Definitions of Risk . . . 9
- 2.2 Conceptualising Risk for Intelligent Agents . . . 11
- 2.3 Uncertainties, Probabilities, Belief and Possible Worlds . . . 13
  - 2.3.1 Possible Worlds Paradigm . . . 14
  - 2.3.2 Encoding Uncertainty in a Likelihood Order . . . 15
  - 2.3.3 Encoding Uncertainty in Probability . . . 16
- 2.4 Measuring Consequence . . . 18
  - 2.4.1 Qualitative Consequence . . . 18
  - 2.4.2 Quantitative Consequence . . . 19
- 2.5 Combining Uncertainty and Consequence . . . 19
  - 2.5.1 Risk Matrix for Qualitative Risk Representation . . . 20
  - 2.5.2 Quantitative Risk Measure . . . 21
    - 2.5.2.1 Expected Utility . . . 22
    - 2.5.2.2 Probability-EU Space . . . 22
  - 2.5.3 Application of the Risk Measure . . . 25
    - 2.5.3.1 Value at Risk . . . 25
- 2.6 Summary . . . 26

3 Requirements for a Generalised Risk Management Framework 29
- 3.1 Benchmark Problems for Risk Modelling . . . 30
  - 3.1.1 Benchmark Problem 1 - Ball Passing Problem . . . 30
  - 3.1.2 Benchmark Problem 2 - Foreign Exchange (FX) . . . 31
- 3.2 Framework Requirements . . . 32

3.3 Existing Risk Management Methodologies 35
3.3.1 Fault Tree Analysis (FTA) 35
3.3.2 Event Tree Analysis (ETA) 37
3.3.3 Failure Mode and Effects Analysis (FMEA) 38
3.3.4 Probabilistic Models 39
3.3.5 Hierarchical Holographic Models 40
3.3.6 Dealing with Uncertainty under Partial Ignorance 40
3.3.7 A Comparison with the Framework Requirements 41

4 Knowledge Representations and Management in Complex and Uncertain Environments 43
4.1 Classical Logic 45
4.1.1 Propositional Logic 46
4.1.1.1 Interpretation and Knowledge Base 46
4.1.1.2 Forms of Propositional Formula 47
4.1.2 First Order Logic 48
4.1.3 Second Order Logic 49
4.2 Non-Classical Logic 50
4.2.1 Default Logic 51
4.2.1.1 Syntax of Default Logic 51
4.2.1.2 Semantics of Default Logic 52
4.2.1.3 Operational Semantics of Default Logic 52
4.2.1.4 Variations of Extensions 53
4.2.1.5 *Applicability of Default Logic 54
4.2.2 Autoepistemic Logic 54
4.2.2.1 Semantics of Autoepistemic Logic 55
4.2.2.2 Computation of Expansions 56
4.2.2.3 *Relation with Default Logic 57
4.2.3 Circumscription 57
4.2.3.1 Syntax of Circumscription 58
4.2.3.2 Semantics of Circumscription 59
4.2.3.3 Applicability of Circumscription 60
4.2.4 *Incomplete Knowledge and Changing Information 60
4.3 Belief Revision 61
4.3.1 AGM Paradigm 62
4.3.1.1 Preliminaries 62
4.3.1.2 Postulates for Expansion 63
4.3.1.3 Postulates for Revision 63
4.3.1.4 Postulates for Contraction 64
4.3.1.5 Belief Contraction versus Belief Revision 65

4.3.2 Belief Operator Selection 65
4.3.2.1 Epistemic Entrenchment 66
4.3.2.2 System of Spheres 67
4.3.3 Belief Base Change 69
4.3.4 Iterative Belief Revision 70
4.3.4.1 Ordinal Conditional Functions (OCF) 70
4.3.4.2 Transmutations 71
4.3.4.3 Ordinal Epistemic Functions 71
4.4 Bayesian Probabilistic Model 75
4.4.1 Conditional Probability 75
4.4.2 Conditional Independence 76
4.4.3 Bayesian Networks 77
4.4.4 Bayesian Network Construction 79
4.4.4.1 Learning Network Parameters from Complete Data . 81
4.4.4.2 Learning Network Parameters from Incomplete Data 83
4.4.4.3 Learning Network Structures 86
4.4.5 Inferences in Bayesian Networks 90
4.4.5.1 Exact Inference 90
4.4.5.2 Approximate Inference 98
4.4.6 *Limitations of Bayesian Networks 103
4.5 Transferable Belief Model 105
4.5.1 Belief Function 106
4.5.2 Simple Support Function 108
4.5.3 Vacuous Belief Function 108
4.5.4 Latent Belief Structure 108
4.5.5 The Λ Operator and Apparent Belief 109
4.5.6 Rules of Combination 109
4.5.7 Conditional Belief Function and Rules of Conditioning 111
4.5.8 Generalised Bayesian Theorem 112
4.5.9 Evidential Network 113
4.5.10 Pignistic Transformation 114
4.5.11 *TBM versus Probabilistic Model 115
4.6 Possibility Theory 116
4.6.1 Possibility Distributions 117
4.6.2 Possibility and Necessity Measures 118
4.6.2.1 Comparative Possibility Relation 119
4.6.2.2 Conditioning and Combination in Qualitative Possibility 121
4.6.2.3 Quantitative Possibility Measure 122
4.6.2.4 Conditioning and Combination in Quantitative Possibility 123

4.6.3 *Connecting Qualitative and Quantitative Uncertainty 124
4.7 Capturing Causal Connections 125
4.7.1 Ramsey Test 125
4.7.1.1 Compatibility with Belief Revision 126
4.7.2 Inductive Causation 128
4.8 *Summary 129

5 An Overview of HiRMA 132
5.1 Three-level Framework Architecture 133
5.2 Iterative Risk Modelling Process 136
5.3 A Knowledge Description Template for Risk Analysis 138
5.4 Theoretical Assumptions 142

6 Risk Management with Qualitative Knowledge 143
6.1 Qualitative Epistemic Reasoning Model 144
6.1.1 Epistemic Reasoning Model Schema 144
6.2 Qualitative Graphical Risk Model 149
6.2.1 Capturing Knowledge in the Ball Passing Problem 150
6.2.2 Risk Modelling in a System of Spheres 152
6.2.2.1 Epistemic Reasoning Models within a Sphere 153
6.3 Risk Model Construction and Revision 154
6.3.1 Revision of Domain Variables 154
6.3.1.1 Semantics of Variable Addition 155
6.3.1.2 Removal of a Domain Variable 156
6.3.2 Revision of Reasons 157
6.3.3 Model Construction and Revision Algorithms 158
6.3.3.1 Modelling with the Ball Passing Problem 163
6.4 Decision Making under Qualitative Risk Model 164
6.5 Discussion 167

7 Risk Management with Quantitative Knowledge 172
7.1 Quantitative Graphical Risk Model 174
7.1.1 Model Quantified Epistemic Reasons 175
7.1.1.1 Lead – Quantified Epistemic Reason 176
7.1.1.2 Frame of Discernment Ω_X 177
7.1.1.3 Data Fusion in Lead 178
7.1.1.4 Latent Lead 179
7.1.1.5 The $\dot{\Lambda}$ Operator and Causal Strength 180
7.1.1.6 The Ranking Structure RS 181
7.2 Risk Model Construction and Revision 182
7.2.1 Revision of Domain Variables 182

7.2.2 Revision of Leads 185
7.2.3 Model Construction and Revision Algorithms 186
7.2.4 Graphical Probabilistic Model Generation 187
7.2.5 Modelling with Ball Passing Problem 189
7.3 Discussion 190

8 A Unified Multi-level Iterative Framework 195
8.1 Bridging the Qualitative and Quantitative Divide in Risk Management 196
8.2 Framework Evaluation 199
8.3 Risk Modelling Strategies 201
8.4 A Generic Software Architecture 203

9 Conclusions 207

A An Extended FX Example 214
A.1 Initial Risk Analysis 214
A.2 Analysis and Selection of Modelling Process 216
A.3 Knowledge Databases for Modelling FX risk 216
A.3.1 Database Structure for $\mathbb{K}$ 216
A.3.2 Database Structure for $\mathbb{K}^c$ 218
A.3.3 Populated Knowledge Database $\mathbb{K}$ 219
A.4 Functional Code Snippets 221
A.4.1 add_variable 222
A.4.2 add_metavariable 223
A.4.3 remove_variable 224
A.4.4 remove_reason 224
A.4.5 revise_reason 225
A.4.6 consolidate_reason 227

B Published Conference Papers 228

Bibliography 229

List of Figures

1.1 An intelligent agent as a black box, adapted from Poole et al. (1998). 3

2.1 Possible worlds in a likelihood preorder. 16
2.2 Probability-EU space and Probability-Consequence space. 23
2.3 VaR in Probability-Consequence space. 27

3.1 Ball passing between robots 30
3.2 An example of a fault tree (Stephans 2005). 36
3.3 An example of an event tree (Andrews & Dunnett 2000). 37
3.4 An example of a FMEA table (Aven 2008). 38
3.5 An example of a Hierarchical Holographic Model (Liu & Jiang 2012). 39

4.1 A System of Spheres 67
4.2 A Directed Acyclic Graph for the Ball Passing Problem. 77
4.3 A (nonminimal) jointree for the Bayesian network shown in Figure 4.2. 94
4.4 Conversion of a Bayesian network (left) to a likelihood weighting network (right) based on evidence on variables B and E. The CPTs for B and E are modified such that, for example, if $B = b$ and $E = \bar{e}$, then $\theta_b = 1$, $\theta_{\bar{b}} = 0$ and $\theta_e = 0$, $\theta_{\bar{e}} = 1$. This figure is taken from Darwiche (2009). 103

5.1 High level theoretical relationships between qualitative and quantitative modelling approach for risk modelling and management. 135

5.2 A typical risk modelling and management process in HiRMA (at medium domain knowledge abstraction). 137

6.1 Epistemic reasoning models for ball-passing under two initial domain contexts. 151

6.2 A ball passing risk model with all possible epistemic reasoning models captured in a System of Spheres. 153

6.3 Revised ball passing risk model after adjustment with $E(ReasonFor(D, S_1) \wedge ReasonFor(D, S_2)) = 1$. 158

6.4 Evolution of a simple risk model for ball passing problem. '+' means addition and '-' means contraction. Vacuous reasons are shown in steps in which new variables are added for illustration purpose. 171

7.1 Evolution of a quantitative epistemic reasoning model for risk: initial model setup (a) and corresponding ranking structure (b). 175

7.2 Evolution of a quantitative epistemic reasoning model for risk: Fuse additional lead information from another source with $m(L_{D \to S_1}) = 0.5$, $m(L_{D \to S_2}) = 0.3$ and $m(\Omega_D) = 0.2$. 178

7.3 Evolution of a quantitative epistemic reasoning model for risk: addition of a risk factor NR (Nearby Robot). 184

7.4 Evolution of a quantitative epistemic reasoning model for risk: update lead $L^{0.7}_{NR \to S_2}$. 185

7.5 Evolution of a simple risk model for ball passing problem. '+' means addition and '-' means contraction. $\oslash$ means data fusion with TBM conjunctive combination rule. Vacuous leads are shown in steps in which new variables are added for illustration purpose. 194

8.1 A flowchart for selecting appropriate risk modelling and management process. 202

8.2 A schematic diagram of a generic software architecture for implementation of the HiRMA framework. 204

A.1 Database schema for a FX risk model at credal level. 216
A.2 Database schema for a consolidated FX risk model. 219
A.3 A snapshot of the quantitative epistemic reasoning model for FX risks. 222

List of Tables

2.1 A conventional risk matrix. 20
2.2 A risk matrix with possible world preorder structure. 21

3.1 A feature comparison of the existing risk management methodologies. 41

4.1 Truth-valued function associated with unary connective. 47
4.2 Truth-valued function associated with the binary connectives. 48
4.3 Conditional Probability Tables (CPT) for $Pr(p|k, o)$ and $Pr(i|k, o)$. . 78
4.4 A complete set of data samples and empirical distribution for ball passing BN. 81
4.5 Examples of incomplete dataset. 83
4.6 A set of samples generated from network Figure 4.2. 99
4.7 An example of a conditional belief function represented in a table of bbms. 113

5.1 A high-level theoretical architecture of HiRMA. 133
5.2 A domain knowledge description template for knowledge engineer undertaking risk analysis. 139
5.3 A simplified analysis of possible scenarios. 140
5.4 A simplified analysis of possible scenarios for FX risk in Australian Dollars. 141

6.1 Graphical symbols and their corresponding meanings. 150

8.1 A high-level theoretical architecture of HiRMA. 197
8.2 A feature comparison between HiRMA and the existing risk management methodologies. 201

A.1 An analysis of possible scenarios for FX risk in Australian Dollars (AUD). 215

Abstract

This thesis presents the development of a generalised risk analysis, modelling and management framework for intelligent agents based on the state-of-art techniques from knowledge representation and uncertainty management in the field of Artificial Intelligence (AI). Assessment and management of risk are well established common practices in human society. However, formal recognition and treatment of risk are not usually considered in the design and implementation of (most existing) intelligent agents and information systems. This thesis aims to fill this gap and improve the overall performance of an intelligent agent. By providing a formal framework that can be easily implemented in practice, my work enables an agent to assess and manage relevant domain risks in a consistent, systematic and intelligent manner.

In this thesis, I canvas a wide range of theories and techniques in AI research that deal with uncertainty representation and management. I formulated a generalised concept of risk for intelligent agents and developed formal qualitative and quantitative representations of risk based on the Possible Worlds paradigm. By adapting a selection of mature knowledge modelling and reasoning techniques, I develop a qualitative and a quantitative approach of modelling domains for risk assessment and management. Both approaches are developed under the same theoretical assumptions and use the same domain analysis procedure; both share a similar iterative process to maintain and improve domain knowledge base continuously over time. Most importantly, the knowledge modelling and reasoning techniques used in both approaches share the same underlying paradigm of Possible Worlds. The close connection between the two risk modelling and reasoning approaches leads us to combine them into a hybrid, multi-level, iterative risk modelling and management framework for intelligent agents, or HiRMA, that is generalised for risk modelling and management in many disparate problem domains and environments. Finally, I provide a top-level guide on how HiRMA can be implemented in a practical domain and a software architecture for such an implementation. My work lays a solid foundation for building better decision support tools (with respect to risk management) that can be integrated into existing or future intelligent agents.

Chapter 1

Introduction

Risks exist in almost every aspect of life. As intelligent beings, humans need to assess and manage risks under various circumstances almost constantly. Even a simple act of crossing a road involves identifying dangerous incoming traffic, assessing the risk of possible collision and taking appropriate actions to reduce such a risk. Most of us can handle this task effortlessly. In more complex task domains such as operating an oil refinery or investing in a foreign company, assessing and managing risks becomes complicated but being able to deal with risks systematically and effectively is critically important for operations in these domains. That is, any loss of life, contamination of the environment or heavy monetary loss due to failure in recognising and managing risk would be disastrous. Clearly, the capability of risk handling should be viewed as an important characteristic of *intelligence*.

With ever increasing processing power and sophistication in software, more and more computer (or generally information) systems are taking over the tasks that used to be performed by people. Such systems range from simple traffic (light) controllers to advanced flight control systems on modern aeroplanes; from managing mundane business transactions to automated electronic share trading systems (Hendershott & Riordan 2009); even automated driverless cars will soon enter into our daily life (Markoff 2011). Clearly, it is necessary for these systems to be able to handle risks associated with these tasks that used to be managed by people. Till now, most existing computerised systems do not conceptualise, assess or manage risk explicitly.

The task of handling risks is usually subsumed into the functions that handle error or failure conditions specific to the problem domain. Consequently, a “risk” handling mechanism found in one system cannot be transferred into a different system. Systems are designed to handle specific conditions and have no or little ability to infer or reason about risks, especially in open and evolving environments. There is little attention to treat risk management as an important integrated functionality for most of the existing information systems. There seems to be little investigation of the meaning of risk from the perspective of an *intelligent* agent or information system, and how to model and manage risks in such a system.

In this thesis, I present my research work in building a Hybrid iterative Risk Management framework for intelligent Agents (HiRMA), based on the existing research work in the field of Artificial Intelligence (AI). I view an intelligent agent or information system[1] mainly as a software system that is designed and implemented according to some pre-defined goals and objectives for operating in a specific problem domain. The agent (Figure 1.1) has certain prior knowledge about the domain. It continuously takes relevant domain information from its operating environment and produces decision making outcomes that can be acted on for reducing risks associated with the domain.

In order to develop this risk modelling and management framework, I first investigate and formalise a concept of risk specifically for intelligent systems, since there is no clear definition of risk for such a system. I canvas the wide field of Artificial Intelligence and adapt a selection of mature AI technologies, particularly in the areas of knowledge representation and uncertainty management, in building the risk modelling and management framework. Throughout this thesis, I draw upon two disparate but challenging domains, namely, treasury risk management for businesses, and risk management for mobile robots, to illustrate the key challenges in the research and my approach in addressing these issues.

Fluctuations in foreign currency exchange (FX) rates present significant risk to a

[1]In this thesis, I use “intelligent agent” and “intelligent information system” interchangeably as I focus on risk management and decision making capabilities of agents and systems.

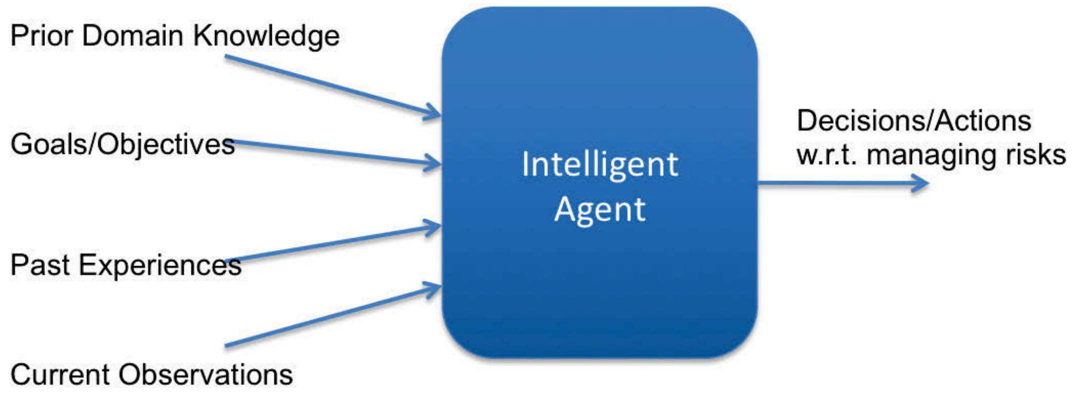


Figure 1.1: An intelligent agent as a black box, adapted from Poole et al. (1998).

business' viability, particularly in a global economy. A mobile robot playing soccer must be able to use its background knowledge and interpret sensory information to assess the risks (with respect to its goals) present in a competition environment. For an example, a ball passing from a robot to another carries risks such as interception from an opponent robot. Whereas in foreign exchange market, it is crucial to be able to respond to ever changing market conditions and minimise various risks presented for many companies to remain viable (Abbott 2009). Being able to deal with various risks rationally and effectively is crucially important for an intelligent system to perform its functions and achieve its objectives. Unfortunately, the concept of risk usually has different meanings for different domains, and there is no precise and generally accepted definition for risk. In the real world, people frequently operate in complex and dynamic environments where dealing with various risks in a consistent manner is difficult and laborious. Utilising works from AI research, we can develop intelligent software tools or agents to assist us in dealing with risks. However, managing risks has not been a main focus in AI research, even though there is a strong tradition and extensive work in dealing and modelling uncertain information in decision making[2]. This thesis explores risk management and takes a holistic approach in developing a

[2]There are some recent works done by Dubois et al.(2011) closely related to this thesis.

formal and practical framework for risk management in *open*, *dynamic* and *complex* environments based on an integration of the existing theories and techniques developed in the AI research. With this research work, I hope to spur further practical development in intelligent software tools/agents to assist risk management for a wide range of domains and help to push existing AI theories and techniques into real world applications, making new and existing information systems more "intelligent". More specifically, I aim to achieve the following key objectives:

1. Provide a clear conceptualisation of *risk* for information systems and a definition of *risk* that is general enough to be applicable to information systems in a wide range of domains; and develop a formal way of evaluating and measuring risks.

2. Develop a new framework for design and development of risk management systems based on the formal concept of risk and a new approach to represent and reasoning about risk. This framework will provide formal approaches, based on an integration of the existing AI theories and methods, to build and maintain a domain knowledge base for risk modelling purpose. Existing AI reasoning methods will be used to reason, evaluate risks with the knowledge base. Consequently, I should be able to develop strategies to mitigate or minimise risks and enhance possible opportunities.

3. The framework will be illustrated and evaluated using practical benchmark problems.

4. Propose a generic software architecture for implementing the risk modelling and management framework.

Aligned with the above stated objectives, the main content of this thesis is divided into seven chapters. To address the first objective, I provide a survey and analysis of definitions of risk in a range of domains in Chapter 2. From this analysis, I distill the essential notions found in (almost) all of these definitions: namely, a *combination* of *uncertainty* and *consequence*. Based on these two key notions, I provide a generalised definition of risk that can be adapted for intelligent agents operating in a wide range of domains. I then analyse the meanings of uncertainty and consequence from the ground up. The result of the analysis is two formal interpretations for uncertainty and consequence. I then propose two improved (and formal) measures for representing qualitative risk and quantitative risk respectively.

Chapter 3 explores a number of key issues that I must address in order to develop a generalised risk management framework for intelligent agents operate in disparate problem domains. To this end, I introduce the two benchmark problems briefly mentioned in the previous paragraphs. I set out a list of key requirements for my framework and discuss the necessity of these requirements using the benchmark problems as the supporting examples. Finally, I provide a quick survey of the existing risk management frameworks with respect to these requirements.

In order to accomplish the second objective, I start with a wide range survey of knowledge representation and management techniques developed in AI in Chapter 4, focusing on methods in the area of uncertainty management. I analyse these techniques with respect to the framework requirements and discuss their individual merits and deficiencies. It should be noted that I only cover the well-known and mature techniques and discuss the key ideas of these methods. It is simply infeasible to cover every aspect of research work done over the past five decades within the practical limits of this thesis. However, I do list the most relevant results and algorithms, and show how they are integrated in my framework as parts of overall strategy for practical risk modelling and management. Discussions in Chapter 4 lead to an overview of the HiRMA framework I developed in this research work. In Chapter 5, I show what techniques are utilised and integrated together in HiRMA and how they are interrelated under the Possible Worlds paradigm. Furthermore, I provide a generic domain risk analysis technique based on the formal risk conceptualisation developed in Chapter 2.

In Chapter 6 and 7, I present a detailed description and analysis of HiRMA framework in two separate approaches: qualitative and semi-quantitative/quantitative risk modelling and management processes respectively. I use the ball passing benchmark problem to provide detailed illustrations of the two approaches. I highlight the key features within both processes and match them against the framework requirements. Limitations within the process are also discussed and model construction and revision algorithms are provided. In Chapter 8, I bring these two approaches together to form the unified, multi-level, iterative risk management framework. I demonstrate how the qualitative risk modelling process may be connected to the quantitative part of risk modelling process. I provide a pathway to translate a pure qualitative model into a quantitative model (with additional information) and vice versa. I will show modelling and decision making strategies an agent may adopt under HiRMA for its specific domain and operating environment. Finally, I evaluate my framework against the key framework requirements I set out in Chapter 2 and compare HiRMA against the well established risk modelling and management methodologies in mainstream risk management literatures. This completes the objective 2 and 3.

I conclude this thesis with a top-level software implementation architecture based on the risk modelling and management framework (objective 4). Combined with the algorithmic recipes listed throughout the thesis, a system implementor should be able to use the architecture as a guidebook for software implementation. I also suggest few current technologies that a system implementor may adopt for software development. Appendix A provides a partial implementation of HiRMA using an extended FX benchmark problem as an example.

To conclude this introduction, I would like to highlight few key points of this project:

- I investigate and develop a concept of risk, risk modelling and management for intelligent information systems and agents from the ground up, with its foundation firmly placed in AI, in particular, in the areas of knowledge representation and uncertainty management. Therefore, this research work emphasises the formal treatment of risk and risk model construction. I attempt to minimise (if not remove) adhoc nature and descriptiveness commonly found in the mainstream risk management related literature. Practical implementation of a risk management framework in intelligent agents demands a normative approach rather a descriptive one. This is the key difference between this research work and mainstream risk management related research.

- Another key achievement of this thesis is to explore a relatively untouched area of application for various AI technologies developed in the past decades. I consider many AI technologies have been, so far, poorly utilised in many practical domains. Adopting these technologies in the area of risk management can help to address real-world problems and further improve the "intelligence" in so-called intelligent agents.

- In this research, I focus my effort in the mature, solid and proven technologies (and algorithms) instead of exploring the latest developments in the same area. This is because practical implementation is one of the main requirements for the framework.

- The risk modelling and management processes presented in this thesis do not represent a direct application of chosen knowledge modelling and reasoning techniques. I modify and integrate the existing methods to satisfy the particular needs of the framework. However, these custom modifications do not deviate from the standard methods theoretically.

Finally, the HiRMA framework will provide a set of procedures for system designers and developer who wish to develop risk modelling and management capabilities for intelligent agents or information systems.

Chapter 2

An Analysis of Risk Related Concepts

Risk as a concept has been long recognised in human societies. However, risk typically carries different meanings in different domains under different contexts. For example, in the health and safety fields, risk is typically viewed from a personal physical safety perspective; whereas in the management of computer data centres, risk is usually considered in terms of security of the physical systems and possible data loss and service disruptions. The connection between risk in disparate fields remains vague and risk is usually managed in entirely different ways, even though risk is a fundamental concept. Most software solutions recently developed for dealing with risk remain domain specific and cannot be applied across different industries so techniques developed in one area cannot easily be transferred to solve problems in another (RiskMetrics 1996, Mediregs 2011, Makhoul, Saadi & Pham 2010). Only recently, have much needed steps been taken towards more general risk assessment management. In my research, the goal is to build a generalised risk modelling and management framework for intelligent agents operating in disparate domains. Therefore, it is essential to formalise the concept of risk and construct a meaning of risk that can be used to enable intelligent agents to deal with risk in a wide range of domains. This formalisation of risk will lead to *representations* of risk so that appropriate knowledge representation and management methods developed in AI can be utilised to model and manage risks.

In this chapter, I will first survey a number of existing definitions for risk found in risk management literature and examine the underlying concepts and key features of risk. My study shows that uncertainty and consequence of events (possibly caused by actions) are fundamental to the understanding of risk in all contexts. I will present a general definition of risk from the perspective of an intelligent agent and provide a formal interpretation of uncertainty using the Possible Worlds paradigm. Combined with different ways of measuring consequence, I investigate both qualitative and quantitative representations of risk with respect to the commonly used methods found in many literatures. I present my own variants of risk representations that formalise and extend the existing techniques. Finally, I will discuss some of the implications of my definition and representations of risk.

2.1 Definitions of Risk

The concept of risk, in its general form, is complex and has not been precisely defined for general applications. Although people share a general notion of risk, it often carries different technical and practical meanings in different domains and can be interpreted from different perspectives. I have surveyed risk related literature widely from various fields. The following prominent definitions of risk illustrate the diversity and common themes in defining risk related concepts in these fields:

1. In *Medical Risk Management*, Richards et al. (1983) states "a risk is an exposure to the chance of injury or financial loss".

2. Tapiero (2004) states "Risk results from the direct and indirect adverse consequences of outcomes and events that were not accounted for", "risk involves (i) consequences, (ii) their probabilities and their distribution, (iii) individual preferences, and (iv) collective, market and sharing effects."

3. Cool (2001) defines risk as the "absolute value of probable loss" in the area of econometrics.

4. Kaplan and Garrick (1981) represent risk as a set of triplets $\{\langle s_i, p_i, x_i \rangle\},\ i = 1, 2, ...N$ where s_i is a scenario, p_i is the likelihood of the scenario and x_i is the consequence of the scenario.

5. In Risk Analysis (Aven 2008), risk is "the combination of (i) events A and the consequences of these events C, and (ii) the associated uncertainties U".

6. ISO[1] Guide 73 (ISO 2009b) defines risk as follows, "Risk can be defined as the combination of the probability of an event and its consequences".

7. ISO 31000:2009 (ISO 2009a) changes the definition of risk to the "effect of uncertainty on objectives"; "Risk is often expressed in terms of a combination of the consequences of an event (including changes in circumstances) and the associated likelihood of occurrence".

Within the seven definitions stated above, the first four are specialised risk definitions from three practical domains, namely, medical health (and insurance), finance and econometrics; while the remaining three are generalised definitions of risk from the risk management literature and the international standardisation body ISO. The definitions from specific domains focus on the chance of events; their probabilities are mainly concerned with the negative effect of such events such as "injury or financial loss". The generalised definitions emphasise that a risk is the combination of probability of an uncertain event with its effect or consequence. The effect or consequence of an event is also no longer exclusively associated with negativity or loss in these definitions. Interestingly, the latest standard definition of risk in ISO 31000 moves away from the past emphasis on the likelihood of an event to the likelihood of an effect on objectives.

[1]International Organisation for Standardisation, i.e. ISO, is an international body that sets worldwide proprietary, industrial and commercial standards.

Clearly, *uncertainty* and *consequence* are two key underlying concepts that appear consistently across all these definitions. Note that, even though the word "probability" is frequently used in some of the definitions, it is just one particular mechanism for representing uncertainty and used in the specialised domains such as finance. The majority of generalised definitions of risk still use uncertainty. In this thesis, I will make a clear distinction between uncertainty and probability. The consequence of an event can only be meaningfully evaluated with respect to a domain, its context and objectives. This important aspect of consequence is clearly reflected in the shift of focus in ISO 31000. In addition, a consequence is anything that deviates from the expected outcomes (ISO 2009a)(note 1). It can have either positive or adverse effects on objectives, goals or desired outcomes.

2.2 Conceptualising Risk for Intelligent Agents

From the range of risk definitions discussed in the previous section, none of the existing definitions of risk fully satisfies what is required for the HiRMA framework. My generalised risk analysis, modelling and management framework's key design objective is to assist agents operate in a variety of task domains. The specialised risk definitions (definition 1 to 4) are tailored to work well in their respective problem domains, e.g. using probability to represent uncertainty in finance, but not in others; whereas the generalised risk definitions do not take the characteristics of a software agent into consideration. That is, a software agent is usually designed with specific goals and specific objectives to achieve. In addition, a software agent usually operates in a well-defined problem domain. Possible events associated with the operation of the agent can normally be analysed and determined during the design and implementation of the agent. The existing generalised definitions (definition 5 to 7) require additional application related details. They do not provide sufficient guidance for domain knowledge acquisition and systematic development of a working risk model for an intelligent agent. Nevertheless, the analysis of the existing definitions of risk is extremely helpful in formalising the concept of risk for the HiRMA framework. I now provide a top-level generalised definition of risk for intelligent agents. It is important to note that we must take the following definition with the formal interpretation of uncertainty and consequence (discussed in the subsequent sections) together in order to have a full conceptualisation of risk for intelligent agents.

Definition 1. *A* ***risk*** *is a combination of the uncertainty of occurrence of a possible event that results from an initial event and the associated positive or negative consequence* [2] *(or payoff/penalty) of the event on an intelligent agent with respect to achieving its objective(s).*

Note that, an initial event is an event identified by risk analysts that is relevant to the system for risk analysis and can trigger subsequent relevant events to occur. For example, the end of financial year could be used as an initial event when we analyse a financial management system.

Definition 2. *A (risk)* **scenario** *is the possible resultant event associated with a risk.*

Uncertainty and consequence used in the above definition should have following general characteristics:

- ***Uncertainty***. The occurrence of an event is not certain. It may or may not happen. The likelihood of its occurrence depends on the domain context and various factors associated with the domain environment in which an agent operates.

- ***Consequence***. Consequences are highly dependent on the agent's goal/agenda, the environment the agent is operating in, and to the agent's capabilities.

[2] *In risk analysis and management literature, 'consequence' is widely used instead of 'payoff' or 'penalty'. I use 'consequence' with the word 'payoff' or 'penalty' interchangeably.*

A consequence could have either positive (payoff) or negative impact[3](penalty) and it must have relevant meaning for the stake-holders.

The characteristics of the key notions indicate that risk is highly dependent on the problem domain and associated environment. Therefore, this definition of risk for intelligent agents still uses the ambiguous terms of uncertainty, consequence and combination in order to accommodate risk modelling in a wide range of domains as possible. However, these terms will be formalised when the definition is applied to a concrete domain. I now give a detailed analysis of these terms under qualitative and quantitative conditions in the following sections.

2.3 Uncertainties, Probabilities, Belief and Possible Worlds

One of the main causes of confusion in defining risk is that the meaning of *uncertainty* is typically ill-defined. Determining risk in any domain requires the formalisation of uncertainty and it could be different in different domains. Therefore, I need to allow for adoption of a number of concrete forms of uncertainty depending on the specific domain and context. For example, in risk management of foreign exchange, there is abundant historical trading data and related quantitative information from which one can carry out statistical analyses to determine a model for uncertainty based on the theory of probability. In this case, probability is a suitable *representation* for uncertainty since it is a faithful computational model for statistical data and can be derived relatively easily from the large amount of quantitative data available. In other domains such as investment in a middle eastern country like Iran (Waring & Glendon 1998), building a risk profile for a country involves many complex factors such as political stability, historical background that are not easily quantifiable and

[3]In common usage, risks are usually associated with the negative consequences.

information for these factors is typically based on the opinions from so-called experts in the field, and as thus, may be inherently biased. Therefore, it is difficult to formulate appropriate probabilities and I need to use alternative representations for uncertainty. Furthermore, I need methods to capture and express uncertainty in a domain for which an agent only has qualitative and/or categorical information. Since one of the key requirements for the risk modelling and management framework is adaptability in many disparate domains, I need to accommodate both qualitative and quantitative domain information. Consequently, my generalised definition of risk should not constrain the key notion of uncertainty under one particular form of representation. One should choose an appropriate form of representation when he (or she) applies HiRMA to a specific problem domain. In order to provide an unambiguous response to the fundamental question of "what is uncertainty", I invoke the concept of *possible worlds* as an elegant philosophical foundation upon which to represent or encode uncertainty at different levels of abstraction. More importantly, a possible worlds approach also provides a common foundation from which I can build a pathway to 'link' different uncertainty representations.

2.3.1 Possible Worlds Paradigm

Possible worlds or alternative universes are those worlds whose characteristics are different from our own. The Possible Worlds paradigm is a world view in which there are alternative plausibilities beside the world *as we know it*. There could be worlds where anything that one could imagine is true as long as each world remains logically consistent. Formally, possible worlds are *models* or knowledge bases (sets of propositions or sentences) that describe possible worlds (Stalnaker 1976). The Possible Worlds paradigm plays a crucial role in the development of modal logics (Blackburn, de Rijke & Venema 2002)[4] and provides an intuitive interpretation of *belief*. That is, if an agent *believes* a proposition A, it means that the agent believes

[4]Modal logic is beyond the scope of this thesis and I will not discuss this topic further apart from providing few references.

A holds in all the worlds the agent can view as possible. Uncertainty arises when an agent is unable to determine which world from all possible worlds (that are consistent with the agent's belief) is the actual world that the agent inhabits. Uncertainty reflects the fact that the agent has incomplete knowledge (or incorrect) information about the actual world. However, it does not mean that the agent has no knowledge in regard to the uncertainty. We can, in fact, encode such knowledge into the relational structure of the possible worlds.

2.3.2 Encoding Uncertainty in a Likelihood Order

Based on an agent's existing knowledge, we can map a set of possible worlds into a preorder $\{\leq : w \in \Omega\}$, where Ω is the set of possible worlds and $\leq$ is a reflective and transitive relation over Ω. We can interpret this comparative relation as: if $u \leq v$, u is a world that is at least as plausible as world v. In addition, the preorder $\leq$ must have at least one world w such that $w \leq v$ for all $v \neq w$. This *smoothness condition* ensures that there is at least one most plausible candidate that corresponds to the real world. With this likelihood order structure, an agent can make a qualitative assessment of uncertainty within a domain according to its current knowledge. For example, the belief of logical entailment from proposition D to propositions $S1$ and $S4$ can be represented as a simple preorder structure of possible worlds as shown graphically in Figure. 2.1, in which every circle represents a possible world. We use $\rightarrow$ to represent a logical entailment and $\nrightarrow$ to represent its negation.

In this example, world $w4$ in which D entails both $S1$ and $S4$ is less plausible than other worlds that D only entails $S1$ or $S4$ or neither[5]. Even though an agent is still uncertain which world corresponds to the actual world, it believes, according to its current knowledge, $w1$ is more plausible than $w2$; $w2$ is more plausible than $w3$ and $w3$ is more plausible than $w4$. This likelihood preorder gives us a qualitative representation of the uncertainty in the domain. An additional advantage that

[5]The reason for using smaller circles to represent higher plausible worlds will become clear in Chapter 6 in which I map likelihood preorder structures into Systems of Spheres.

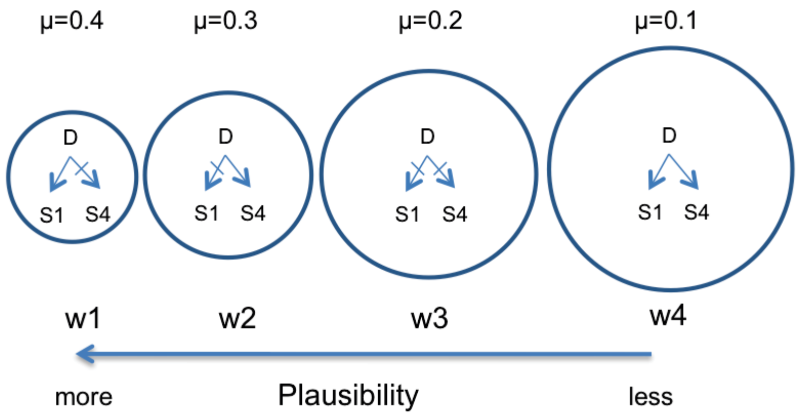


Figure 2.1: Possible worlds in a likelihood preorder.

this preorder structure brings is that it is also conducive to systematic revision of the knowledge base. When new information is acquired, the existing preorder may be modified to reflect the agent's updated knowledge base and consequently a new assessment of the uncertainty can be derived. A mechanism such as Belief Revision (and update) (Gärdenfors 1992, Peppas 2007) provides a systematic approach to revise knowledge and maintaining this preorder. In addition, this likelihood preorder is equivalent to comparative possibility relation (Lewis 1973) and strongly tied to the qualitative possibility theory (Dubois & Prade 1998). We can, in fact, translate this likelihood order into a fuzzy set (Ω, m) in which every possible world is an element of the set Ω and the member function of the fuzzy set m maps Ω to $[0, 1]$ (Zadeh 1965). The likelihood of a possible world w is represented as the grade of membership of w in (Ω, m).

2.3.3 Encoding Uncertainty in Probability

We can now give a reinterpretation of probability from the possible worlds paradigm by associating each possible world with a *measure*, i.e. a real value. For a finite number of possible worlds, the measure[6] of world w, denoted as $\mu(w)$, has two properties:

- $0 \leq \mu(w)$ for all $w \in \Omega$.
- $\sum_{w \in \Omega} \mu(w) = 1$.

The probability of a formula a is the sum of the measures of all possible worlds in which a is true (i.e. $w \models a$). Formally, the probability of a is expressed as

$$P(a) = \sum_{w \models a} w. \tag{2.1}$$

It can be easily shown that this semantics of probability is consistent with the axioms of probability (DeGroot & Schervish 2002). One remaining question is how the measure of a possible world is determined. Intuitively, a more plausible world shall have higher probability measure than less plausible worlds, i.e. the measures of possible worlds reflect an agent's belief of the likelihood of all possible worlds. We can say the probability measure quantifies the preorder structure of the possible worlds. Using the previous example, we can assign measures to the four possible worlds (Figure. 2.1). By taking $D \rightarrow S1$ as a formula a and following equation 2.1, we have the probability of D *entails* $S1$ is 0.5.

I conclude Section 2.3 by highlighting the following:

- Uncertainty is an integral part of risk and it can be represented in many forms using the Possible Worlds paradigm. Probability is one of such representations.
- My risk modelling and management framework requires appropriate representations for uncertainty with respect to the types of domain knowledge available.

[6]This measure is same as probability measure.

- The Possible Worlds paradigm provides a solid philosophical foundation for modelling knowledge of uncertainty both qualitatively and quantitatively.

2.4 Measuring Consequence

I now turn my attention to *consequence*. According to ISO 31000 (ISO 2009a), consequence is anything that deviates from the desired outcome. From the perspective of an intelligent system, the desired outcomes are in direct correspondence with the designed objectives of the system. Any deviation from the desired outcome that is beneficial for achieving the system objectives is considered to be positive (payoff), whereas deviations that have adverse impact on achieving the objectives are considered to be negative (penalty). Measuring the extent of deviations is dependent on whether the system objectives and outcomes are quantifiable. I treat quantitative consequence and qualitative consequence separately, as different modelling mechanisms can be applied and used.

2.4.1 Qualitative Consequence

In some problem domains, consequence cannot be easily quantified. For example, the current market condition perceived by individual investors cannot be fully expressed in numbers meaningfully. In these cases, we collect all plausible scenarios, including the desired outcome, and arrange them in a preference order $\{\leq : s \in S\}$, where S is the set of all plausible scenarios. $\alpha \leq \beta$ means scenario β is preferred over α. Furthermore, if d is the desired outcome and $\alpha \leq d \leq \beta$, it means that α is a scenario of negative consequence; whereas β is a scenario of positive consequence. This preferential order mirrors the similar approach used in encoding uncertainty in a likelihood order. It can also be converted into a fuzzy set (S, c) in which the member function c grades all possible scenarios. With reference against the desired outcome, c can be further mapped into a set of common categorical consequences, for example,

$\{Severe, Bad, Minor, Good, Excellent\}$.

2.4.2 Quantitative Consequence

For a task domain with quantifiable outcomes, consequences can be measured, in real values, as the "distances" between the expected outcomes and actual results. For example, all financial outcomes can be monetized. Therefore, consequences can be naturally measured in terms of dollars. Any loss is normally treated as negative consequence and takes on a negative value while any financial gain is positive consequence associated with a positive value. In this case, we assume a financial neutral position is the expected outcome and is designated with a reference value of zero. This is not always the case as the expected outcome may not be the financially neutral position and we may assign a different reference value to the expected outcome. Non-zero reference values are useful in situations that the actual outcome, while it does not match with the expected outcome, partially achieve the objective or still remains useful. Formally, I define a quantitative consequence as the following:

$$C_s = O_s - O_d, \tag{2.2}$$

where O_d is a reference value assigned to the desired outcome and O_s is a value assigned to a scenario scaled against the reference value.

2.5 Combining Uncertainty and Consequence

Armed with the formal interpretation of uncertainty at both qualitative and quantitative levels and how consequence may be evaluated and measured, I can now investigate how risk may be described as the combination of uncertainty and consequence at these two levels. I assume each plausible scenario (or scenarios[7]), i.e. a resultant event from an initial event, has a specific consequence for an agent tied directly with

[7] We may have multiple scenarios occur simultaneously. In this case, we can summarise them into a compound scenario.

it. Different scenarios may lead to the same consequence for an agent. The same scenario may lead to different consequence for different agents. However, I do not consider the case in which a scenario may have many consequences.

2.5.1 Risk Matrix for Qualitative Risk Representation

Risk matrices are a common technique used in risk assessment to represent risk (Aven 2008, Waring & Glendon 1998). It is usually presented in a table form, as shown in Table 2.1, with columns representing categorical likelihood or probability bands and rows representing categorical consequences. With the formal qualitative interpretation of uncertainty discussed in Section 2.3, we can replace classical probability bands found in most of risk management literature with a preorder structure of possible worlds (or an equivalent fuzzy set derived from the comparative possibility relation of possible worlds). A conventional probability band or likelihood category can be regarded as a possible world or a set of consecutive possible worlds with similar plausibilities in the preorder structure (Table 2.2). This small but significant modification of the risk matrix not only removes the "ad hoc-ness" in the technique enforcing a natural order in rows of risk matrix table, but also enriches it by placing the corresponding domain model (from risk analysis) directly in the table. Assuming each domain model is dominated with one particular scenario, there is no need for a secondary mechanism to associate the scenarios with its risk categorisation. One possible drawback of this modified risk matrix is that a large number of possible worlds and possible consequences may make the risk matrix table large, unwieldy and difficult to read for people. However, intelligent agents or information systems can store and access them in large databases. Note that $C1$ ... $C4$ in Table 2.1 and 2.2 are generic terms for qualitative consequence labels.

To compare an arbitrary pair of risks in a risk matrix, e.g. (R_1, R_4) in Table 2.2, we first assign ordinal numbers to rows ($Ordinal_c$) and columns ($Ordinal_w$) of the matrix according to the likelihood order and preference order respectively with 1 attached to the least plausible world and consequence of lowest preference. A comparative

Consequence	**Likelihood**			
	Rare	Unlikely	Possible	Likely
$C1$	*			
$C2$		*		
$C3$			*	
$C4$			*	

Table 2.1: A conventional risk matrix.

Consequence	**World** w $\{\leq: w \in \Omega\}$			
	$w1$	$w2$	$w3$	$w4$
$C1$	R_1			
$C2$		R_2		
$C3$			R_3	
$C4$			R_4	

Table 2.2: A risk matrix with possible world preorder structure.

risk ranking can thus be calculated from $Ordinal_c \times Ordinal_w$ associated with the risk. Therefore, R_4 is more favourable compared with R_1. Note that, a colour coding scheme is normally used in risk management literature instead of this "quantification" procedure, though the underlying semantics are the same.

2.5.2 Quantitative Risk Measure

Risk needs to be quantified in many domains. We need a quantitative measure for representing risk. In mainstream risk management literature (Aven 2008, Waring & Glendon 1998), quantitative risk is usually represented using expected probability (or value) given by the summation of the products of probabilities of the scenarios and their quantified consequences. However, this approach does not take account of the risk attitude of a respective stakeholder. This missing dimension may have significant impact on risk analysis of a target domain with multiple stakeholders. For example, a company may have a number of shareholders with various interests in the company. Consequences due the operation of the company will have different meaning and impacts for individual shareholders. Therefore, different stakeholders may have different risk attitudes towards the operational and strategic outcomes of the company. My risk analysis and the measure for risk should reflect this risk attitude. To this end, I adopt *expected utility* for quantifying risk.

2.5.2.1 Expected Utility

An ***Expected Utility***(EU) (Savage 1954) is commonly used in finance[8] to provide a quantitative expression of the uncertain prospect of rewards and an investor's attitude towards the 'risks' of such rewards (Tapiero 2004). It can be expressed in a general form as follows:

$$EU = \sum_{j=1}^{n} P_j u(c_j) \tag{2.3}$$

where P_j is the subjective probability of an event j, c_j is the consequence of event j, u is the utility function and n is number of possible events. In this thesis, I use the simple lottery form of the expected utility which considers only one event with two possible outcomes, i.e. the event occurs and the event does not occur. Consequently, a simplified version of Equation 2.3 can be given as:

$$EU = Pu(c_1) + (1 - P)u(c_2), \tag{2.4}$$

where we have two possible quantitative consequences c_1 and c_2. The use of this simple lottery form greatly simplifies the risk metric system. More importantly, it is a reasonable simplification since complex events of more than two event outcomes (compound lottery form) can be reduced into this simple form by combining the probabilities of event outcomes (Fishburn 1970). In other words, a simple lottery model can be used to model arbitrarily complex compound lottery models.

2.5.2.2 Probability-EU Space

When we render expected utility to null, i.e. $EU = 0$, it means that an agent (as a stakeholder) is indifferent towards either results of the event. In other words, neither

[8] More generally, EU is used in decision making under uncertainty.

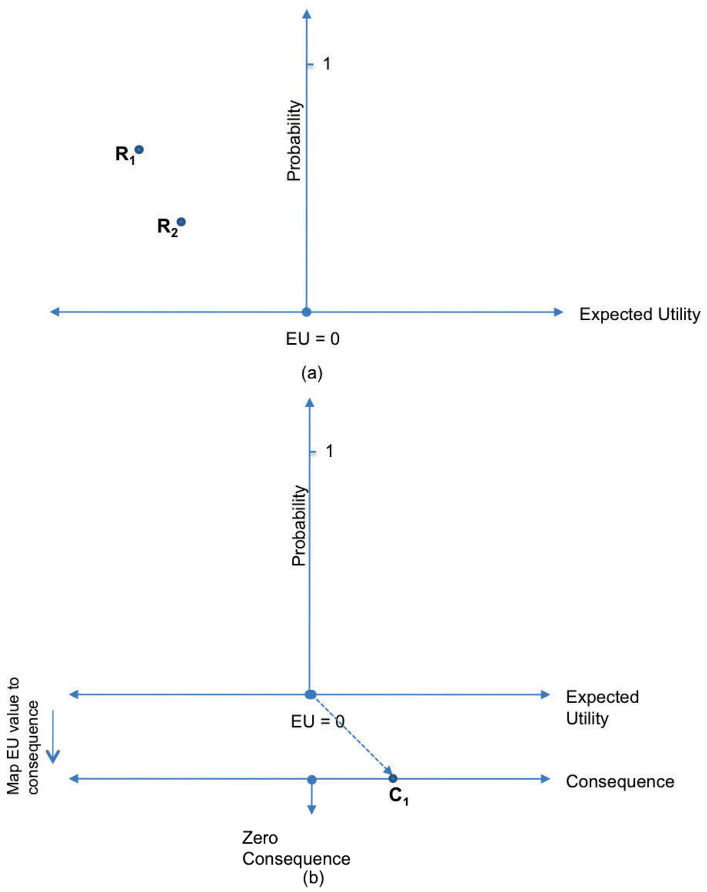


Figure 2.2: Probability-EU space and Probability-Consequence space.

results pose as a 'risk' towards the agent. Therefore, we have a point that effectively represents a 'zero' risk point. It connects probability of an event and its utility function. We can construct a two-dimensional Probability-EU space (Figure 2.2) in which we place *risk* data point R. The *"distance vector"* from the origin to R provides an effective measure of risk for a stakeholder. We can therefore make comparison of various risks. Furthermore, with the addition of a third axis *stakeholder*, it is possible to study the perceived impact of the same risk for different stakeholders. We define an *effective risk measure* ER^2 as following:

$$ER^2 = sin(2 \cdot atan(EU/P)) \cdot (EU^2 + P^2), \tag{2.5}$$

where $sin(2 \cdot atan(EU/P))$ is a normalisation factor based on the distance vector angle with respect to EU axis. This factor ensure the effective risk measure goes to zero when we approach either P or EU axis. Since we can rearrange equation 2.4 as following:

$$EU = K \cdot P + u(c_2) \tag{2.6}$$

where $K = u(c_1) - u(c_2)$, we can now calculate ER^2 with known probability P and utility function u:

$$ER^2 = sin(2 \cdot atan(K + u(c_2)/P)) \cdot ((K \cdot P + u(c_2))^2 + P^2). \tag{2.7}$$

Relationship between Probability, Consequence and EU

If the utility function is known and one of the consequences (say c_2) is also known and fixed (i.e. the expected result inline with the objective), we can work out c_1 from the probability P and EU. That is, we can map an EU value into a consequence value with the utility function. This means, we can essentially use the utility function as a translation function to convert the Probability-EU (P-EU) space into Probability-Consequence (P-C) space (Figure 2.2) which has been used in traditional risk assessment (Aven 2008). To ensure the validity of Equation 2.6, the following constraints have to be applied to the equation:

$$u(c_2) - u(c_1) \neq 0. \tag{2.8}$$

This means we have to assume the utility for different consequences must be different. An interesting observation from this translation from P-EU space to P-C space is that zero consequence does not (necessarily) translate to zero EU value. This means it is possible to model a situation in which an agent does not satisfy with the neutral result of an event (i.e. consider the effort is ineffective); or a situation in which the negative consequence of an event may not be a bad thing for an agent. For example, a badly damaged car from a fire may mean an insurance company has to replace it with a brand new vehicle for the owner who holds an insurance policy, i.e. a beneficial outcome for the car owner; at the same time it also means the insurance company will suffer some financial losses.

2.5.3 Application of the Risk Measure

The Probability-EU/Probability-Consequence space and the corresponding risk measure provide a simple system for evaluating risks for many problem domains. In order to apply this risk measure to a specific domain, we need to first define the semantics of *consequence* for the domain and ensure the corresponding consequences are quantified (see Section 2.4.2). We would also need an effective utility function to calibrate the Probability-Consequence space. Both tasks are non-trivial and require detailed study and analysis of the domain. Interestingly, some domains such as in finance, have already developed their own risk measure, namely Value at Risk (VaR). I will demonstrate that VaR fits reasonably well within the Probability-EU/Probability-Consequence space settings.

2.5.3.1 Value at Risk

The concept of *Value at Risk* (RiskMetrics 1996) is widely used in the financial industry as a way of measuring risk. *VaR* is defined as loss of market value, over a time period of T, that exceeds probability of $1 - P_{VaR}$. It is a product of the total investment and probability of (predefined) maximum tolerable loss (P_{VaR}).

$$VaR = I_{total} * P_{VaR} \tag{2.9}$$

where I_{total} is the total investment.

VaR is applied to various types of financial risks such as interest-rate risk, price risk of titles, credit risk and exchange risk. VaR is most notably used in the calculation of capital reserve in commercial banks. For example, in foreign exchanges, a commercial bank has a position of one million Australian dollars. With volatility in AUS/USD exchange rate, if probability of 5% depreciation in Australian dollar (over a day) is 3.4%. Then, as a result, the VaR is

$$VaR_{AUS} = 1000000AUS * 0.034 = 34000AUS.$$

We can directly cast VaR as the risk measure R into the Probability-Consequence space with associated probability as P_{VaR} on the probability axis (Figure 2.3). From equation 2.9, we can deduce following with the consequence axis, denoted as X, quantified in dollar.:

$$\begin{aligned}
VaR^2 &= I_{total}^2 * P_{VaR}^2 = P_{VaR}^2 + X^2 \\
X^2 &= I_{total}^2 * P_{VaR}^2 - P_{VaR}^2 \\
X^2 &= (I_{total}^2 - 1) * P_{VaR}^2 \\
\text{since } & I_{total}^2 \gg 1 \\
X^2 &\simeq I_{total}^2 * P_{VaR}^2 \\
X &\simeq VaR \qquad (2.10)
\end{aligned}$$

Equation 2.10 shows that with total investment I_{total} much greater than 1, VaR is asymptotically aligned with the consequence axis. This is consistent with that both risk and consequences in the finance industry is measured by monetary values. VaR can be used to act as a risk constraint in terms of risk management. With known total investment and maximum tolerated loss, i.e. VaR, we can determine the constraints we need to place on the probability of loss of investment. If the probability of such a loss is greater, we might reject the investment.

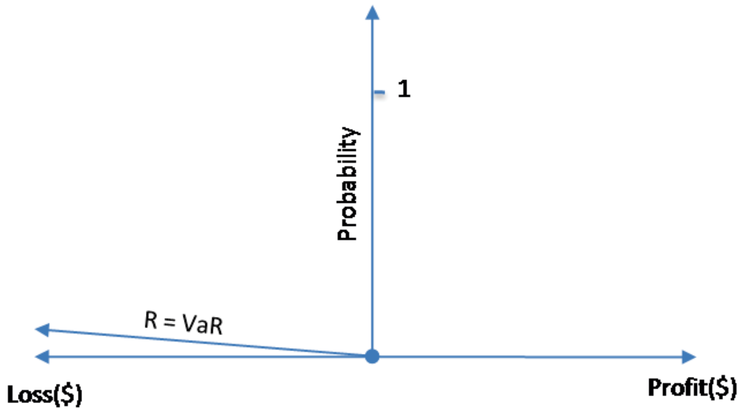


Figure 2.3: VaR in Probability-Consequence space.

2.6 Summary

Risk as a general notion is part of everyday discussion. Analysing, assessing and managing risks is an integral part of decision making process in many domains. However, the definition of risk varies from domain to domain even though the common theme remains the same. The objective of this chapter is to develop a formal conceptualisation of risk from the perspective of an intelligent agent. This work provides a solid foundation upon which a generalised risk modelling and management framework can be developed for intelligent agents. I surveyed the existing risk definitions from a range of practical problem domains and analysed the essential elements within the notion of risk, namely uncertainty and consequence. Based on these two general concepts, I presented a top-level definition of risk tailored for intelligent agents. However, in order to adopt this generalised risk definition for agents operating in specific fields, I need to formalise the concept of uncertainty and consequence and concrete forms of representation for both concepts and, thereof, risk.

I provided a formal interpretation of uncertainty based on the Possible Worlds paradigm. I represent uncertainty qualitatively in a preorder structure of possible worlds and represent quantitative uncertainty with probability. I highlighted that probability is one of many forms of representation for uncertainty. Mainstream risk management literature distinguishes the difference between uncertainty and probability. However, they do not provide any alternative representations for uncertainty apart from probability. In fact, probability (bands) is continually being used in risk matrices as *the* representation for qualitative uncertainty. I interpreted the concept of consequence as deviations from the objectives of an agent. The interpretation is inline with the similar interpretation found in ISO 31000. Consequence can be represented qualitatively through a comparative preference order of all possible scenario with agent's objective as the referential element; quantitatively using the numerical distances between the objective and the scenarios.

In order to accurately represent risks in agents, I provided a modified version of risk matrix using preorder structure of possible worlds which eliminates the ad hoc-ness in conventional risk matrices. Furthermore, my modified risk matrix have domain models and associated scenarios directly presented in the table so that we have results from risk analysis captured in one visual representation. This gives us a full picture of the current domain knowledge in relation to risk. On the quantitative side, I presented an alternative measure for risk based on expected utility. Unlike traditional risk measures with expected value (probability), my risk measure takes account of the risk attitudes of stakeholders using utility functions. This enables us to accurately represent risk with respect to individual stakeholders. Even though EU has been utilised in risk evaluation (Charette 1989, Tapiero 2004, Garvey 2008), my approach differs from the conventional decision theory approach used previously. I construct a more intuitive risk measure from a Probability-Expected Utility (P-EU) space that references on null EU value. The construction of P-EU space does not rely on any particular type of utility function. P-EU space can be easily translated into Probability-Consequence (P-C) space which is commonly used to represent expected value for risk. This translation depends on the utility function of stakeholders which could be challenging task to model. Finally, I showed the risk measure developed in the financial industry, namely Value at Risk (VaR) is consistent with P-EU/P-C space setting.

In summary, I have analysed and constructed a general definition of risk from fundamental concepts. I have taken a holistic view of risk on both qualitative and quantitative levels. My risk definition is based upon the well understood theories of possible worlds and expected utility that have numerous applications in logic and decision theories. This formal conceptualisation of risk forms the basis of HiRMA framework. Furthermore, the qualitative risk representation and numeric risk measure developed in this chapter can be utilised in assessing risks.

Chapter 3

Requirements for a Generalised Risk Management Framework

From the conceptual analysis of risk in Chapter 2, the concept of risk and its associated notions are highly dependent on task domain, its environment and agents' objectives in question. This is the main reason that risk cannot be modelled and represented accurately and precisely without a detailed analysis of the domain and its context from the perspective of the key stakeholder, i.e. agent. Therefore, I need to analyse and model the task domain and agent's objectives systematically through a stepwise knowledge engineering process, in order to assess and manage risks. We cannot discuss and develop a risk management framework for intelligent agents without concrete examples. In this chapter, I will first present two benchmark problems from two disparate fields: autonomous mobile robot soccer player and managing foreign (currency) exchange (FX) in small-medium companies. They will be used to discuss some common characteristics found in many real-world domain environments and the challenges they bring with respect to risk modelling and management. I will set out a number of key requirements for the generalised risk management framework to address the challenging issues raised from the discussion so that the framework can be applied to a wide range of domains. I will evaluate some of the most common risk management methodologies in mainstream literature against these requirements and demonstrate the need for a new solution.

3.1 Benchmark Problems for Risk Modelling

3.1.1 Benchmark Problem 1 - Ball Passing Problem

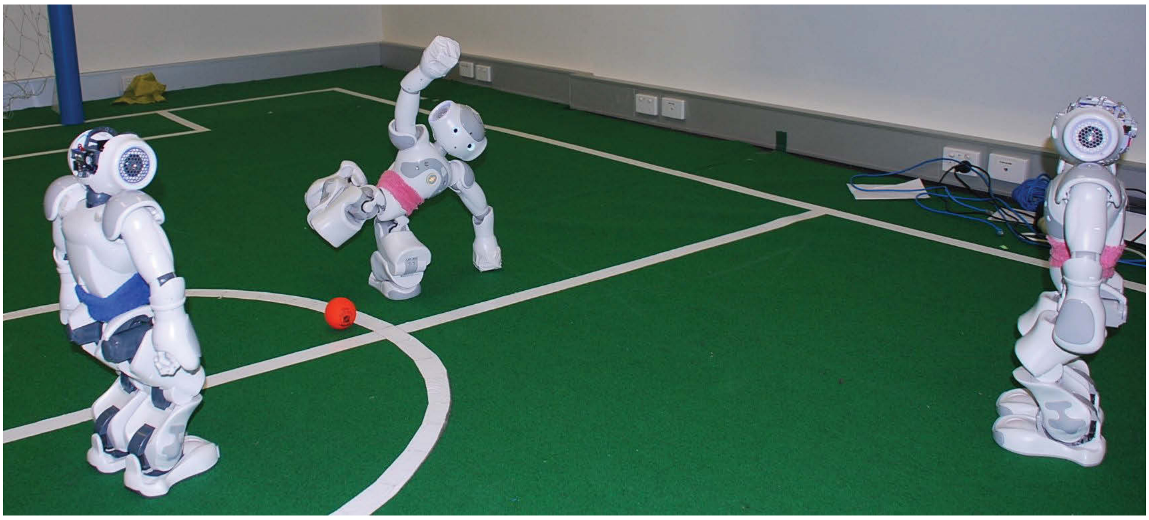


Figure 3.1: Ball passing between robots

For many years, RoboCup[1] has been one of driving forces advancing and application of theoretical ideas in AI to real world problems and has successfully pushed the boundaries of the research field (Kitano 1998, Takahashi, Edazawa, Noma & Asada 2006, Seekircher, Laue & Röfer 2011). Robots are decision making agents and must respond to changes in a fast moving game environment. Uncertainty is high with each event during a game. There are many challenges with complex consequences. One of key challenges in robot soccer matches is ball passing between two robot team mates (Figure 3.1). There is still little deliberate ball passing between robots after many years of competition. The few successful ball passing events that occurred in competitions were usually unintentional coincidences. The ball passing

[1]www.robocup.org

problem, despite its small scope, presents a rich scenario which enables the exploration and analysis of various risk factors and events involved in passing a ball from one robot to another in a game situation; the development of risk models of increasing sophistication to handle more aspects of the problem. The Ball Passing Problem is an excellent benchmark problem because it is a challenging real world problem where empirical data and results can be collected and the performance of sophisticated risk modelling methods can be examined, compared, tested, and evaluated. I give a more detailed description for the Ball Passing Problem in the following section

Problem Description

A standard match of RoboCup standard league consists of two opposing teams each with four NAO[2] robots. During the game play, a NAO robot R_a from the red team is in control of the soccer ball and it has an opportunity to pass the ball to one of its team members R_b. There is some distance between robot R_a and R_b. R_a can only kick the ball to a certain distance. A NAO from the opposing team is closing in to intercept the soccer ball. R_a can either pass the ball to R_b or take no such an action. If R_a decides to pass the ball to R_b, we may have R_b takes the possession of the ball; ball may be intercepted or the ball simply does not reach R_b. Many factors could affect the result of passing the ball. All robots on the field must follow the rules stipulated in the RoboCup Standard Platform Rule Book (2011).

3.1.2 Benchmark Problem 2 - Foreign Exchange (FX)

The Australian dollar is one of the most traded currencies on the open foreign exchange market. It is also one of the most volatile currencies in the market. It is crucial for many companies in the business of importing/exporting goods in Australia to manage their foreign exchange exposures carefully in order to avert/minimise possible loss due to fluctuations in the exchange rate (Abbott 2009). However, a domain such

[2]NAO is the standard robot platform used in the current RoboCup standard platform league. Check www.aldebaran-robotics.com for details.

as the foreign exchange market is extremely complex and dynamic. There are many macroeconomic and microeconomic factors influencing the Australian dollar exchange rate. Therefore, modelling and managing the risks is difficult task. For our purpose, I consider a simple risk modelling scenario from the perspective of a medium size importer of electronic goods.

Problem Description

A medium size importer of electronic goods wants to minimise its risk exposure due to fluctuations in foreign exchange markets. The objective is to maintain a neutral FX position and no significant financial loss due to changes in Australia-US dollar exchange ratio. The importer regularly orders large batches of electronic goods directly from manufactures located overseas. Orders normally take one to two months to be fulfilled and shipment of goods takes two weeks. The importer has options to make payment in full at the beginning or in several instalments during the entire business transactions. In order to keep a neutral FX position, the importer has the options of currency hedging; maintain a large foreign currency reserve; or do nothing. Financial gain or loss may occur due to fluctuations of Australia dollar during the business transactions period.

3.2 Framework Requirements

The two benchmark problems described in the previous section highlight some of the major practical issues that my risk modelling and management framework must address; the framework must be able to accurately capture and model the relevant knowledge (in relation to risk) of a problem domain in a real-world environment. Recall that the goal of this research is to adapt, extend and integrate well established AI theories and techniques used in modelling uncertainty as a means to enable intelligent agents to reach effective and optimal decisions that take account of domain risks.

Before I can embark on the development of the framework, I need to identify and analyse some aspects of the intrinsic nature of real-world domains and environments to which my framework will be applied. A closer look at the benchmark problems allows us to observe three critical and challenging features present in these practical domain environments:

- **Complexity**: Many risk factors (or variables) are involved and interrelated in the task domain. Some variables are quantitative in nature; whereas other variables are qualitative and cannot be quantified appropriately. In the relatively simple robot soccer match domain environment, we have quantitative variables such as kicking distance between the robots and kicking power of a robot; other variables such as the ability of ball interception of opposing robot team are difficult to measure and quantify[3]; whereas impacts of some of the soccer match rules are purely qualitative. For example, the illegal defender rule (rule 4.13) in RoboCup Standard Platform Rules Book (2011).

- **Openness**: In practice, we may not know all of the possible variables involved in the domain, and we may not have sufficient information in relation to known variables. With regard to foreign exchange, there are many hidden variables in global currency exchange markets that are unknown to a large number of market participants (Lyons 2001). For a small-medium sized enterprise trading goods internationally, it has much less information and historical data of exchange rates in comparison to large financial institutions that specialise in currency trading. This means that we cannot assume we have a complete information about the domain when we develop the risk model for any real world application.

- **Dynamics**: The domain environments evolve and change over time. Relationships between variables may evolve and the number of relevant variables in the domain may also change. Information available often changes and new information may contradict previous background knowledge. In the FX, the Australia

[3]Some readers may argue this variable might be measured and learned by repeatedly play matches with the opponent. In reality, such opportunities are very rare. In addition the opponent is not static either and its capabilities are constantly evolving and changing.

dollar is traditionally "tied" to resource exports and global economic outlook. The global economic environment is constantly evolving. Ten years ago, economic fortunes in China had little influence on the Australia dollar. In contrast, nowadays, current heavy investments in infrastructure in China have significant influence on the Australian exchange rate, due to the significant increases in commodity exports to China. Clearly, any risk models built ten years earlier need to be revised in accordance with the evolving environment.

A *practical* framework for risk analysis, knowledge capture and risk modelling must carefully address the challenges of complexity, openness and environmental dynamics. Therefore, I identify the following key design requirements for my risk management framework in correspondence to these challenges

1. The framework should provide a standard methodology for analysing and modelling knowledge in relation to risk for disparate domains from the perspective of an intelligent agent. This is the overarching requirement for the framework. More specifically,

2. The framework should make no specific assumption that complete knowledge is available for the task domain modelling; i.e. it is based on an "Open World Assumption"[4].

3. The framework should support intelligent agents that are able to continuously acquire and incorporate new domain knowledge over time.

4. The framework should handle both quantitative and qualitative domain knowledge.

5. The framework should capture causal relationships among the domain variables to ensure the stability of the risk model. The causally connected risk

[4]Open World Assumption is the assumption that the truth value of a statement is not known. It represents a notion that no single agent has the complete knowledge of a domain.

model generated from the framework can support the development of appropriate treatments to influence the desirable and undesirable variables in the system.

6. The framework should accommodate frequent update and revision of the existing knowledge base in order to reflect changes in the domain environment. In particular, the framework should capture the evolution of the causal structures of domain risk models as opposed to the changing (operational) states of the models.

3.3 Existing Risk Management Methodologies

Risk analysis and management has been actively studied for decades. In this section, I take a quick survey of several predominant methodologies that have been widely used in many fields. I summarise their features and evaluate them against the framework requirements discussed in the previous section.

3.3.1 Fault Tree Analysis (FTA)

Fault Tree Analysis (FTA) was first developed to analyse the reliability of control systems (Watson 1961). The method essentially translates a physical system into a structured logic diagram that consists of causes and one top event of interest connected largely with AND and OR logic gates. The top events of interest are undesired system states that resulted from some sub-system functional faults (event). They are usually generated from a preliminary hazard analysis as the first step in the Fault Tree Analysis. A functional layout of entire physical system is also produced in order to show all functioning components in the system and their interconnections. This forms the basis for fault tree construction. Furthermore, system boundary conditions are defined to specify what situation the fault tree is generated. Figure 3.2 shows a typical fault tree with a corresponding physical system layout.

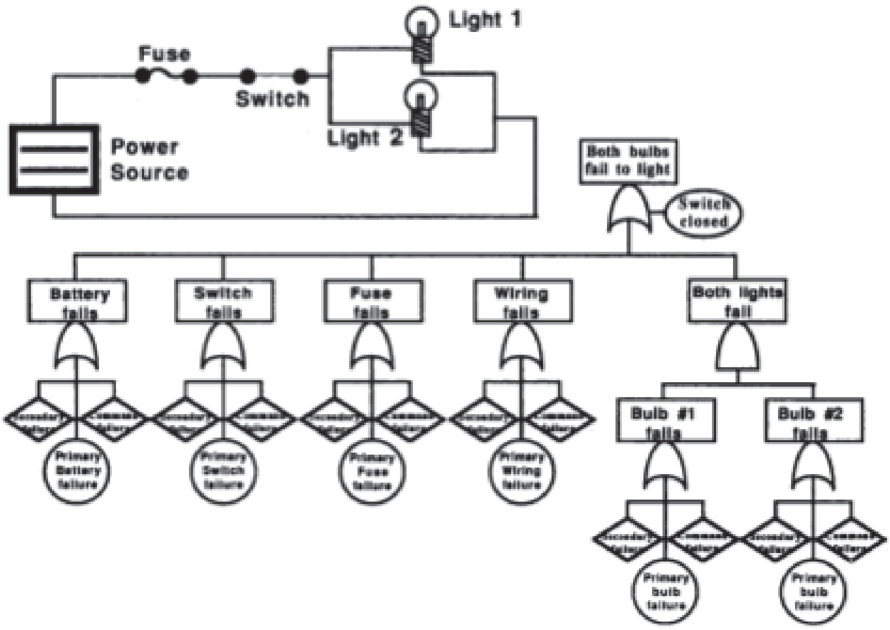


Figure 3.2: An example of a fault tree (Stephans 2005).

Fault tree construction is manual and laborious. There are numerous software tools available to assist the construction task (Lee, Grosh, Tillman & Lie 1985). Fault trees can be evaluated both qualitatively and quantitatively. Qualitative evaluation is based on expansion or reduction of the top event of a fault tree through boolean algebra to determine the minimal path sets and the common cause failures. Quantitative evaluation of fault trees extends the qualitative evaluation method and relies on the rate of occurrence, fault duration of all basic events and statistical dependency of basic events in order to determine the probability of the top event.

Fault tree analysis is a classic top down system failure analysis technique. The system must be fully analysed and modelled for proper fault tree construction. It is well suited for static systems that do not change over time. FTA cannot handle open-ended systems that require continuous modifications and extensions. Although

FTA (with extensions) can be used for quantitative evaluation, the technique is inherently qualitative and uses boolean algebra for its basic deductive operations, unlike formal probabilistic models rely on conditional probability and Bayesian Theorem. Therefore, FTA only partially meets the framework requirements.

3.3.2 Event Tree Analysis (ETA)

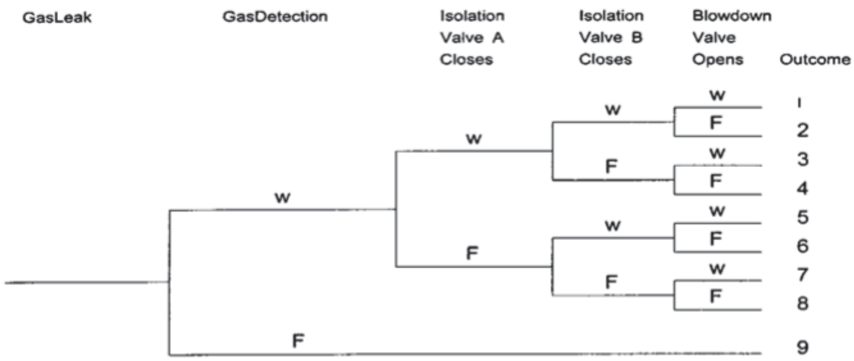


Figure 3.3: An example of an event tree (Andrews & Dunnett 2000).

Event Tree Analysis (ETA) is a bottom up deductive system analysis technique. ETA has a tree structure (Figure 3.3) similar to fault tree without binary logic gates. An event tree does not require a top event of interest as in FTA. From an initial event, an ETA (binary) branches out based on decisions or actions to trace out every permutation of the entire system operations. Each path from the initial event will end with a success or failure state. One of key advantages of ETA is that no single end event must be identified at the beginning of the analysis unlike FTA. However, every operational path of the system must be obtained through the analysis. ETA complements the FTA method in analysing and modelling static systems in which full knowledge can be obtained.

3.3.3 Failure Mode and Effects Analysis (FMEA)

Failure Mode and Effects Analysis (FMEA) is a simple inductive system analysis method to reveal possible failures in a system and provide prediction on the effects of the failures on the entire system. Similar to FTA and ETA, FMEA requires systematic analysis of components of a system, examining the criticality of their failures on the system. FMEA technique takes a tabular form as shown in Figure 3.4. It lists all the components in the system and describes the function of each component. The table identifies all possible ways that a component may fail and its effects on the other units within the system and the system itself.

SYSTEM/EQUIPMENT: Storage tank			EXECUTED BY: TAV					
REF. DIAGRAM/DRAWING.NO.:			DATE: 01.01.08		PAGE: 2		OF: 2	
Identification	Function/ operational state	Failure mode	Effect on other units in the system	Effect on the system	Corrective measures	Failure frequency	Failure effect ranking	Remarks
V2	Stop the supply when the liquid level is abnormally high. The valve is normally open	Does not close on signal		Undesired supply to the tank. The fluid is drained if V3 opens		2% of total number of demands	2	
		Closes when not intended		The fluid supply stops		Once in 10 years on average	1	
		Significant leakage		The fluid supply stops		Once in 10 years on average	1	
V3	Drain the fluid when the liquid level is abnormally high. The valve is normally closed	Does not open on signal		Undesired supply to the tank		2% of total number of demands	2	
		Opens when not intended		The fluid is drained		Once in 10 years on average	2	
		Significant leakage		The fluid supply stops. The fluid is drained		Once in 10 years on average	1,2	

Figure 3.4: An example of a FMEA table (Aven 2008).

The key advantage of FMEA is its simplicity. There is no additional graphical notation and operator. The tabular format provides a summary of an entire system. However, FMEA focuses only on system failure modes and does not model other undesired events. FMEA does not scale well with complex systems with large numbers of components and subunits[5]. FMEA also suffers the same limitation suitable for static and closed systems.

3.3.4 Probabilistic Models

The probabilistic (Bayesian) model has become a popular tool for risk modelling and management in recent years (Singpurwalla 2006, Kelly & Smith 2011). Unlike the previous methods that were born out of system analysis, probabilistic risk modelling was derived from the (conditional) probability theory. The method is best suited for quantitative risk analysis and modelling. Probabilistic models can also incorporate additional knowledge after their initial model construction. There are well developed techniques for model refinement so that risk models can be continuously improved based on new information. However, probabilistic models do not perform well for open systems for which there is no sufficient knowledge available for modelling. I will give a more in-depth analysis of this method in Section 4.4.3.

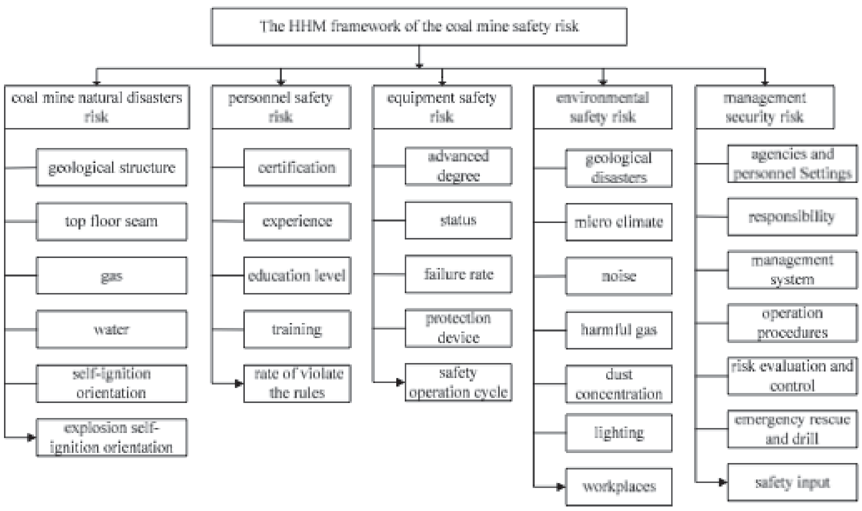


Figure 3.5: An example of a Hierarchical Holographic Model (Liu & Jiang 2012).

[5]For these systems, FTA gives a better modelling solution with its graphical representation.

3.3.5 Hierarchical Holographic Models

The Hierarchical Holographic Model (HHM) (Haimes 1981) was designed for system analysis and modelling for large scale complex systems. The method decomposes a system into multilayered sub models that connect together through overlapping objectives, constraints and input-output systems. HHM provides a holistic view of the modelled system and address different aspects of the system via separate sub models. Each sub model can be either qualitative or quantitative. HHM also provides a natural correspondence with complex systems hierarchical objectives and sub objectives. Due to its complexity, HHM has only been adopted in a few systems with multi-facet sub systems (Boudreau, Davis, Delery, Korbich, Lambert, Vogel, Tawney & Bennett 2005, Blais, Henry, Lilley, Pan, Grimes & Haimes 2009) and there is no technique in dealing with open systems.

3.3.6 Dealing with Uncertainty under Partial Ignorance

In recent years, risk analysis practitioners are increasingly aware that quantitative risk analysis cannot fully rely on (single) probability in dealing with uncertainty (Guyonnet, Come, Perrochet & Parriaux 1999, Helton, Johnson & Oberkampf 2004, Colyvan 2008). Uncertainty can also arise from the incomplete knowledge acquired from domain experts due to their partial ignorances. Dubois et al. (2010, 2011) gave a comprehensive account of how to represent and propagate uncertain information more accurately using various techniques such as fuzzy interval and upper and lower bound probabilities. This helps to improve the accuracy of an existing risk model in dealing with uncertain inputs. However, the construction of the risk model itself may be derived from the incomplete knowledge of experts. There is no well known techniques in dealing with uncertain information for risk model construction.

Methodology	Feature Requirements				
	Openness	Continuous Knowledge Inputs	Iterative Model Revision	Qualitative + Quantitative	Causally Stable Model
Fault Tree				$\surd^*$	$\surd$
Event Tree				$\surd^*$	$\surd$
FMEA				$\surd^*$	$\surd$
Probabilistic		$\surd$	$\surd$		$\surd$
HHM				$\surd$	$\surd$

Note:
FMEA - Failure Mode and Effects Analysis.
HHM - Hierarchical Holographic Model.
$\surd^*$ - Only supports these features through extensions.

Table 3.1: A feature comparison of the existing risk management methodologies.

3.3.7 A Comparison with the Framework Requirements

Most of above described techniques for risk analysis and management were developed from system analysis. They are designed for closed and static systems, i.e. all attributes of the system are known, and do not change over time. All above methods supports both qualitative and quantitative analysis and evaluation. However, apart from probabilistic model and HHM, the rest are qualitative in nature, i.e. they were not initially designed for quantitative evaluation and were not based on a formal theory of probability. Hierarchical Holographic Model is the only (hybrid) method that can build system models with both qualitative and quantitative sub models. However, its complexity causes to few adoptions in real-world environments. Finally, apart from probabilistic models, all existing techniques do not have built-in mechanisms to handle model revisions upon domain changes. They do not rely on knowledge management techniques (developed in AI) that can be easily implemented in software agents. There is also no formalised and theoretically sound transformation mechanism to convert a qualitative model to a quantitative model, and vice versa. Table

3.1 gives a summary of surveyed risk techniques with respect to the framework requirements stated in Section 3.2. I conclude this chapter with a view that no mature risk management technique currently available fully satisfies with the requirements. A new generalised risk management solution for intelligent agents is required.

Chapter 4

Knowledge Representations and Management in Complex and Uncertain Environments

The central theme of this thesis is the application of existing AI theoretical works in the areas of knowledge representation and management for risk management in intelligent agents operating in complex and uncertain environments. In this chapter, I will review a number of prominent and mature knowledge modelling and uncertainty management techniques. I will review the key features of these techniques and discuss their merits and shortcomings with respect with the key framework requirements discussed in the previous chapter. Most of the techniques discussed in details here will be incorporated into my final risk management framework for intelligent agent. Therefore, the literature review presented this chapter also serves as a reference material that provides sufficient theoretical background knowledge for system developers who implement the risk modelling and management framework in software. For readers who understand these techniques, it is safe to bypass the main descriptive parts of the chapter content and jump straight to the final discussion section of each technique labelled with '*'.

One of the critical requirements for modelling risk is the ability to capture and represent relevant domain knowledge in relation to the risk. Without the appropriate representation for the existing knowledge about risk, we have no basis for automated reasoning and management of risks. In the first section, I provide a brief review of classical logics, specifically first and second order logic, that have long been used to accurately represent static information and knowledge through sets of predefined systems since the beginning of AI. However these techniques do not work well when we have an incomplete picture of the domain in question; and give rise to the classical *frame problem* (McCarthy & Hayes 1969) in a changed domain environment due to an action. A number of solutions to these problems are offered by the class of so-called non-monotonic logics. Non-monotonic logics are constructs, based on the classical logics, that can support non-montonic reasoning. That is, they can provide tentative logical conclusions based on existing knowledge. However, the conclusions may be withdrawn or changed in light of new information. I review the classical default logic, autoepistemic logic and circumscription in the second section. These techniques all suffer from a common problem that of high computational complexity, particularly in face of changing domain knowledge that requires updates in defaults, or expansion/extension[1]; and in determining whether a formula is true in all minimal models in circumscription.

In the third section, I discuss an alternative class of non-monotonic formalism, Belief Revision(BR) under the so-called AGM postulates. It provides a logic based mechanism for integrating new information with the existing domain knowledge while maintaining the overall consistency of the knowledge base. AGM based BR provides a better solution in terms of incorporating inconsistent qualitative domain knowledge under an open world environment in comparison with the previously discussed methods. Furthermore, AGM based BR can be subsumed under qualitative part of possibility theory discussed in Section 4.6. I utilise this method for qualitative risk modelling due to its intuitiveness and simplicity.

All above methods are not designed to capture and model quantitative uncertainty. I give a detailed discussion on several prominent techniques such as Bayesian-based

[1]This happens when newly required information is in conflict with the existing domain knowledge.

probabilistic model and Transferable Belief Model developed for modelling and managing numerical uncertainty in Section 4.4, 4.5 and 4.5.11. These techniques will play an important role in the quantitative risk modelling portion of the HiRMA framework. Finally, the possibility theory discussed in Section 4.6 provides an elegant theoretical framework that can manage uncertainty with both qualitative and quantitative knowledge. Even though the HiRMA framework does not use possibility theory explicitly, the theory provides the crucial underlying support for HiRMA.

At the end of this chapter, I review two disparate approaches in capture causal inference knowledge from subjective opinions of domain experts and from available statistical domain knowledge. The first approach uses a logic based reasoning test to capture experts' *belief* in the causal inference relations within a domain; whereas, the second approach distills the *factual* causal inference information of a domain based on the structural analysis of a statistical Bayesian network of the domain. Both approaches play a critical role in the construction of stable causally connected risk models in Chapter 6 and 7.

4.1 Classical Logic

Classical logic in AI is a formalism for declarative representation of knowledge together with sound and complete deductive reasoning mechanisms[2]. The most popular form of classical logic is first order logic. It uses the language of first order predicate formulas to represent knowledge. An important subset of first order logic is the so-called propositional logic. The key difference between propositional logic and first order logic is in their expressive power. Propositional logic is strictly less expressive that first order logic. That is, every sentence that can be expressed in the propositional language can also be expressed in first order logic, the reverse is however not true. In addition, propositional logic is decidable: given sufficient time

[2]In fact, mathematical logicians had developed declarative knowledge formalisms long before the advent of Artificial Intelligence. However, they are not interested in automated reasoning.

there is a computational reasoning process will eventually terminate with an answer of true or false; whereas first order logic is only semi-decidable. There are decidable fragments in first order logic, e.g. database systems, datalog, some description logics. The following sections give further details on both propositional and first order logic.

4.1.1 Propositional Logic

The language of propositional logic (or propositional signature) is made of non-empty set of symbols called *atoms*. An atom is consist of two basic elements:

- Constant. A constant is a word or numerals (words consisting only of digits).
- Predicate symbol. A predicate symbol is also a word defined as a symbol in the language.

A set of atoms connected by a number of *logical connectives* forms a predicate formula. Predicate formulas are used to represent specific knowledge in propositional logic. There are three types of logical connectives and each connective has its own semantics. Specifically, they are:

- 0-place connective: $\bot$ (contradiction) and $\top$ (tautology).
- Unary connective: $\neg$ (not).
- Binary connective $\wedge$ (and), $\vee$ (or), $\rightarrow$ (implication) and $\leftrightarrow$ (equivalence).

Using the Ball Passing Problem as an example, we can use atom S_A to express "robot A is stationary" and "robot B is moving" as $\neg S_B$. Therefore, to represent the notion that either robot A or B is moving, we can use a predicate formula as $\neg S_A \vee \neg S_B$.

4.1.1.1 Interpretation and Knowledge Base

The language of propositional logic also includes two truth values $TRUE$ and $FALSE$. An interpretation of a propositional language is a function that maps atoms into

x	$\neg(x)$
FALSE	TRUE
TRUE	FALSE

Table 4.1: Truth-valued function associated with unary connective.

$\{TRUE, FALSE\}$. Every propositional connective has a corresponding truth-valued function associated with it (see Table 4.1 and 4.2). The semantics of a propositional formula is given by the truth value to which an interpretation maps the formula. It is noteworthy that propositional logic is a *decidable* logic system. That is, the logical validity of any propositional formula can be determined through use of truth-valued functions.

4.1.1.2 Forms of Propositional Formula

In propositional logic, a propositional formula can take a number of forms depending on the logical connectives are used. A clause is a formula made of a disjunction of literals, e.g. $\neg S_A \vee \neg S_B$; a formula is in so-called Conjunctive Normal Form (CNF) if it is a conjunction of clauses, i.e. clauses connected with $\wedge$ connectives. On the other hand, a disjunction of conjunctions of atoms is called Disjunctive Normal Form (DNF). Any arbitrary formula can be converted into a Negation Normal Form (NNF) which is made of only conjunctions and disjunctions of literals. Formulas in NNF can then be converted into CNF using logic equivalence relationships such as $(A \vee (B \wedge C))$ to $((A \vee B) \wedge (A \vee C))$. The significance of transforming an arbitrary propositional formula into a clausal form is that we can use automated clausal theorem provers to logically deduce the formula to $TRUE$ or $FALSE$, if we have a complete knowledge base.

x	y	$\wedge(x,y)$	$\vee(x,y)$	$\rightarrow(x,y)$	$\leftrightarrow(x,y)$
FALSE	FALSE	FALSE	FALSE	TRUE	TRUE
FALSE	TRUE	FALSE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	TRUE	FALSE	FALSE
TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

Table 4.2: Truth-valued function associated with the binary connectives.

4.1.2 First Order Logic

First order logic extends both the syntax and semantics of propositional logic. It has a set of symbols (or signature) that are made of *function constants* and *predicate constants*. A non-negative integer called *arity* is assigned to each symbol such that a function constant with n arity has n number of arguments; whereas a predicate constant with n arity represents relations between n elements. As a special case, 0-nary function constants are called object constants and 0-nary predicate constants are the propositional constant.

Object variables (or variables) in first order logic are elements of fixed infinite sequence of symbols. *Terms* of a first order signature consist of both variables and functional constants. An *atomic formula* is in the form of predicate constant $P(t_1, ..., t_n)$ of arity n where each t_i is a term of first order signature. A first order formula is formed from a number of atomic formulas connected by a number of propositional connectives.

For example, we can express such a statement that the "distance between two teammates robot A and robot B is 20 centimetres and there is an opposition robot C near robot A" as the following formula:

$$Distance(RobotA, RobotB, 20) \wedge Team(RobotA, RobotB) \wedge$$
$$Nearby(RobotA, RobotC) \wedge \neg Team(RobotA, RobotC)$$

where $RobotA$, $RobotB$ and $RobotC$ are constant symbols representing robot A, B and C respectively; $Team$, $Distance$ and $Nearby$ are all predicate constants. Together with the $\wedge$ binary connective, we have a formula that maps to $TRUE$ under (current) interpretation.

Finally, first order logic introduces two quantifiers $\forall$ (for all), $\exists$ (exists) which are used to qualify formulas. The key difference between propositional logic and first order logic is that both function and predicate constants in first order logic can take a greater than zero number of arguments. Combined with $\forall$ and $\exists$ quantifiers, first order logic has greater expressive power than propositional logic. It provides simple representation of knowledge applicable for an arbitrary set of object instances whereas propositional logic can only handle individual objects effectively. However, first order logic is not a decidable logic system in general. We cannot easily determine the logical validity of individual formula independently.

4.1.3 Second Order Logic

Compared with first order logic, the main feature of second-order logic is its signature which consists of *function variables* and *predicate variables* instead of function and predicate constants. That is, we assume an infinite set of function variables of arity n for each $n > 0$ and an infinite set of predicate variables of arity n for each $n \geq 0$. Function variables can form terms with object variables in the same way as function constants. Similarly, predicate variables can form atomic formula in the same way as predicate constants. Both $\forall$ and $\exists$ quantifiers can be applied to function variables and predicate variables in addition to object variables. Therefore, second-order logic has a much richer syntax and more expressive power. For example, we can express possible composition of any two functions but the following sentence which is not possible under a first order language.

$$\forall\alpha\beta\exists\gamma\forall x(\gamma(x) = \alpha(\beta(x)))$$

Because of this richness in its expressive power, second-order logic has many applications such as defining transitive closure(Lifschitz, Morgenstern & Plaisted 2008, p.

17)[3]. For our interest, in particular, second-order logic is used in the definition of circumscription, one of the prominent non-monotonic logics. However, the expressive power of second-order logic also leads to highly complex theorem provers[4] that typically require human guidance for effective usage e.g. the HOL prover (Gordon & Melham 1993). From a practical view of modelling and reasoning about risk, the language of second-order logic does not provide significant benefits and is computationally unattractive.

4.2 Non-Classical Logic

The usefulness of classical logic is severely limited by its monotonic nature. The logical consequences reached by classical logic reasoning cannot be invalidated or revised by new information. An agent uses classical logic can only add new information that is consistent with the existing knowledge into its knowledge base. As its knowledge base grows, more conclusions can be drawn. However, no new conclusions can ever contradict old conclusions, so the body of true facts simply grows monotonically. This kind of reasoning can be used to model information safely only in a closed static world. However, the real world is neither closed nor static. An intelligent agent typically possesses incomplete information about real-world environments, and often new information is discovered to be in contradiction with existing information. Furthermore, things used to be true can become false as the environment evolves. For example, in the Ball Passing Problem, robot A may reach a conclusion that it is safe to pass the soccer ball to its closest neighbour robot B based on the its current knowledge that robot B is its teammate. However, as the game dynamic quickly changes, an opposing robot C may come in to challenge for ball possession, in which case it is no longer be safe to pass the ball to robot B. Hence, the conclusion an intelligent agent, in this case robot A, previously reached may be incompatible/inconsistent with

[3]There is even proposal to formalise mathematics using second-order logic (Väänänen 2001).
[4]In addition, the deductive systems for second-order logic lack completeness.

incoming new information. Agents must be able to revise and update their knowledge base consistently. Several classes of non-monotonic logic reasoning were developed to address this limitation in the classic logic.

4.2.1 Default Logic

The representative formalism of Default Logic was developed by Reiter (1980) based on rules like if A is true, and B is consistent with the knowledge base then deduce C. In contrast, propositional logic can only express if A is true, then C is true. Default logic formalises the intuition that we should be able to deduce new conclusions not on the basis of hard evidence, but on the absence of contrary evidence and on the basis of consistency with what is known. This ability is particularly important for hypothetical reasoning and in risk management when we have only incomplete information at hand. It allows an intelligent agent to "jump" to a conclusion based on its current knowledge.

4.2.1.1 Syntax of Default Logic

In default logic, a default theory T is a pair of (W, D), where W is a set of predicate logic formulas (i.e. facts or axioms) and a countable set of *defaults* $D = \delta_i, ...\delta_n$. A *default* δ is in the form of

$$\frac{\varphi : \psi_1, ..., \psi_i}{\chi}$$

where φ is the prerequisite (denoted as $pre(\delta)$), $\psi_1, ..., \psi_i$ are the justifications (or assumptions denoted as $just(\delta)$), and χ is the consequent of δ (denoted as $cons(\delta)$). φ, ψ and χ are all predicate formulas. We can represent a rule of thumb that "it is safe to pass a soccer ball to a friendly nearby robot B" in the robot soccer as a default

$$\frac{Nearby(RobotB) : Friendly(RobotB)}{SafePass(RobotB)}$$

where $Nearby(RobotB)$ is the prerequisite, $Friendly(RobotB)$ is the justification and $SafePass(RobotB)$ is the consequent.

4.2.1.2 Semantics of Default Logic

An informal interpretation of a default $\frac{\varphi:\psi_1,...,\psi_i}{\chi}$ is that "based on the current knowledge, if φ is known and $\psi_1, ..., \psi_i$ are also consistent with the current knowledge, we can conclude χ". Given a default theory (W, D), current knowledge is obtained by applying successive defaults δ in our default set D as long as application of δ does not lead to inconsistency. The semantics of default logic can be given in terms of an *extension* which is defined as the current knowledge satisfying certain conditions. Formally, we can define an extension E as following (Reiter 1980):

Definition. *Given a default theory $T = (W, D)$ and a set of formulas E. Let $\Lambda_T(E)$ be the least set of formulas that contains W, is closed under logic conclusion and closed under D with respect to E. E is an extension of T if and only if $E = \Lambda_T(E)$.*

This fixed-point definition implies that we can determine E is an extension of T by using E as the initial belief set and check whether we can obtain exactly the same E through applications of available defaults. The definition gives us no clue of how to compute the extension E. We have to *guess* possible extension E first and check its validity. This leads to severe limitation of practical applications of default logic. To resolve this limitation, an alternative operational definition of extension was developed Antoniou et al. (1994) that gives rise to a process model to calculate extensions.

4.2.1.3 Operational Semantics of Default Logic

Given a default theory $T = (W, D)$, let $\Pi = (\delta_0, \delta_1...)$ be a finite or infinite sequence of defaults from D. That is, we apply defaults in the sequence order in Π. For each sequence Π of a finite length of k, we define two sets of first order formulas,

$In(\Pi) = Th(W \cup \{cons(\delta) | \delta \text{ occurs in } \Pi\})$, and $Out(\Pi) = \{\neg\psi | \psi \in just(\delta)$ for some δ occurring in $\Pi\}$. $In(\Pi)$ represents the current knowledge after the defaults in Π have been applied; and $Out(\Pi)$ represents the formulas that should not be part of current knowledge even after subsequent application of other defaults. Π is a *process* of T if and only if δ_k is applicable to $In(\Pi)$ for every k such that δ_k is in Π.

For a given process Π of T, Π is *successful* if and only if $In(\Pi) \cap Out(\Pi) = \phi$. It *fails*, otherwise. Π is also *closed* if and only if every $\delta \in D$ applicable to $In(\Pi)$ already occurs in Π. This conforms to the desired property of an extension E that E should be closed under application of defaults in D. We now can give an alternative definition for extension (Antoniou & Sperschneider 1994):

Definition. *A set of formula E is an extension of a default theory T if and only if there exists some closed and successful process Π of T such that $E = In(\Pi)$.*

It has been shown that this definition is equivalent to the fixed-point definition (Antoniou & Sperschneider 1994). Compared with the original definition, the operational definition for extension provides a path way to construct extensions mechanically instead of "guessing". We can search through the $In(\Pi)$ space and any $In(\Pi)$ that is successful and closed is an extension of T.

4.2.1.4 Variations of Extensions

The ability to generate extensions systematically does not guarantee the existence of an extension for an arbitrary set of defaults. Classic default logic also does not obey semi-monotonicity, i.e. addition of a new default should yield more information not less. One way to address these issues is to restrict defaults to be *normal defaults*, i.e. their consequent is their own justification. This class of default theories is strictly less expressive than the classical default theories and sometimes produces counterintuitive results. It turns out that the the less restrictive *semi-normal* default theories can give more reasonable results (Etherington 1987). An alternative approach is to modify the concept of extension such that at least one extension exists. Several variants of default logic have been carefully studied, for example, justified default logic (Lukaszewicz 1988) seeks for maximal successful processes Π as the *modified extension* of T; and constrained default logic (Schaub 1992) enforces joint consistency of justifications of applied defaults. For further details on these approaches, we point readers to above referenced materials.

4.2.1.5 *Applicability of Default Logic

Default logic offers an elegant solution for logic based reasoning in domains that we have only limited knowledge. It addresses the issues raised from exceptions of rules in static and closed worlds through use of defaults[5]. In default logic, the knowledge base of a domain is captured as sets of known facts and defaults and is represented in the form of extensions. When new domain information is acquired in the form of facts, any existing default that is in conflict with the new information will have to be retracted, and consequently, extensions of the default theory will have to be recomputed. Addition of new defaults will also trigger similar re-computation of extensions. This means, every revision of domain knowledge base will incur a heavy cost of extension calculation. Furthermore, default logic works on justified (or grounded) knowledge and does not have a concept of belief. It is not clear how one may encode uncertainty related information into a default logic based knowledge base. Therefore, default logic does not seem to be ideal candidate for capturing and representing risk related knowledge for the target domains that face frequent knowledge revision with (potentially conflicting) information in open world environments, where knowledge may be incomplete and uncertain.

4.2.2 Autoepistemic Logic

Autoepistemic logic (Moore 1985) is another prominent non-monotonic reasoning formalism. It extends the language of first order logic by introducing a modal operator

[5]In fact, the Closed World Assumption was introduced by Reiter (1980) and can be expressed as a default $\frac{true:\neg\varphi}{\neg\varphi}$.

K. This K operator (and its negation) may be applied (repeatedly) to a first order formula[6], i.e. a sentence, such that $K\varphi$ means "I *know* φ (or I believe in φ)[7]". Apply a negation of the operator K to a sentence, e.g. $\neg K\varphi$, means "I *do not know* φ". Therefore, the language of autoepistemic logic consists of

- Every closed first order formula (i.e. sentences).
- $K\varphi$, if φ is an autoepistemic formula.
- $\neg\varphi$,$(\varphi \vee \psi)$, $(\varphi \wedge \psi)$ and $(\varphi \rightarrow \psi)$, if both φ and ψ are autoepistemic formulas.

4.2.2.1 Semantics of Autoepistemic Logic

Autoepistemic logic is based on the notion of *belief* and introspection of a rational agent. One of the key concepts in the logic is stability. That is,

- If $\varphi \in E$, then $K\varphi \in E$.
- If $\varphi \notin E$, then $K\varphi \notin E$.

This means if φ is in my knowledge (E) then I *know* φ; if φ is *not* in my knowledge (E) then I do not *know* φ. Formally, the semantics of autoepistemic logic is given by *expansion* which is defined as a fixed-point function.

Definition. *(Moore 1985) Let T be an autoepistemic theory, a closed set of autoepistemic formulas E is an expansion of T if E satisfy the following equality:*

$$Cn(T \cup \{K\varphi | \varphi \in E\} \cup \{\neg K\varphi | \varphi \notin E\}) = E,$$

where Cn is a consequence operator that collects all logical consequences of the enclosed formula. An expansion represents a "world view (or belief)" a rational agent holds according to its own beliefs (and knowledge).

[6] K cannot be applied to a formula with free variables, i.e. the formula must be quantified.
[7] We use know and believe interchangeably.

4.2.2.2 Computation of Expansions

Similar to default logic extension, the fixed-point definition of expansion does not provide a constructive method of finding expansions. An alternative method was developed to compute expansions. First we need to introduce few syntactic concepts. The *degree* of an autoepistemic formula φ, denoted as $degree(\varphi)$, is the maximum depth of K nesting. For example, $degree(K\neg K\varphi) = 2$. A first order formula has degree of 0. The *kernel* of autoepistemic theory T is the set of all first order formulas that are members of T, i.e. T_0.

Definition. *(Antoniou & Sperschneider 1993) For an autoepistemic theory T, $sub(T)$ is the union of $sub(\varphi)$, for all $\varphi \in T$, where $sub(\varphi)$ is defined as:*

- $sub(\varphi) = \emptyset$ *for first order formula* φ.
- $sub(\neg\varphi) = sub(\varphi)$.
- $sub(\varphi \vee \psi) = sub(\varphi \wedge \psi) = sub(\varphi \rightarrow \psi) = sub(\varphi) \cup sub(\psi)$.
- $sub(K\varphi) = \varphi$.

It has been shown that it is sufficient to consider belief or non-belief in formulas in $sub(T)$ to determine the expansions of T, and we can devise a computational procedure to determine the expansions for T (Antoniou & Sperschneider 1993):

$Expansion := \emptyset$. AE_0 is a set of kernel autoepistemic formulas.
Partition $sub(T)$ into a partition of beliefs $E(+)$ and a partition of non-beliefs $E(-)$.
for all partition $E(+)$ and $E(-)$ of $sub(T)$ **do**
  $E(0) := \{\varphi \in AE_0 | T \cup \{K\varphi | \varphi \in E(+)\} \cup \{\neg K\varphi | \varphi \notin E(-)\} \models \varphi\}$
  if $E(+) \subseteq E(0)$ and $E(-) \cap E(0) = \emptyset$ **then**
    $Expansion := Expansion \cup \{E(0)\}$
  end if
end for

4.2.2.3 *Relation with Default Logic

Autoepistemic logic shares many similarities with default logic. Both expansions and extensions are defined as fixed-point functions from which their respective nonmonotonic inference relations are defined. Both logics are required to work within certain contexts. Central to autoepistemic logic is the notion of belief; whereas default logic is rooted in facts and justified knowledge. Expansions allow *self-justifications* whereas extensions do not. In others words, a formula φ is justified through $K\varphi \wedge \neg K \neg \psi \supset \varphi$, relying on believing in φ (as long as no information contradicting ψ). In extension, a default $\frac{\varphi:\psi}{\varphi}$ cannot be applied until φ is derived independently. This key difference sets these two logics apart. In fact, it has been shown that default logic can be viewed as a "restricted" version of autoepistemic logic (Denecker, Marek & Truszczyński 2003). The concept of belief (or belief set) used in autoepistemic logic is the central concept that allows the logic to be characterised in terms of possible world structures. This matches nicely with the requirement of capturing and representing uncertainty. However, similar to default logic, autoepistemic logic only deals with static worlds and suffers from the same problem of recomputing everything from scratch in light of new information. This deters us from adoption of autoepistemic logic in our framework.

4.2.3 Circumscription

Both default logic and autoepistemic logic discussed previously provide their nonmonotonic reasoning functionalities by modifying classical logic with additional syntactical constructs of defaults and the modal operator K respectively. Circumscription (McCarthy 1980), however, does not take this approach and makes use of the unmodified language of classical logic. Instead, circumscription transforms a logical sentence A into a logically stronger sentence A^* by minimising the extents of predicate (or functions) within the sentence A. For example, we can express "professors normally teach" as the following first order formula:

$$\forall x(prof(x) \wedge \neg abnormal(x) \supset teaches(x)).$$

Naturally, we do not expect a professor to be *abnormal* in most situations. Circumscription uses exactly this intuition that "we do not expect an entity to be abnormal unless we have explicit information that tells us otherwise", and minimise the extents of *abnormal* so that we can draw the conclusion without the need to have a complete picture of the target domain. We provide formal definition of circumscription in the following section.

4.2.3.1 Syntax of Circumscription

Let us first define some useful abbreviations in order to express the formal definition of circumscription in a clear form. Let P and Q be two predicate symbols of the same arity n such that:

$$P = Q \text{ stands for } \forall x_1...x_n((P(x_1,...,x_n) \equiv Q(x_1,...,x_n)),$$
$$P \leq Q \text{ stands for } \forall x_1...x_n((P(x_1,...,x_n) \supset Q(x_1,...,x_n)),$$
$$P < Q \text{ stands for } (P \leq Q) \wedge \neg(P = Q).$$

These formulas mean: P and Q have the same extent; the extent of P is a subset of the extent of Q; and the extent of P is a proper subset of the extent of Q respectively.

Definition. *(McCarthy 1980) Let $A(P)$ be a first order sentence containing the predicate constant P. Let p be a predicate variable of the same arity as P. The circumscription of P in $A(P)$, denoted as $CIRC[A(P);P]$, is the second order sentence:*

$$A(P) \wedge \neg\exists p[A(p) \wedge p < P].$$

The second order formula $\neg\exists p[A(p) \wedge p < P]$ means we cannot find a predicate p which has smaller extent than P. In other words, P is the predicate symbol with the minimum extent and $CIRC[A(P);P]$ is the original first order sentence modified by the predicate P of minimum extent. We can further generalise this definition by introducing additional predicates and functional constants such that multiple predicates are minimised in parallel with respect to other varying functional symbols.

Definition. *(McCarthy 1980) Let* $P = P_1, ..., P_k$ *be a sequence of predicate constants,* $Z = Z_1, ..., Z_m$ *a sequence of function constants. Let* $A(P, Z)$ *be a first order sentence containing the predicate constants* P_i *and function constants* Z_i*. Let* $p = p_1, .., p_k$ *and* $z = z_1, ..., z_m$ *be predicate/function variables of the same type and arity as* $P_1, ..., P_k$ *and* $Z = Z_1, ..., Z_m$ *respectively. The circumscription of* P *in* $A(P, Z)$ *with varied* Z*, denoted as* $CIRC[A(P, Z); P; Z]$*, is the second order sentence:*

$$A(P, Z) \wedge \neg\exists pz[A(p, z) \wedge p < P].$$

4.2.3.2 Semantics of Circumscription

Minimisation of the extent of predicates in circumscription can be explained in terms of a preference relation on the models of circumscribed sentence A. We prefer a model M_1 over a model M_2 if the extent of predicate P is smaller in M_1 than in M_2 and both M_1 and M_2 share the same universe and agree on the fixed constants. Put this in a formal language:

Definition. *(McCarthy 1980) Let* M_1 *and* M_2 *be structures,* $|M|$ *denoted as the universe of* M*, and* $M[\![C]\!]$ *is the interpretation of the (individual or function or predicate) constant* C *is* M*.* P *is a sequence of predicate constants; and* Z *is a sequence of predicate (or function) constants.* M_1 *is at least as* $P; Z$*-preferred as* M_2*, denoted as* $M_1 \leq^{P;Z} M_2$*, whenever the following conditions hold :*

- $|M_1| = |M_2|$,
- $M_1[\![C]\!] = M_2[\![C]\!]$ *for every constant* C *which is neither in* P *nor* Z*,*
- $M_1[\![P_i]\!] \subseteq M_2[\![P_i]\!]$ *for every predicate constant* P_i *in* P*.*

The relation $\leq^{P;Z}$ is a transitive and reflexive relation. A structure M is said to be $\leq^{P;Z}$-minimal within a set of structures $\mathcal{M}$ when there is no structure $M' \in \mathcal{M}$ such that $M' \leq^{P;Z} M$. The process of circumscription of a sentence can be understood as preferentially selecting a model of A (over other models) in which the extent of predicate P is minimal. That is,

Proposition 1. *(McCarthy 1980) M is a model of $CIRC[A;P;Z]$ if and only if M is $\leq^{P;Z}$-minimal among the models of A.*

Nonmonotonic reasoning is achieved through restricting logical entailment to the most preferred models. Formulas that entail from the original sentence A can still be entailed from the logically stronger A^*.

4.2.3.3 Applicability of Circumscription

Circumscription deals with incomplete knowledge and solves the frame problem by identifying the most preferred (or most expected) models of A over other models and restricting entailment to the most preferred models. Syntactically, a preferential relation between models of A is established from calculating the extent of predicates within A and the most preferred model of A is the predicate with the minimal extent. For this, circumscription translates a first order formula into a second order formula which is not even semi-decidable. This means there is no general theorem prover we can use to do sceptical inference, i.e. determine whether a formula is true in all minimal models. Therefore, apart from certain special cases (Lifschitz 1985) first order circumscription is highly uncomputable (Schlipf 1986). Furthermore, similar to default logic and autoepistemic logic, circumscription assumes a static world and an environment with *changing* information will force a repeat of circumscription process.

4.2.4 *Incomplete Knowledge and Changing Information

So far I have surveyed three prominent nonmonotonic reasoning formalisms that focus on dealing with incomplete knowledge. Having only limited domain knowledge is extremely common in practical real-world environments. However, this is only one key characteristic of many real-world environments. As I have discussed earlier in the framework requirements section of the previous chapter, most domains we are interested continuously evolve and change. Any viable framework must support the continuous acquisition of new information and deal with *changing* information. All three formalisms discussed above essentially ignore this important issue which prevents the effective adoption of these nonmonotonic reasoning techniques in many practical applications.

For this thesis, I assume that an agent has limited knowledge of the domain and environment in which it operates. After all, uncertainties (as in risk) only arise when we have incomplete domain knowledge. I expect that intelligent agents build up their knowledge base in a gradual and timely fashion by continuously acquiring new domain information. Therefore, the framework requires methods that can effectively handle new, potentially conflicting (with the existing knowledge base) information.

In the following sections, I introduce *Belief Revision* based on the so-called AGM postulates[8], an alternative class of nonmonotonic reasoning that has been developed in (almost) parallel with above discussed formalisms over the past thirty years. Belief Revision are mechanisms concerned with revising and maintaining existing knowledge bases. I will review and discuss the foundation, syntax, semantics and algorithms of AGM based BR in much detail since it will become one of the main pieces in the qualitative risk management introduced in Chapter 6. In addition, AGM Belief Revision is strongly related to possibility theory. I will discuss their connection in Section 4.6.2.1.

4.3 Belief Revision

Belief Revision (BR) develops formal mechanisms and processes for modifying and maintaining knowledge repositories with new information. The new information may

[8]From here on, Belief Revision (or BR) in this thesis refers specifically to AGM based Belief Revision.

not be consistent with the existing knowledge; some of existing knowledge may be retracted in order to maintain the consistency. Therefore, the very nature of BR matches nicely with the main assumption and requirements of HiRMA framework.

4.3.1 AGM Paradigm

Seminal work of Alchourrón, Gärdenfors and Makinson gave rise to the so called *AGM paradigm* (Alchourron, Gardenfors & Makinson 1985) which has been the dominant framework for logic-based belief revision. AGM provides number of postulates for expansion (adding knowledge), contraction (retracting knowledge) and revision (modifying knowledge) operations on a classical logic theory that represents a set of beliefs. These postulates ensure that any belief changes that adhere to them are logically consistent and respect the principle of minimal change, i.e. making minimum amount of changes that are absolutely necessary but no more. We give a brief summary of these postulates. For a more detailed description, see (Gärdenfors 1992).

4.3.1.1 Preliminaries

AGM requires a formal language $\mathcal{L}$ that is closed under all Boolean connectives and a simple logical consequence relation $\vdash$ such that:

1. $\vdash \alpha$ for all tautologies.
2. if $\vdash (\alpha \rightarrow \beta)$ and $\vdash \alpha$, then $\vdash \beta$.
3. $\vdash$ is consistent, i.e. $\nvdash L$.
4. $\vdash$ satisfies the deduction theorem. That is, $\{\alpha_1, \alpha_2, ..., \alpha_n\} \rightarrow \beta$ iff $\vdash \alpha_1 \wedge \alpha_2 \wedge ... \wedge \alpha_n \rightarrow \beta$.
5. $\vdash$ is compact.

4.3.1.2 Postulates for Expansion

The expansion operation takes a new sentence/information α (with its logical consequences) and add to a belief set K; resulting K is expanded by α.

(K+1) For a sentence α and a belief set K, $K + \alpha$ is a belief set. (Closure)

(K+2) $\alpha \in K + \alpha$.

(K+3) $K \subseteq K + \alpha$.

(K+4) If $\alpha \in K$, then $K + \alpha = K$.

(K+5) If $K \subseteq H$, then $K + \alpha \subseteq H + \alpha$.

(K+6) For all belief sets K and all sentences α, $K + \alpha$ should be the smallest belief set that satisfies (K+1) to (K+5).

4.3.1.3 Postulates for Revision

The revision operation takes a new sentence/information α that is not fully consistent with the belief set K and add to the K. Some existing sentences in K are removed in order to incorporate the new information and remain consistent.

(K*1) For a sentence α and a belief set K, $K * \alpha$ is a belief set. (Closure)

(K*2) $\alpha \in K * \alpha$.

(K*3) $K * \alpha \subseteq K + \alpha$.

(K*4) If $\neg\alpha \notin K$, then $K + \alpha \subseteq K * \alpha$.

(K*5) If α is consistent, then $K * \alpha$ is also consistent.

(K*6) If $\vdash \alpha \leftrightarrow \beta$, then $K * \alpha = K * \beta$.

(K*7) $K * (\alpha \wedge \beta) \subseteq (K * \alpha) + \beta$.

(K*8) If $\neg\beta \notin K * \alpha$, then $(K * \alpha) + \beta \subseteq K * (\alpha \wedge \beta)$.

Among these eight postulates: (K*1) and (K*5) are self-explanatory. (K*2) states that after revision, the result belief set should always include new information. (K*3) and (K*4) means when new information is consistent with the existing belief set, there is no reason to remove any existing belief in the belief set. These two postulates carry the notion of minimal change in a limited way. (K*6) means that syntax of new sentence has no effect on the revision process. The last two postulates (K*7) and (K*8) are known as the optional postulates concerning composite Belief Revision. They, again, carry the expression for the minimal change principle. They basically state that for any two sentences α and β, expansion (with minimal change) of K to include both α and β should be the same as the expansion of $K + \alpha$ by β if β is consistent with $K + \alpha$.

4.3.1.4 Postulates for Contraction

The contraction operation retires an existing sentence from K without adding any new information.

(K$\dot{-}$1) $K\dot{-}\alpha$ is theory.

(K$\dot{-}$2) $K\dot{-}\alpha \subseteq K$.

(K$\dot{-}$3) If $\alpha \notin K$, then $K\dot{-}\alpha = K$.

(K$\dot{-}$4) If $\nvdash \alpha$, then $\alpha \notin K\dot{-}\alpha$.

(K$\dot{-}$5) If $\alpha \in K$, then $K \subseteq (K\dot{-}\alpha) + \alpha$.

(K$\dot{-}$6) If $\vdash \alpha \leftrightarrow \beta$, then $K\dot{-}\alpha = K\dot{-}\beta$.

(K$\dot{-}$7) $(K\dot{-}\alpha) \cap (K\dot{-}\beta) \subseteq K\dot{-}(\alpha \wedge \beta)$.

(K$\dot{-}$8) If $\beta \notin K\dot{-}(\alpha \wedge \beta)$, then $K\dot{-}(\alpha \wedge \beta) \subseteq K\dot{-}\alpha$.

Postulates (K$\dot{-}$1) and (K$\dot{-}$2) are self-evident. (K$\dot{-}$3) tells us if the sentence is not in the initial belief set K, then contraction should not change K. (K$\dot{-}$4) means all sentences can be removed from the belief set K (in principle), apart from tautologies. (K$\dot{-}$5) postulate is called the recovery postulate. It means by removing and later adding the same sentence α, we should recover the original belief set. This postulate is often considered to be overly restrictive (Gärdenfors 1992) and often not satisfy by belief base revision. The sixth postulate is in parallel with the one in revision postulate. The last two postulates are also analogous to the last two revision postulates. They also express the principle for minimal change. That is, a belief that survives the contraction of α and β should not be affected by contraction of $\alpha \wedge \beta$. $K\dot{-}\alpha$ is the minimal change of removing α. It contains (at least) all the beliefs of $K\dot{-}(\alpha \wedge \beta)$ when $\alpha \notin K\dot{-}(\alpha \wedge \beta)$.

4.3.1.5 Belief Contraction versus Belief Revision

The belief contraction and revision operations are closely related. In fact, it has been shown through *Levy* and *Harper* identities that, there is one-to-one relationship between revision and contraction functions (Gärdenfors 1988).

$$K * \alpha = (K\dot{-}\neg\alpha) + \alpha \quad (Levy) \tag{4.1}$$

$$K\dot{-}\alpha = (K * \neg\alpha) \cap K \quad (Harper) \tag{4.2}$$

That is, an AGM compliant contraction function can be generated from an existing revision function and vice versa. Therefore, obtaining one belief change operator, either contraction operator or revision operator, is sufficient for modifications of a knowledge repository within the AGM framework.

4.3.2 Belief Operator Selection

In most cases, there are multiple belief change functions satisfy all of the AGM postulates. Some extra-logical information of some epistemic ordering is needed to select the most appropriate unique function for belief change. There are two main approaches for solving this problem, namely, *epistemic entrenchment* developed by Gärdenfors et al. (1988) and *System of Spheres* (Grove 1988).

4.3.2.1 Epistemic Entrenchment

The concept of Epistemic Entrenchment represents the resistance of a belief α against change. "It is formally defined as a preordering relation $\leq$ on $\mathcal{L}$, encoding the relative 'retractability' of individual belief" (Peppas 2007). For example, $\alpha \leq \beta$ means that β is less easily given up (by an agent) than the α. To define the meaning of an epistemic entrenchment ordering more precisely, five additional postulates were introduced (Gärdenfors & Makinson 1988):

(EE1) If $\alpha \leq \beta$ and $\beta \leq \chi$, then $\alpha \leq \chi$.

(EE2) If $\alpha \vdash \beta$ then $\alpha \leq \beta$.

(EE3) $\alpha \leq \alpha \wedge \beta$ or $\beta \leq \alpha \wedge \beta$.

(EE4) When K is consistent, $\alpha \notin K$ iff $\alpha \leq \beta$ for all $\beta \in \mathcal{L}$.

(EE5) If $\alpha \leq \beta$ for all $\alpha \in \mathcal{L}$, then $\vdash \beta$.

(EE1) obviously states $\leq$ is transitive. However, (EE2) is less clear, since it means logically stronger beliefs are at least as entrenched as the weaker ones. This axiom can be understood using the principle of minimal change. To remove a logically stronger belief β, it requires α also to be removed, whereas removing α does not require removal of β. This means removing β causes more information loss. Hence, (EE2) adheres to the minimal change principle. (EE3) is a direct result from contraction of $\alpha \wedge \beta$ in AGM. Since removal of $\alpha \wedge \beta$ means either α or β (even both) would have to be removed, it implies that α or β is no more entrenched than $\alpha \wedge \beta$. An epistemic entrenchment order can define a contraction operation as (Gärdenfors & Makinson 1988)

$$\beta \in K\dot{-}\alpha \text{ iff } \beta \in K \text{ and either } \alpha < \alpha \vee \beta \text{ or } \vdash \alpha$$

It turns out, not surprisingly, we can generally find more than one preordering $\leq$ that conforms to the five axioms stated above for a fixed belief set K. This is to due to the inherited subjective nature of preordering. An agent can elect to use different epistemic entrenchment ordering schemes depending on its perspective. This issue also exists in different forms in other BR operator constructions such as System of Spheres. AGM Belief Revision does not address this issue at all. Determining the preordering to be used, is a critical issue for successful applications of Belief Revision. In fact, there is a strong connection between epistemic entrenchment ordering with specific application of Belief Revision, since it requires "meaning" which can only be discussed within a certain domain/context. The meaning of *risk* is crucial for our application of BR in the risk management framework.

4.3.2.2 System of Spheres

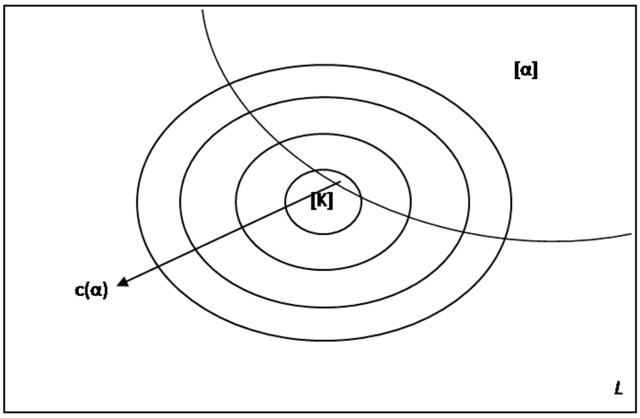


Figure 4.1: A System of Spheres

System of Spheres (Grove 1988) models the complete theories in a nested sphere structure S similar to Figure 4.1. The smallest sphere in the centre of S is the initial belief set $[K]$. The rest of spheres which represent all possible (consistent) worlds/theories are ordered based on some relative plausibility rule. The more plausible a world is, the closer it is to the centre of S. All sentences $\alpha \in \mathcal{L}$ will be members of the largest sphere S.

Formally, a System of Spheres S must adhere to the following rules (Grove 1988):

(S1) S is totally ordered by set inclusion; that is, if $V, U \in S$, then $V \subseteq U$ or $U \subseteq V$.

(S2) $[K]$ is the smallest sphere in S; i.e. $[K] \in S$, if $V \in S$, then $[K] \subseteq V$.

(S3) $M_L \in S$ and is the largest sphere in S.

(S4) If a sentence $\alpha \in \mathcal{L}$ and $[\alpha]$ intersects any sphere in S, then there must have a smallest sphere in S intersecting α.

Noted (S4) is also known as *Limit Assumption*. It ensures there exists a smallest sphere that contains every α. The smallest sphere that intersects $[\alpha]$ is denoted as $c(\alpha)$. The Belief Revision function for the system of spheres comes out naturally as (Peppas 2007):

$$K * \alpha = \begin{cases} \bigcap(c(\alpha) \cap [\alpha]) & \alpha \text{ is consistent} \\ \mathcal{L} & otherwise \end{cases} \tag{4.3}$$

Compared with epistemic entrenchment, the system of spheres provides a more intuitive representation of ordered belief sets as extra-logical information is inherently encoded in the nested structure. It provides a means of constructing revision functions. Both epistemic entrenchment and system of spheres construction are, however, only capable of once-off revision. Further revision operations would require an updated ordering relation rather than just an updated belief set. Therefore, iterated revision requires more sophisticated machinery to be able to revise the revision function itself. The solution is provided by Ordinal Conditional Functions(OCF) and a more generalised *transmutation* process.

4.3.3 Belief Base Change

One of the fundamental assumptions of the AGM paradigm is that agents' belief sets are modelled as theories. Theories could consist of infinite number of sentences; implementations of revision functions within AGM paradigm are computationally infeasible. We could only realise our Belief Revision applications based on the finite representations of theories, i.e. belief bases (Nebel 1998, Hansson 1999). Furthermore, it is common to distinguish between explicit beliefs (Grove 1988, Gärdenfors 1992, Nayak 1994), beliefs that agents accept in their own right, and those beliefs that are the logical entailments of the explicit beliefs. The derived beliefs should be removed once their supporting explicit beliefs are gone. Therefore, research in belief base revision has more practical significance.

Most work on belief base revision starts from a theory base B and a preference ordering on the sentences in B; they provide various methods of revising B under existing preference ordering. There are two approaches to the belief base revision problem. One approach, belief base revision operation, focuses on developing parallel revision operations for belief bases instead of belief theories. It takes account of explicit beliefs and derived beliefs and only operates on explicit beliefs (Nebel 1998). The end results from the revision operations are also belief bases. Revision operations on belief bases can be transformed into revisions on theories by logical closure (Hansson 1999).

The other type of belief base revision is called belief base revision scheme. This approach can be considered as a construction model for contraction (or revision) functions similar to selection functions, epistemic entrenchment. Belief base revision schemes work on explicit belief bases and generate theories as the end results. Therefore, they are more useful than the belief base revision operation for practical considerations. It has been proven that base-generated contract functions satisfy all AGM contraction postulate (K-1) to (K-8) (Rott 1993).

Similar to the introduction of epistemic entrenchment to represent full ordering of beliefs on theories; the concept of *ensconcement* was introduced to represent a total ordering on belief base B. *Ensconcement* satisfies the so-called Priority Consistency Condition (PCC) (Rott 1991):

Proposition 2. *For all $\alpha \in B$, if B' is a non-empty subset of B that entails α then there is a $\beta \in B'$ such that $\beta \preccurlyeq \alpha$.*

$\preccurlyeq$ is an ensconcement ordering relation and $\beta \preccurlyeq \alpha$ means β is a weaker belief than α. Furthermore, it has been shown that PCC is a necessary and sufficient condition for ensconcement ordering to be extended to a full epistemic entrenchment ordering (Rott 1991) and ensconcement generated revision functions are AGM revision functions (Williams 1994a, Williams 1994b).

4.3.4 Iterative Belief Revision

4.3.4.1 Ordinal Conditional Functions (OCF)

An OCF function κ is a function that maps a set of possible worlds to set of ordinals such that at least one world has an ordinal value of zero. The world with zero κ value are the most plausible worlds, whereas the larger κ value a world r has, the less plausible the world. In addition to the OCF, a degree of firmness d is used as an additional input information for revision. This degree of firmness indicates how an agent should accept the new information. The new revision process produces not only the revision of K by α but also a new OCF defined as following:

$$\kappa * \langle \alpha, d \rangle (r) = \begin{cases} \kappa(r) - \kappa(\alpha) & r \in [\alpha] \\ \kappa(r) - \kappa(\neg\alpha) + d & otherwise \end{cases} \tag{4.4}$$

This process is called *conditionalisation* (Spohn 1988). It essentially means that all α worlds shift downwards (decrease κ value) against $\neg\alpha$ worlds till the most plausible α world reaches zero value; and, at the same time, the $\neg\alpha$ worlds shift upwards (increase κ value) till the most plausible of them is at d from zero. Not only the knowledge set (where $\kappa(K) = 0$) is modified, so are the total ordering of all possible worlds.

4.3.4.2 Transmutations

An alternative process to conditionalisation is called *adjustment* (Williams 1994b). Compared with the *conditionalisation*, it minimises number of changes to the grades of possible worlds. The only worlds that change their grades are the most plausible α worlds (changes to zero) and $\neg\alpha$ worlds with grades smaller than d (Williams 1994b).

$$\kappa \circ \langle \alpha, d \rangle (r) = \begin{cases} 0 & \text{if } r \in [\alpha] \text{ and } \kappa(r) = \kappa(\alpha) \\ d & \text{if } r \in [\neg\alpha] \text{ and } \kappa(r) = \kappa(\neg\alpha) \\ & \text{or } \kappa(r) \leq d \\ \kappa(r) & \textit{otherwise} \end{cases} \tag{4.5}$$

Both *conditionalisation* and *adjustment* processes are *transmutations* that modify an ordinal conditional function under the constraints of revision and contraction postulates described in Section 4.3.1.3 and 4.3.1.4. We will use the *adjustment* as the main risk model revision mechanism in the new framework introduced in Chapter 6.

4.3.4.3 Ordinal Epistemic Functions

In contrast to Ordinal Conditional Functions, an Ordinal Epistemic Function(OEF) is a function E that maps formulae (or sentences) in a language to the class of ordinals. It satisfies the following conditions (Williams 1994b):

(OEF1) For all $\alpha, \beta \in \mathcal{L}$, if $\alpha \vdash \beta$, then $E(\alpha) \leq E(\beta)$.

(OEF2) For all $\alpha, \beta \in \mathcal{L}$, $E(\alpha) \leq E(\alpha \wedge \beta)$ or $E(\beta) \leq E(\alpha \wedge \beta)$.

(OEF3) $\vdash \alpha$ if and only if $E(\alpha) = \mathcal{O}$, where $\mathcal{O}$ is the ordinal we defined for our system.

(OEF4) If α is inconsistent, then $E(\alpha) = 0$.

Basically, an OEF ranks all formulae in $\mathcal{L}$: the higher ordinal is assigned to a formula, the more entrenched (or firmly believed) the formula is in our system. In particular, formulae that are self-contradictory has lowest ranking of zero. That is, we do not believe it at all. Note, the ordering in OEFs is opposite of the ordering in OCFs. We can define various transmutation strategies for OEFs similar to those transmutations for OCFs such that (α, i)-conditionalisation for E is:

$$E * (\alpha, i)(\beta) = \begin{cases} -E(\neg\alpha) + E(\beta) & \text{if } \alpha \wedge \neg\beta \nvdash \perp \\ -E(\alpha) + E(\beta) + i & \textit{otherwise}, \end{cases} \tag{4.6}$$

and (α, i)-adjustment for E becomes:

$$E \circ (\alpha, i)(\beta) = \begin{cases} 0 & \text{if } E(\neg\alpha) = E(\neg\alpha \vee \beta) \\ E(\beta) & \text{if } E(\neg\alpha) < E(\neg\alpha \vee \beta) \\ & \text{and } E(\beta) > i \\ i & \text{if } E(\neg\alpha) < E(\neg\alpha \vee \beta) \\ & \text{and } E(\beta) \leq i < E(\neg\alpha \vee \beta) \\ E(\neg\alpha \vee \beta) & \textit{otherwise}. \end{cases} \tag{4.7}$$

It has been shown (Williams 1994b) that OCF and OEF are closely related to each other. In fact, they are *similar* as long as all nontautological formulae α, β satisfies the following condition:

$$E(\alpha) \leq E(\beta) \text{ if and only if } C([\neg\alpha]) \leq C([\neg\beta]).$$

Furthermore, if and only if, all non tautological formulae have the same relative ranking with respect to both E and C, then OCF and OEF are equivalent; their transmuted knowledge sets are also equivalent. Therefore, in the rest of this thesis, we use system of spheres/OCF and epistemic ranking structure/OEF interchangeably when we discuss transmutation process in our risk modelling. The main advantage of the OEFs over the OCFs is that they are more practical in terms of actual implementation in computation.

Algorithm 1 below lists an transmutation algorithm for OEFs called *maxi-adjustment*. The maxi-adjustment algorithm is a variant of standard adjustment transmutation procedure and its sentence ranking adjustment is based on Sphon's notion of reason (Spohn 1983). That is, β is a *reason* for α, if and only if raising the epistemic rank of β will also raise the epistemic rank of α. Information should only be retracted if there is a good reason to do so. Maxi-adjustment can be implemented as an anytime algorithm so that it can produce a partial solution that approximates the final transmutation solution if it is interrupted, and the longer it runs a better approximation it produces. This property is important of resource bound intelligent agents.

Algorithm 1 Maxi-adjustment Algorithm

Require: A partial entrenchment ranking E; a sentence α with a natural number i representing the new desired ranking for α
Ensure: A new partial entrenchment ranking E'

if $\alpha \in dom(E)$ **then**
    $degree_\alpha = E(\alpha)$.
else
    $degree_\alpha = \textsc{Degree}(E, \alpha)$.
end if
$max\_degree = max(E(dom(E))$.
if $degree_\alpha > i$ **then**
    $E' = \textsc{MoveDown}(\alpha, degree_\alpha, i, E)$
else if $degree_\alpha < i$ **then**
    if $degree_\alpha = 0$ **then**
        $degree_{\neg\alpha} = \textsc{Degree}(E, \neg\alpha)$.
    else
        $degree_{\neg\alpha} = 0$.
    end if
    if $degree_{\neg\alpha} > 0$ **then**
        $E' = \textsc{MoveDown}(\alpha, degree_{\neg\alpha}, 0, E)$.
    end if
    $E' = E$.
    $E' = \textsc{MoveUp}(\alpha, i, E)$
else
    $E' = E$.
end if
return E'.

Algorithm 2 Maxi-adjustment Algorithm: Degree, MoveUp and MoveDown functions.

function DEGREE(E, α)
    $degree = max\_degree.$
    while $\{\beta : degree \geq E(\beta)\} \nvdash \alpha$ and $degree \neq 0$ **do**
        $degree = degree - 1.$
    end while
    $E(\alpha) = degree.$
return *degree*
end function
function MOVEDOWN(α, i, j, E)
    for $k = max\_degree$ down to $i + 1$ **do**
        for all $\beta \in dom(E)$ **do**
            if $E(\beta) = k$ **then**
                $E'(\beta) = k.$
            end if
        end for
    end for
    for $k = i$ down to j **do**
        $minimum\_set = \emptyset.$
        for all $\{\beta : E(\beta) = k\}$ **do**
            if $(\beta \wedge dom(E')) \vdash \alpha$ **then**
                $minimum\_set = minimum\_set \cup \{\beta\}.$
            end if
        end for
        for all $\{\beta : \beta \in dom(E) \wedge \beta \notin minimum\_set\}$ **do**
            $E'(\beta) = k.$
        end for
    end for
return E'.
end function
function MOVEUP(α, i, E)
    for $k = max\_degree$ down to $i + 1$ **do**
        for all $\beta \in dom(E)$ **do**
            if $E(\beta) = k$ **then**
                $E'(\beta) = k.$
            end if
        end for
    end for
    $E'(\alpha) = i.$
    for $k = i - 1$ down to 1 **do**
        $E'(\beta) = $ DEGREE$(E', \alpha \vdash \beta).$
    end for
return E'.
end function

4.4 Bayesian Probabilistic Model

Both the classical logics and the standard non-monotonic logics only deal with qualitative data, they cannot handle numerical information[9] usually required to represent uncertainty[10]. Most common and mature methods of representing uncertainty use probability. One of most popular probabilistic models in AI is the so-called Bayesian Network (BN) (Pearl 1988). A Bayesian Network consists of an intuitive Directed Acyclic Graph (DAG) that uses nodes to represent domain variables and directed arcs between nodes represent dependencies between the variables. Mathematically, BN can be viewed as a joint probability distribution of variables and it handles numerical non-monotonic reasoning nicely. Because of these useful features, BNs have been extensively studied since their first formal introduction by Pearl (1988). Numerous extensions and techniques for BN construction, network learning (both structure learning and parameter learning), refinement and inferences have been developed since. In this section, we will give a detailed survey of BNs and review the main concepts and algorithms developed for BN in the past thirty years. A large part of this review is based on a recent book by Darwiche (2009) which gives a comprehensive account of these developments in BN.

4.4.1 Conditional Probability

In Section 2.3.3 we introduced a semantic definition of probability that can be used to represent uncertainty. However, we usually do not have direct access to the (absolute) probability of a proposition A; rather we have the probability of A based on some evidence. Conditional probability captures the intuition that the "probability of B based on the observation of variable A". Formally, such a conditional probability, denoted as $Pr(B|A)$, is the sum of the measures of the possible worlds in which both

[9]There are various extensions to classical logics that supposed to accommodate numerical information or probabilities. However, I avoid these variations since classical logics are fundamentally qualitative and should be used where they are naturally fit.

[10]This does not mean that we accept that uncertainty can *only* represented or captured with numbers.

A and B are true. That is,

$$Pr(B|A) = \sum_{w \models A \wedge B} Pr(w|A).$$

A normalisation constant $1/Pr(A)$ for the worlds that satisfy A is used to ensure the sum of the measure of the worlds remains at 1. Hence,

$$\begin{aligned} Pr(B|A) &= \frac{1}{Pr(A)} \sum_{w \models A \wedge B} Pr(w) \\ &= \frac{Pr(A \wedge B)}{Pr(A)} \end{aligned}$$

The last form above is known as *Bayes conditioning*[11] (Pearl 1988). It is the direct consequence of the following commitments:

- Worlds that contradict evidence A have probability of zero.
- Worlds that have zero probability will always have zero probability.
- Worlds that are consistent with evidence A will maintain their relative probabilities.

A direct result from a repeated application of Bayes condition is the so-called chain rule (Pearl 1988):

$$Pr(A_1 \wedge A_2 \wedge ... \wedge A_n) = Pr(A_1|A_2 \wedge ... \wedge A_n)Pr(A_2|A_3 \wedge ... \wedge A_n)...P(A_1) \quad (4.8)$$

The rule plays an important role in the calculation of joint probability from the conditional probabilities in Bayesian networks.

4.4.2 Conditional Independence

Independence is a dynamic notion. Independent events may become dependent given new evidence; at same time, dependent events may become independent given new

[11] We assume that $Pr(A)$ will always have a positive value.

evidence. Therefore, a more general definition of independence is needed so that an event α is *conditionally independent* of event β given evidence γ (Pearl 1988).

$$Pr(\alpha|\beta \wedge \gamma) = Pr(\alpha|\gamma).$$

This means, with the evidence of γ, the probability of α does not change in light of the additional evidence of β. Since conditional independence is symmetric, we have α conditional independent of β given γ if any only if β is conditional independent of α given γ. To emphasise the symmetry, conditional independence is frequently defined as following:

$$Pr(\alpha \wedge \beta|\gamma) = Pr(\alpha|\gamma)Pr(\beta|\gamma)$$

Conditional independence plays a key role in defining the Markovian assumption (Section 4.4.3) which the Directed Acyclic Graph (DAG) is based on.

4.4.3 Bayesian Networks

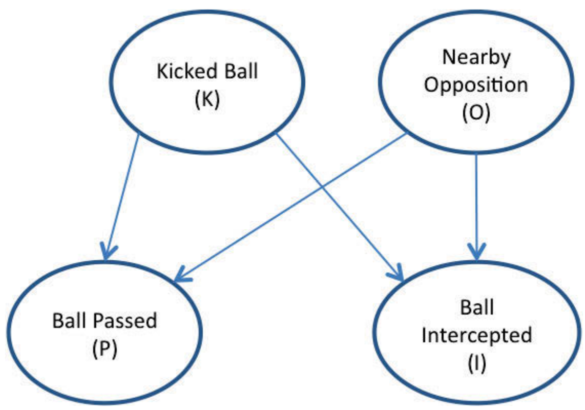


Figure 4.2: A Directed Acyclic Graph for the Ball Passing Problem.

A BN is graphically represented as a DAG (Figure. 4.2) that uses nodes[12] to represent propositional variables and directed arcs between nodes represent the dependencies between the variables. For a given node V, any node that has an arc

[12]We uses node and variable interchangeably.

point into V is a parent node of V; descendants of V are the nodes that have arcs directly from V. Mathematically, BN belongs to a class of mathematical models called Markovian Models that assume every variable in a model is conditionally independent (denoted as I)of its non-descendants variables given its parent variables are known (Pearl 1988), i.e.

$$I(V, Parents(V), Non\_Descendants(V)) \text{ for all variables } V \text{ in DAG } G.$$

The DAG of a Bayesian network graphically captures this so-called *Markovian assumption* among the variables. Each node X in the DAG is also coupled with a set of conditional probabilities $Pr(X|\mathbf{U})$[13] for every value of X and every instantiation of $\mathbf{u}$ of its parents $\mathbf{U}$. For a simple DAG shown in Figure 4.2, four conditional probabilities are needed, namely,

$$Pr(k), Pr(o), Pr(p|k,o), Pr(i|k,o).$$

where k, o, p and i are the values of the respective variables. The conditional probability for a variable P is expressed in a table form known as a *Conditional Probability Table* (CPT) for variable P. Examples of $Pr(p|k,o)$ and $Pr(i|k,o)$ are shown below:

K	O	P	$Pr(p\|k,o)$
true	true	true	0.16
true	true	false	0.84
true	false	true	0.82
true	false	false	0.18
false	true	true	0
false	true	false	1
false	false	true	0
false	false	false	1

K	O	I	$Pr(i\|k,o)$
true	true	true	0.62
true	true	false	0.38
true	false	true	0
true	false	false	1
false	true	true	0
false	true	false	1
false	false	true	0
false	false	false	1

Table 4.3: Conditional Probability Tables (CPT) for $Pr(p|k,o)$ and $Pr(i|k,o)$.

Note that the following constraints ensure that the total probability of variable P

[13]This is sometime referred as network families.

(being either true or false) remains at one, given the values of its parent variables.

$$\left.\begin{array}{r} Pr(p|k,o)+Pr(\neg p|k,o) \\ Pr(p|k,\neg o)+Pr(\neg p|k,\neg o) \\ Pr(p|\neg k,o)+Pr(\neg p|\neg k,o) \\ Pr(p|\neg k,\neg o)+Pr(\neg p|\neg k,\neg o) \end{array}\right\} = 1.$$

That is, the conditional probability of variable P is normalised. Normalisation of conditional probability reflects the fact that BN is based on the Closed World Assumption. With both the conditional probability for every variable and the Markovian independence assumption, a Bayesian Network as a whole defines an unique joint probability distribution of all variables in the network. Therefore, BN can be used to model the probability distributions in the domains that satisfy the Markovian independence assumption and associated properties[14]. Using various BN inference techniques, the probabilities of any subsets of the network variables, i.e. the marginal distributions, are defined as (Pearl 1988)

$$Pr(x_1, ..., x_m) = \sum_{x_{m+1},...,x_n} Pr(x_1, ..., x_n), \tag{4.9}$$

where $Pr(x_1, ..., x_n)$ is the joint distribution of a BN and $m \leq n$. This can be used to answer queries, using the Ball Passing Problem as an example, the probability of the ball being successfully passed to a team member in presence of a nearby opposition robot or find out the most probable explanation for a team member did not receive the ball after it was kicked by its teammate.

In the following sections, we will first give a concise discussion on the current progress in constructing Bayesian networks before moving on to a brief survey of BN inference mechanisms.

4.4.4 Bayesian Network Construction

A BN is usually constructed in three distinct steps. The first step is to define a set of variables and their (possible) values to be used in the graphical network. The

[14]Such as symmetry, decomposition etc. We omit discussion these properties.

variables can be categorised into three types: evidence, query and intermediate variables. Evidence variables are the variables provide direct observational data; whereas query variables are the variables used to ask questions. Intermediate variables are the variables that have relationships with evidence and/or query variables; they connect the evidence variables and query variables together to form a relationship map in a DAG. Using a knowledge engineering procedure for risk management (I will introduce the knowledge management procedure in Section 5.3.), evidence variables can be easily identified from the description of the initial event and environment; query variables correspond to the scenarios and any associated factors are the intermediate variables. Identification of network variables and their values is a critical step that maps out the boundary of domain knowledge; and it defines the final quality of the model constructed for the domain.

The second step is to define the structure of the network, i.e. the relationships between the variables. A causal interpretation for the network structure says variable A is a direct cause of variable B, if there is a directed arc from node A to node B (Pearl 2000). Once the network structure is determined, we need to work out the network parameters, i.e. CPT, for each variable in the network. Determining the network structure and parameters of a BN is not a simple task. For a (quantitative) data poor problem, only the inputs from domain experts are available to model the network. There is no formal systematic procedure for the model construction. The CPTs of these manually constructed models only reflect the subjective beliefs of the experts. Furthermore, increasing number of variables in the network make determining CPTs increasingly difficult. We will discuss these knowledge engineering and model building issues for data poor problems in Chapter 7 and show our new framework can provide a more systematic way to build a DAG based model from experts' knowledge. On the other hand, for data rich problems, there are a number of techniques developed in learning the network structure and estimating CPTs from data. We provide a summary of the standard techniques in the following sections.

4.4.4.1 Learning Network Parameters from Complete Data

When we have a known Bayesian network structure such as the one depicted in Figure 4.2 and a set of data samples for the variables in the network (Table 4.4 (a)), we can estimate the network parameters such as $Pr(p|k,o)$ directly from the dataset. Note that each row of the dataset represents a complete instance of observed values of the network variables. From such a dataset D, we can summarise an empirical distribution as shown in Table 4.4 (b). The empirical probability of instantiation k, o, p is simply the frequency of its occurrences in the dataset.

Case	K	O	P
1	true	true	true
2	true	false	false
3	true	false	true
4	true	true	false
5	false	true	false
6	true	false	false
7	true	true	false
8	true	true	false
9	true	false	true
10	true	true	false
11	true	false	true
12	false	false	false
13	true	false	true
14	true	false	true
15	true	false	true
16	true	true	false

(a) Complete dataset

K	O	P	$Pr_D(.)$
true	true	true	1/16
true	true	false	5/16
true	false	true	6/16
true	false	false	2/16
false	true	true	0/16
false	true	false	1/16
false	false	true	0/16
false	false	false	1/16

(b) Empirical distribution

Table 4.4: A complete set of data samples and empirical distribution for ball passing BN.

$$Pr_D(k,o,p) = \frac{D\#(k,o,p)}{N},$$

where $D\#(k,o,p)$ is the number of instances of an instantiation of k, o, p in the dataset D and N is the size of the dataset. We can estimate the network parameter

$Pr(p|k, o)$ with k, o and p are true as follows:

$$Pr(p|k, o) = \frac{Pr_D(k, o, p)}{Pr_D(k, o)} = \frac{1/16}{6/16} = 1/6.$$

Therefore, for a given BN and a complete dataset generated from the network, we can estimate a parameter $\theta_{x|\mathbf{u}}$ using the empirical probability (Darwiche 2009).

$$\theta^{ml}_{x|\mathbf{u}} = Pr_D(x|\mathbf{u}) = \frac{D\#(x, \mathbf{u})}{D\#(\mathbf{u})}. \tag{4.10}$$

Clearly, the parameter estimate $\theta_{x|\mathbf{u}}$ is dependent on the given dataset D. Increasing the size of the dataset, the distribution of estimate $\theta^{ml}_{x|\mathbf{u}}$ will asymptotically approach a normal distribution with variance of

$$\frac{Pr(x|\mathbf{u})(1 - Pr(x|\mathbf{u}))}{N Pr(\mathbf{u})},$$

according the Law of Large Numbers (Grinstead & Snell 1997). In other words, the accuracy of the estimate depends on the size of the dataset and the probability of the parent instantiation $Pr(\mathbf{u})$. Another important property for parameter estimate is the likelihood of the estimate.

$$L(\theta|D) = \prod_{i=1}^{N} Pr_\theta(\mathbf{d}_i), \tag{4.11}$$

where θ is the set of all parameter estimates for a given network structure and $\mathbf{d}_i$ is a complete instantiation of all the variables in the dataset D. $L(\theta|D)$ represents the probability of observing the dataset D under these estimates. The convenient log-likelihood function is defined as

$$LL(\theta|D) = logL(\theta|D) = \sum_{i=1}^{N} logPr_\theta(\mathbf{d}_i). \tag{4.12}$$

It turns out that the estimate defined by the empirical probabilities (Equation 4.10) is the only estimate that maximises the likelihood function. Consequently, these estimates are called maximum likelihood (ML) estimates and defined as following (Darwiche 2009)

$$\theta^{ml} = \arg\max_\theta L(\theta|D).$$

This important result implies that we can derive empirical distribution based estimates by maximising the likelihood function. In fact, when we are dealing with incomplete data, we take exactly this approach in which we seek estimates that maximise the likelihood function. In the next section, we will highlight some of the key ideas in learning parameters with incomplete data.

4.4.4.2 Learning Network Parameters from Incomplete Data

We have shown in the previous section that we can derive unique, asymptotically normal network estimates from a complete dataset. However, in practice, we often face datasets with missing values (e.g. Table 4.5). The cause of incomplete data could be that we are unable to observe a variable directly[15] or in some cases, we have missing data values due to various reasons, for example, robots have limited field of view and they may not observe the opponent or the outcome of the ball passing.

Case	K	O	P
1	true	true	?
2	true	false	?
3	true	false	?
4	true	true	?
5	false	true	?
6	true	false	?

(a) A dataset with an hidden variable.

Case	K	O	P
1	true	?	true
2	true	false	false
3	true	false	?
4	true	true	false
5	false	true	?
6	true	?	false

(b) A dataset with missing values.

Table 4.5: Examples of incomplete dataset.

All available techniques for estimating network parameters are iterative search methods that require an initial estimates θ^0 and search for new estimates that maximise the likelihood of getting the observed data. These algorithms differ only in terms of searching strategies they employ. One of the popular learning methods is

[15]Such a variable is called a hidden or latent variable.

called Expectation Maximisation (EM) (Dempster, Laird & Rubin 1977). It first instantiates all the missing values in the incomplete dataset inferenced from the initial estimate. The completed dataset defines an *expected* empirical distribution and the probability of an instantiation (or event) can be computed from the number of its occurrences in the dataset. The expected empirical distribution of dataset D under network parameter θ^i is defined as the following (Dempster et al. 1977):

$$Pr_{D,\theta^i}(\alpha) = \frac{1}{N} \sum_{\mathbf{d}_j, \mathbf{c}_j \models \alpha} Pr_{\theta^i}(\mathbf{c}_j | \mathbf{d}_j), \tag{4.13}$$

where α is an event and $\mathbf{C}_j$ are the variables with missing values in case $\mathbf{d}_j$. $\mathbf{d}_j, \mathbf{c}_j \models \alpha$ means that event α is satisfied by complete case $\mathbf{d}_j$ of $\mathbf{c}_j$. New estimates can then be computed from the expected distribution, similar to the computation in learning from the complete dataset:

$$\theta^{i+1}_{x|\mathbf{u}} = Pr_{D,\theta^i}(x|\mathbf{u}) = \frac{Pr_{D,\theta^i}(x, \mathbf{u})}{Pr_{D,\theta^i}(\mathbf{u})}, \tag{4.14}$$

The new likelihood for $\theta^{i+1}_{x|\mathbf{u}}$, i.e. expected likelihood function, is (Dempster et al. 1977)

$$L(\theta^{i+1}|D) = \prod_{j=1}^{N} Pr_{\theta^i}(\mathbf{d}_j). \tag{4.15}$$

It has been shown that the log-likelihood functions of EM estimates cannot decrease after each iteration, i.e. $LL(\theta^{i+1}|D) \geq LL(\theta^i|D)$. This means the new parameter estimates after an iteration are guaranteed to be better or equal to the estimates from which they are derived. Therefore, the EM algorithm (Algorithm 3) guarantees parameter estimates converge. It is important to note that the EM algorithm is sensitive to the initial estimates θ^0; different θ^0 may converge to different parameters. That is, we can only find a local maxima of likelihood function. It is common to run the EM algorithm multiple times and choose different initial estimates each time to ensure we find an optimal maxima of the likelihood function. Another issue of the EM method is its slow convergence when a large fraction of data is missing.

Algorithm 3 The EM Algorithm (Lauritzen 1995)

Require: Bayesian network structure G with families $X\mathbf{U}$. Initial parameterisation of G, θ^0. Dataset of size N, D.
Ensure: EM parameter estimates for structure G.

```
k = 0
while θ^k ≠ θ^{k-1} do
    c_{xu} = 0 for each family instantiation xu
    for i = 1 to N do
        for each family instantiation xu do
            c_{xu} = c_{xu} + Pr_{θ^k}(xu|d_i) (from inference on (G, θ^k).
        end for
    end for
    compute parameter θ^{k+1} from θ^{k+1}_{x|u} = c_{xu} / Σ_{x*} c_{x*u}
    k = k + 1
end while
return θ^k
```

An alternative approach to the EM algorithm is to treat parameter learning as a problem of optimising a non-linear function. Again, we start with an initial parameter estimate $\theta^0_{x|\mathbf{u}}$ and step through the parameter space in searching for a maximum likelihood function by an increment of $\delta^i_{x|\mathbf{u}}$, i.e. $\theta^{i+1}_{x|\mathbf{u}} = \theta^i_{x|\mathbf{u}} + \delta^i_{x|\mathbf{u}}$. The value for the increment $\delta^i_{x|\mathbf{u}}$ is usually determined using gradient information. Specifically, we need to maximise the log-likelihood function $LL(\theta|D)$ based on the gradient $\delta LL(\theta|D)/\delta\theta_{x|\mathbf{u}}$ while maintaining the constraints $\theta_{x|\mathbf{u}} \in [0, 1]$ and $\sum_x \theta_{x|\mathbf{u}} = 1$. It has been shown that (Russell, Binder, Koller & Kanazawa 1995)

$$\frac{\delta LL(\theta|D)}{\delta\theta_{x|\mathbf{u}}} = \sum_{j=1}^{N} \frac{Pr_\theta(x\mathbf{u}|\mathbf{d}_j)}{\theta_{x|\mathbf{u}}},\ when\ \theta_{x|\mathbf{u}} \neq 0. \tag{4.16}$$

This means we can calculate the gradient from the family marginals using inference algorithms (Section 4.4.5) without the need to compute derivatives. The performance of the gradient-based algorithms is significantly influenced by the learning rate. Selecting a proper learning rate will greatly speed up the convergence. In terms of computational complexity, the gradient-based algorithms are at the same level as the EM algorithm, since both approaches require computation of all family marginals for every distinct cases in the dataset as shown in Equation 4.13 and 4.16. Next, I will turn my attention to methods for discovering the structure of a BN from data.

4.4.4.3 Learning Network Structures

Previously, we assume the structure of a BN is known and we have a set of corresponding data for the network. We estimate the values of the network parameters that maximise the probability of observing the given dataset. In this section, we take a similar approach to find the network structure itself from data. That is, we search for network structures that maximise the probability of observing the given dataset. We define a log-likelihood function for structure G, $LL(G|D)$, similar to the log-likelihood function for network parameter estimates $LL(\theta|D)$. It has been shown that $LL(G|D)$ can be effectively decomposed into a number of subcomponents, one for each family $X|\mathbf{U}$ in the Bayesian network structure (Darwiche 2009):

$$LL(G|D) = logL(G|D) = -N \sum_{X|\mathbf{U}} ENT_D(X|\mathbf{U}), \tag{4.17}$$

where D is a complete dataset of size N and $ENT_D(X|\mathbf{U})$ is the conditional entropy defined as

$$ENT_D(X|\mathbf{U}) = -\sum_{x|\mathbf{u}} Pr_D(x|\mathbf{u}) log_2 Pr_D(x|\mathbf{u}).$$

An effective way to search for a network structure that maximises the likelihood value is to decompose the network, starting with an empty set of parents for a variable X and successively adding variable U_i to the set with the edge $U_i \rightarrow X$. However, it turns out, by adding more parents and the corresponding edges to a variable, we never decrease the log-likelihood of the resulting structure. The search for a structure that maximises the log-likelihood function will end up with a complete DAG in which no more edges can be added. This complete DAG structure reveals no useful feature of the Bayesian model and is practically useless in terms of inference and reasoning. A complete DAG represents an *overfit* of the known data. A common approach for avoiding the problem of overfitting is to invoke the principle of *Occam's Razar*. That is, we should prefer simpler models over more complex models. A better scoring function than the log-likelihood function for determining an optimal structure for a dataset is needed. A simple solution is to add a penalty term to the existing log-likelihood function (Bouckaert 1993):

$$Score(G|D) = LL(G|D) - \psi(N) * ||G||, \tag{4.18}$$

where $\psi(N)$ is a penalty weight and $||G||$ is the dimension of a DAG G defined as

$$\begin{aligned} ||G|| &= \sum_{i=1}^{N} ||X_i \mathbf{U}_i|| \\ ||X_i \mathbf{U}_i|| &= (X_i^{\#} - 1)\mathbf{U}_i^{\#}, \end{aligned}$$

in which $X_i^{\#}$ denotes the number of instantiations of variable X. The $Score(G|D)$ can be further decomposed as

$$Score(G|D) = \sum_{X\mathbf{U}} Score(X_i, \mathbf{U}_i|D), \tag{4.19}$$

where

$$Score(X_i, \mathbf{U}_i|D) = -N * ENT_D(X_i|\mathbf{U}_i) - \psi(N) * ||X_i \mathbf{U}_i||. \tag{4.20}$$

Clearly, the penalty term $\psi(N)*||G||$ depends on the number of independent variables in DAG G. This term acts as a counter-balance to the log-likelihood function so that simpler models are preferred over more complex models even though the more complex models have higher log-likelihood values. The penalty weight $\psi(N)$ in turn controls the impact of the penalty term. Therefore, we can adjust our model preferences by changing the penalty weight. One of the popular choices for $\psi(N)$ is $\frac{log_2 N}{2}$ because it allows the penalty term to dominate the score function while N is small but its influence decreases as N grows. The score function based on this penalty weight is called *minimum description length* (Rissanen 1978) score[16]. To further reduce the search space for the network structure, we assume a total ordering on the network variables so that we can exhaustively search network structures for variables in order.

[16]It is also known as *Bayesian information criterion*.

Algorithm 4 The K3 Algorithm

Require: Node X_i in a total order of nodes.
Ensure: $\mathbf{U}_i$ with maximum $Score(X_i, \mathbf{U}_i|D)$ for variable X_i.

$Pred_i = \{X_1, X_2, ..., X_{i-1}\}$
Parents $\mathbf{U}_i = \phi$
$Score_{max} = Score(X_i, \mathbf{U}_i|D)$
while $\mathbf{U}_i \neq Pred_i$ **do**
    Let X_j be the node in the $Pred_i \setminus \mathbf{U}_i$ that maximises $Score(X_i, \mathbf{U}_i \cup X_j|D)$
    $Score_{new} = Score(X_i, \mathbf{U}_i \cup X_j|D)$
    if $Score_{new} > Score_{max}$ **then**
        $Score_{max} = Score_{new}$
        $\mathbf{U}_i = \mathbf{U}_i \cup X_j$
    else return $\mathbf{U}_i$ and $Score_{max}$
    end if
end while

The algorithm based on this approach is called the $K3$ algorithm (Bouckaert 1993). It is a greedy algorithm that produces a set of parents for variable X_i that maximises the score function $Score(X_i, \mathbf{U}_i|D)$, i.e. local optimal. To identify the optimal set of network structures, we need optimal search algorithms in which the $K3$ algorithm is used as a starting point. A notable optimal search algorithm is based on branch-and-bound depth first search. It first uses the $K3$ algorithm to compute the maximum score for every variable in the network. The entropy values from the maximum scores are then used to arrange the network variables in a tree order. The depth first search starts from the node with the lowest entropy and moves to the nodes with higher entropy. At each step, a single variable is added to the parent set of node X and evaluates $Score(X_i, \mathbf{U}_i|D)$. The score is compared with the best score the search has obtained so far. Since the depth first search ensures every parent set is visited, search is guaranteed to terminate with an optimal parent set. This algorithm can be further optimised by applying an upper bound of score calculated based on the largest parent sets rooted at X_i to the search tree and pruning any subtree that does not return a better score than the best one found so far. The Algorithm 5, uses this optimisation to improve its search efficiency.

Algorithm 5 The Depth First Branch-and-Bound Search Algorithm with $K3$

Require: Node X_i

Ensure: Parent set $\mathbf{U}_i$ maximum $Score(X_i, \mathbf{U}_i|D)$ for variable X_i.

Execute $K3$ and produce $\mathbf{U}_i$ with $Score_{max}$.
Arrange variables in the tree order of $\{X_{k_1}, ..., X_{k_{i-1}}\}$ such that $ENT_D(X_i|X_{k_1}) \leq ... \leq ENT_D(X_i|X_{k_{i-1}})$.
Penalty $p_1 = \frac{log_2 N}{2} * (X_i^{\#} - 1)$; $T = \phi$.
Call procedure DFBnB(T, X_{k_0}, p_1).

procedure DFBnB(T, X_{k_j}, p)
    $Score = -N * ENT_D(X_i|T) - p$.
    if $Score > Score_{max}$ **then**
        $Score_{max} = Score$ and $\mathbf{U}_i = T$.
    end if
    Let W be the set of variables after X_{k_j} in the tree order.
    $Bound_{upper} = -N * ENT_D(X_i|T, W)$.
    for all $X_q \in W$ **do**
        $p_{new} = p * X_q^{\#}$.
        if $Score_{max} < Bound_{upper} - p_{new}$ **then**
            Call procedure DFBnB$(T \cup \{X_q\}, X_q, p_{new})$
        end if
    end for
end procedure

4.4.5 Inferences in Bayesian Networks

The main use of Bayesian Networks is to provide answers to queries through network inference. In other words, BN can compute the marginal distribution of a (set of) target variables using network inference. The marginal distribution of target variables can be calculated without any data (or evidence). This is known as *prior marginal*. At the same time, we can also ask for the probability of the target variables based on data from some observed variables (or evidence), i.e. posterior marginal. For example, network inference can be used to estimate the likelihood of ball interception without any given data. On the other hand, by observing the ball being kicked by robot R_A and the distance between R_A and its teammate R_B, a BN can be used to predict on the probability of a successful pass. In the case of detecting an unsuccessful ball pass, the robot can also draw up the most likely explanation using BN that the ball was intercepted by an opponent and subsequently take an appropriate response. Because of these practical uses of network inference, the majority of work in BN is related to either the development of new efficient network inference algorithms or the improvement of existing inference algorithms. Bayesian network inference algorithms can be grouped into two categories: exact inference and approximate inference. I will provide a broad summarisation of both types of algorithms in the following section.

4.4.5.1 Exact Inference

Exact Bayesian network inference algorithms can be summarised into three types: inference by variable elimination, inference by factor elimination and inference by conditioning. We give a slightly more detailed summaries for the first two approaches and briefly touch on the last approach.

Inference by Variable Elimination (VE) is normally used to calculate prior and posterior marginal of a set of target variables in a Bayesian network (Zhang & Poole 1994). For example, we can use the VE method to calculate the marginal of P in the Ball Passing Problem. The basic idea of VE is to construct a marginal probability distribution over the target variables, in our case variable P, by summing

Algorithm 6 Prior Marginal Calculation with Variable Elimination

Require: Bayesian network G; target variables $\mathbf{Q}$ and π ordering of variables not in $\mathbf{Q}$.

Ensure: The prior marginal $Pr(\mathbf{Q})$.

$S =$ CPTs of G.
for $i = 1$ to the length of π **do**
    $f = \prod_j f_j$, where $f_j \in S$ and contains variables $\pi(i)$.
    $f_i = \sum_{\pi(i)} f$.
    Replace all factors f_j in S with f_i
end for
return $\sum_{f \in S} f$

Algorithm 7 Posterior Marginal Calculation with Variable Elimination

Require: Bayesian network G; target variables $\mathbf{Q}$; $\mathbf{e}$ instantiation of some variables in G and π ordering of variables not in $\mathbf{Q}$.

Ensure: The joint marginal $Pr(\mathbf{Q}, \mathbf{e})$.

$S = \{f^{\mathbf{e}} : f$ is CPT of G and $f^{\mathbf{e}} = f(\mathbf{x})$ if $\mathbf{x} \sim \mathbf{e}\ \}$.
for $i = 1$ to the length of π **do**
    $f = \prod_j f_j$, where $f_j \in S$ and contains variables $\pi(i)$.
    $f_i = \sum_{\pi(i)} f$.
    Replace all factors f_j in S with f_i
end for
return $\sum_{f \in S} f$

up the remaining variables in the given network, i.e. variables K, O and I (Figure 4.2). Specifically, the remaining variables are *eliminated* one by one when all rows in the network joint probability distribution that agree on the values of all variables are merged except for the variable being eliminated. Once all the non-target variables are eliminated from the distribution, we are left with the marginal probability distribution of the target variables. It is not necessary to construct the joint probability distribution used in VE since the joint distribution can be factored into a set of conditional probabilities according to the chain rule (Equation 4.8). Such a conditional probability is a *factor* function over variables $\mathbf{X}$ that maps each instantiation of the variables $\mathbf{X}$ to a non-negative number, denoted as $f(\mathbf{x})$. Summing out a variable X from a factor f over variables $\mathbf{X}$ is another factor over variables $\mathbf{X} \setminus \{X\}$ denoted as $\sum_x f$. The process of summing out every variable in variables $\mathbf{X}$ is called *marginalising* variables $\mathbf{X}$. The order of variable elimination has no effect on the end results since the summation operation is commutative. Another key property of the summation operation is that

$$\sum_X f_1 f_2 = f_1 \sum_X f_2,$$

if the variable X appears only in f_2. This means, it is not necessary to multiply all the factors together before the summation operations when some factors do not have the variables to eliminate. It also means summing out variables as early as possible and as many as possible will greatly reduce the size of the factors and the computation required for factor multiplication. Therefore, selecting which variables should be eliminated earlier, i.e. the order of variable elimination, can have significant impact on the computation cost. The Algorithm 6 and 7 listed above show the calculation of prior marginal $Pr(\mathbf{Q})$ and joint marginal of $Pr(\mathbf{Q}, \mathbf{e})$ using VE respectively; and Algorithm 8 below presents a method of choosing an variable elimination order by always eliminating the variable that leads to constructing the smallest factor possible.

Inference by Factor Elimination (FE) is a variation of the variable elimination method. Instead of eliminating the non-target variables one at a time, the FE

Algorithm 8 Minimum Degree Elimination Order

Require: Bayesian network G; variables $\mathbf{X}$ in G.
Ensure: The ordering π of variables $\mathbf{X}$.

I =InteractionGraph(G)
for $i = 1$ to number of variable in $\mathbf{X}$ **do**
  $\pi(i)$ = a variable in $\mathbf{X}$ with smallest number of neighbours in I.
  Add an edge between every pair of non-adjacent neighbours of $\pi(i)$ in I.
  Delete variable $\pi(i)$ from I and $\mathbf{X}$.
end for
return π
function InteractionGraph(G)
  Graph I with all variables appear in G and no edges between any variables.
  for factor $f = f_1$ to f_n **do**
    Connect all variables appear in f pair-wisely in I.
  end for
return an undirected graph I
end function

algorithm eliminates factors that do not contain the target variables step by step. Generally, the factor elimination process follows a few basic steps:

1. Organise factors of a Bayesian network into a tree structure (e.g. Figure 4.3) with a factor that contains the target variables selected as the root node (i.e. double circled node).

2. Choose a leaf node factor from the tree for elimination.

3. The selected factor is eliminated by summing up all variables that exist only in the factor and then multiply into the nearest neighbour of the factor.

4. Repeat above two steps till all the leaf nodes are removed from the tree. The elimination then moves to the new leaf nodes and continues the process untill it reaches the root node.

5. After the root node is reached, the marginal of the target variables can be computed from the projection over the target variables.

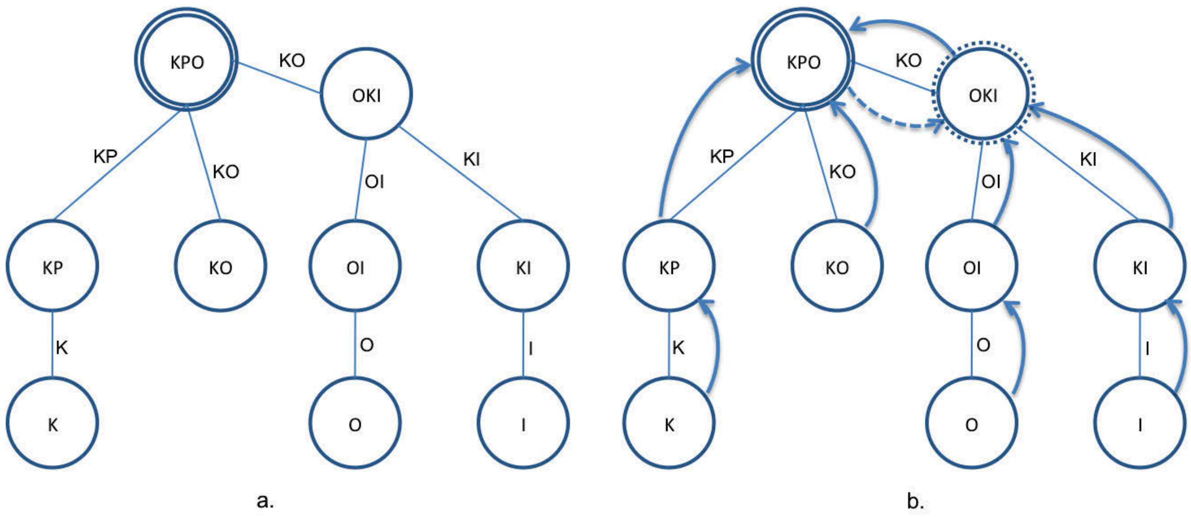


Figure 4.3: A (nonminimal) jointree for the Bayesian network shown in Figure 4.2.

Note that the variable summation operations used in step 3 and 5 are factor project operations which are defined as

$$project(f, \mathbf{Q}) = \sum_{vars(f)\setminus\mathbf{Q}} f$$

This project operation simply sums out all variables in factor f but not in $\mathbf{Q}$. The prominent feature in this FE approach is that a tree structure is used for the organising the elimination process. This tree structure plays a similar role as the elimination order used in VE algorithms. The most commonly used tree structure is so-called the *jointree* which is defined as follows:

Definition. *(Lauritzen & Spiegelhalter 1988) A jointree for a DAG G is a pair* $(\mathcal{T}, \mathbf{C})$ *where* $\mathcal{T}$ *is a tree and* $\mathbf{C}$ *is a function that maps each node in* i *in tree* $\mathcal{T}$ *to a cluster* $\mathbf{C}_i$*. A jointree must satisfy the following properties,*

- *The cluster* $\mathbf{C}_i$ *consists of a set of nodes from* G*.*
- *Each family* $X|\mathbf{U}$ *of the Bayesian network must appear in some cluster* $\mathbf{C}_i$*.*

- *If a node appears in two clusters $\mathbf{C}_i$ and $\mathbf{C}_j$, it must also appear in every cluster $\mathbf{C}_k$ on the path between nodes i and j in the jointree. This property is also known as the jointree property or the running intersection property.*

The intersection of two connecting clusters i and j in a jointree, i.e. $\mathbf{C}_i \cap \mathbf{C}_j$, is the separator of edge $i - j$, denoted as $\mathbf{S}_{ij}$.

The standard method for constructing a jointree was first introduced by Lauritzen and Spiegelhalter (1988). It first turns a DAG into a so-called moral graph in which all parent nodes of a child node are connected pairwise and the direction on all graph edges are dropped. The resulting undirected graph is further triangulated (using maximum cardinality search) such that for every cycle of length $n \geq 4$ there must be a chord which connects a pair of nonconsecutive nodes in the cycle. A fully triangulated graph is then broken up into a set of cliques[17] and reordered into a tree structure according to the maximum note in each. Such a tree structure satisfies the intersection property of the jointree (Lauritzen & Spiegelhalter 1988) and is, therefore, a proper jointree. The full jointree construction procedure is listed in Algorithm 9 below.

The corresponding FE algorithm that uses jointree is called the Jointree Algorithm and is listed in Algorithm 10.

Notice that, at step 4 of Algorithm 10, the factor of node j is updated by the projection of its neighbour i, after i is removed from the tree. We can view this process as node i passing a message to its single neighbour j, while it is being eliminated. In addition, node i cannot pass its message to j until it receives all messages from its neighbours $k \neq j$ (otherwise, it does not satisfy the criteria at the step 3). In fact, the entire FE process can be viewed as messages computed and propagated from the leaf nodes to some root node of a tree (Pearl 1988). A significant advantage of this message passing formulation is that the message computed at each node can be reused when we want to change the root node and compute a different marginal. For

[17] A clique is a maximal subset of nodes that have edges between all nodes.

Algorithm 9 Construction of a Jointree

Require: A Bayesian DAG network G.
Ensure: A jointree $(\mathcal{T}, \mathbf{C})$ for G.

G' =MoralGraph(G)
Ordering α =MaximumCardinalitySearch(G')
G'' =FillingComputation(G', α)
Let $\mathbf{C}(G'')$ = a set of cliques of G'' (they are clusters).
Order cliques according to the maximum note and connect into a tree $(\mathcal{T}, \mathbf{C})$.
return The jointree $(\mathcal{T}, \mathbf{C})$.
function MoralGraph(G)
    Let $var(G)$ be a set of variables in G.
    for all $X_i \in var(G)$ **do**
        for all $X_j, X_k \in Parents(X_i)$ **do**
            Connect X_j and X_k with an undirected edge.
        end for
    end for
    Remove directions in all directed edges between variables.
return an undirected graph.
end function
function MaximumCardinalitySearch(Undirected graph G)
    Let $var(G)$ be a set of variables in G and $edge(G)$ be a set of edges in G.
    for $i = 1$ to number of nodes in G - 1 **do**
        Let $set(i) = \phi$
    end for
    for all $v \in var(G)$ **do**
        Let $size(v) = 0$ and add v to $set(0)$.
    end for
    Let $i = n$ and $j = 0$.
    while $i \geq 1$ **do**
        v = extract any node from $set(j)$.
        $\alpha(v) = i$; $\alpha^{-1}(i) = v$ and $size(v) = -1$.
        for all $\{v, w\} \in edge(G)$ and $size(w) \geq 0$ **do**
            Delete w from $set(size(w))$;
            Decrement $size(w)$ by 1 and add w to $set(size(w))$.
        end for
        Decrement i by 1 and increment j by 1.
        while $j \geq 0$ and $set(j) = \phi$ **do**
            Decrement j by 1.
        end while
    end while
return An ordering α.
end function

example, in Figure 4.3 (b), when we compute the marginal for OKI, i.e. moving the root node from KPO to OKI, most of the messages computed from calculating the marginal KPO are still valid for computing the OKI marginal. Only one additional message from node KPO to OKI needs to be computed (shown as the dashed-line in Figure 4.3). In other words, the amount of computations required for calculating multiple marginals can be greatly reduced by caching the message computed at each node of a jointree. This is the key advantage that FE algorithms offer over the VE algorithms. We can compute the marginals over every cluster of a jointree with a *pull and push* message passing scheme. Specifically, we first select a root node from the jointree and collect (or pull) messages from all its leaf nodes. We then distribute (or push) message from the root node towards the leaf nodes. The entire computation for every marginals requires total $2(m-1)$ messages[18] and the computation of every message bounded by $O(exp(w))$ where w is the width of the jointree[19]. This means, the total computation complexity of FE methods is $O(n \cdot exp(w))$, compared with $O(n^2 \cdot exp(w))$ for the VE methods.

Algorithm 10 The Jointree Algorithm for Prior Marginal Calculation

Require: A Bayesian network G; target variables $\mathbf{Q}$; jointree $(\mathcal{T}, \mathbf{C})$ for the CPTs of network G and ϕ_i is the factor assigned to the cluster $\mathbf{C}_i$.
Ensure: The prior marginal of $Pr(\mathbf{Q})$.

```
1: Choose a root node r in tree T where Q ⊆ C_r.
2: while tree T has more than one node do
3:     Remove a node i ≠ r having a single neighbour j from tree T, i.e. a leaf node.
4:     φ_j = φ_j * project(φ_i, S_ij)
5: end while
6: return project(φ_r, Q)
```

[18]With m nodes, $m-1$ edges and two messages per edges.
[19]Defined as the size of the largest cluster minus one.

Algorithm 11 Filling Computation Algorithm for Jointree construction.

```
function FillingComputation(Undirected graph G, α)
    for i=1 to number of nodes in G do
        Let w = α^{-1}(i); f(w) = w and index(w) = i.
        for all {v, w} ∈ edge(G) and α(v) < i do
            Let x = v
            while index(x) < i do
                index(x) = i and add {x, w} to edge(G) ∪ FillIn(α) (add fill-in edge).
                x = f(x)
            end while
            if f(x) = x then
                f(x) = w.
            end if
        end for
    end for
return G with edges of edge(G) ∪ FillIn(α).
end function
```

4.4.5.2 Approximate Inference

All exact inference methods for BN have computational complexity in the order of $O(exp(w))$. Therefore, in practice, they are only suitable for relatively small networks with relatively small tree width. For any arbitrary large scale network of thousands variables, the practical usefulness of exact inference algorithms is severely constrained by limited computational resources available. Consequently, an active branch of BN research is the development and improvement of approximate mechanisms for BN inference. System designers we can make appropriate tradeoffs between the accuracy and computational requirement of network inference. Due to the physical limitation of this review, I will discuss only one mature class of approximation methods based on stochastic sampling.

Inference with Stochastic sampling is a method of estimating the probability of some event in a Bayesian network by generating a series of random instantiations of the network variables according to the network probability distribution and counting the fraction of instantiations at which the event is true. For example, to estimate the

	K	O	P	I
$\mathbf{x}^1$	true	false	false	false
$\mathbf{x}^2$	false	false	false	false
$\mathbf{x}^3$	true	true	false	false
$\mathbf{x}^4$	true	false	true	false
$\mathbf{x}^5$	true	false	true	false
$\mathbf{x}^6$	true	false	false	false
$\mathbf{x}^7$	true	false	true	false
$\mathbf{x}^8$	true	true	false	true
$\mathbf{x}^9$	true	true	false	true
$\mathbf{x}^{10}$	true	true	false	false

Table 4.6: A set of samples generated from network Figure 4.2.

probability of ball interception by an opponent robot, i.e. $i = true$, we generate 10 random simulated situations (Table 4.6) by traversing through the network illustrated in Figure 4.2 from parents to children. At each node X, we draw a instantiation of the variable X according to its conditional probability $Pr(X|\mathbf{u})$ where $\mathbf{u}$ is an instantiation of parent nodes of X. We then count the number of samples that have $i = true$ and the estimate for $P(i = true)$ is 2/10. To estimate the conditional probability $Pr(i = true|o = true)$, we first estimate $Pr(i \wedge o = true)$ and $Pr(o = true)$ from the samples and then calculate the ratio $Pr(i \wedge o)/Pr(o)$, i.e. 1/2. This procedure is called *direct sampling* and shown in Algorithm 12. Direct sampling is a special instance of the general stochastic sampling approach in which the probability of an event is formulated using expected value of some function $f(\mathbf{X})$ (DeGroot & Schervish 2002), i.e.

$$Ex(f) = \sum_{\mathbf{x}} f(\mathbf{x}) \cdot Pr(\mathbf{x}),$$

and the quality of estimation is measured by the variance:

$$Var(f) = \sum_{\mathbf{x}} (f(\mathbf{x}) - Ex(f))^2 \cdot Pr(\mathbf{x}).$$

For direct sampling of an event α, we define a function $\alpha'(\mathbf{X})$ that maps each instantiation $\mathbf{x}$ to a number as following:

$$\alpha'(\mathbf{x}) = \begin{cases} 1, & \text{if } \alpha \text{ is true in instantiation } \mathbf{x} \\ 0, & \textit{otherwise}. \end{cases}$$

The corresponding expectation and variance are:

$$\begin{aligned} Ex(\alpha') &= Pr(\alpha) \\ Var(\alpha') &= Pr(\alpha)Pr(\neg\alpha). \end{aligned}$$

The main reason for using expectation functions in stochastic sampling is to utilise many expectation related theorems to control the quality of the approximation outcomes. One of the key theorems, the central limit theorem (DeGroot & Schervish 2002) states that

Theorem 1. *As the sample size n tends to infinity, the sample distribution of a function $f(\mathbf{x})$ converges to a normal distribution with the sample mean, defined as $Avg_n f(x) = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}^i)$, and sample variance, defined as $S_n^2(f) = \frac{1}{n-1}\sum_{i=1}^{n}(f(\mathbf{x}^i) - Avg_n(f))^2$, asymptotically approaches the expectation μ and variance σ^2 of function $f(\mathbf{x})$.*

This result implies that the accuracy of estimates for the expectation value of function $f(\mathbf{x})$ monotonically improves with the increase of sample size. Furthermore, the quality of probability estimation in the sampling algorithm can improve by careful selection of function $f(\mathbf{x})$ with tighter variance σ^2. Reducing the variance is particularly important when estimating the probability of rare events or estimating probabilities that are conditioned on rare events. The standard method of dealing with such problems is to sample from an alternative distribution $\widetilde{Pr}(\mathbf{x})$ that focuses on the instantiations consist with the rare event. This technique is called *importance sampling* (Rubin et al. 1988). It introduces an importance sampling function for event

Algorithm 12 Approximate Inference with Direct Sampling(Darwiche 2009).

Require: A Bayesian network N; event α for probability estimation and n sample size.

Ensure: An estimation of the probability $Pr(\alpha)$.

$P = 0$.
for $i = 0$ to n **do**
    $\mathbf{x}$ =SimulateBN(N)
    if α is true in instantiation of $\mathbf{x}$ **then**
        $P = P + 1$.
    end if
end for
return P/n.
function SimulateBN(Bayesian Network N)
    Let π = a total order of network variables with parents precede their children.
    Let n is the number of network variables.
    Let Σ be the trivial variable instantiation.
    for $i = 0$ to n **do**
        Let X be the variable at position i in order π.
        $\mathbf{u}$ = values of parents of X in instantiation Σ.
        x = value of X sampled from probability distribution of $Pr(X|\mathbf{u})$.
        Append value x to instantiation Σ.
    end for
return Σ.
end function

α, denoted as $\widetilde{\alpha}$, and is defined as follows

$$\widetilde{\alpha}(\mathbf{x}) = \begin{cases} Pr(\mathbf{x})/\widetilde{Pr}(\mathbf{x}), & \text{if } \alpha \text{ is true in instantiation } \mathbf{x} \\ 0, & otherwise, \end{cases}$$

where $Pr(\mathbf{x})$ is the true distribution induced by the network and $\widetilde{Pr}(\mathbf{X})$ is the importance distribution that emphasises the instantiations consistent with event α. $\widetilde{Pr}(\mathbf{X})$ must satisfy the condition that $\widetilde{Pr}(\mathbf{x}) = 0$ only if $Pr(\mathbf{x}) = 0$ for all instantiations $\mathbf{x}$ in which α is true. The expectation and variance of an importance sampling function $\widetilde{\alpha}$ with respect to the importance distribution $\widetilde{Pr}(\mathbf{X})$ are

$$\begin{aligned} Ex(\widetilde{\alpha}) &= Pr(\alpha) \\ Var(\widetilde{\alpha}) &= \sum_{\mathbf{x} \models \alpha} \frac{Pr(\mathbf{x})^2}{\widetilde{Pr}(\mathbf{x})} - Pr(\alpha)^2. \end{aligned}$$

This means, as long as the importance distribution assigns a greater probability to event α, i.e. $\widetilde{Pr}(\mathbf{x}) > Pr(\mathbf{x})$ for all instantiations $\mathbf{x}$ implicate event α, the variance $Var(\widetilde{\alpha})$ is guaranteed to be less than the variance of direct sampling $Pr(\alpha)Pr(\neg\alpha)$. In terms of estimating conditional probability $Pr(\alpha|\beta)$ based on rare event β, the variance is no longer dependent on $Pr(\beta)$ but the importance distribution $\widetilde{Pr}(\beta)$ such that

$$Var(Pr(\alpha|\beta)) = \frac{Pr(\alpha|\beta)Pr(\neg\alpha|\beta)}{n\widetilde{Pr}(\beta)}.$$

Clearly, the key element in reducing the variance when using importance sampling is the selection of importance distribution $\widetilde{Pr}(\mathbf{x})$ which is non-trivial in practice. One popular form of importance sampling, the so-called likelihood weighting method, that uses $\widetilde{Pr}(\mathbf{x})$ induced from a modified network based on the available evidences. This modified network $\widetilde{N}$, i.e. likelihood-weighting network (Figure 4.4), is obtained from the original network N by deleting edges going to nodes $\mathbf{E}$ that have been instantiated with evidence $\mathbf{e}$. The corresponding CPTs for these nodes are reset such that for $E \in \mathbf{E}$ is instantiated to e in evidence $\mathbf{e}$ $\widetilde{\theta}_e = 1$ and $\widetilde{\theta}_e = 0$ otherwise.

It has been shown (Shachter & Peot 1989) that for a network N and its corresponding likelihood network $\widetilde{N}$ based on evidence $\mathbf{e}$, the respective distribution $Pr(\mathbf{X})$

and $\widetilde{Pr}(\mathbf{X})$ has the following property:

$$Pr(\mathbf{x})/\widetilde{Pr}(\mathbf{x}) = \prod_{\theta_{e|\mathbf{u}}} \theta_{e|\mathbf{u}} \leq 1,$$

where $\theta_{e|\mathbf{u}}$ is the network parameters range over variable $E \in \mathbf{E}$ and $e|\mathbf{u}$ is consistent with instantiation $\mathbf{x}$. This result implies that the variance of likelihood weighting cannot be larger than the variance of direct sampling. The ratio $Pr(\mathbf{x})/\widetilde{Pr}(\mathbf{x})$ can be efficiently computed for any instantiation $\mathbf{x}$ from the product of all network parameter $e\mathbf{u}$ of the CPTs of evidence variables that are consistent with $\mathbf{x}$. The result sampling algorithm is shown in Algorithm 13.

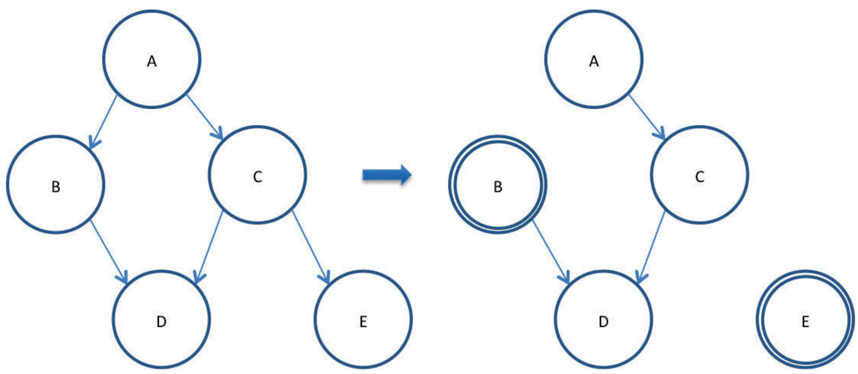


Figure 4.4: Conversion of a Bayesian network (left) to a likelihood weighting network (right) based on evidence on variables B and E. The CPTs for B and E are modified such that, for example, if $B = b$ and $E = \bar{e}$, then $\theta_b = 1$, $\theta_{\bar{b}} = 0$ and $\theta_e = 0$, $\theta_{\bar{e}} = 1$. This figure is taken from Darwiche (2009).

4.4.6 *Limitations of Bayesian Networks

Bayesian networks have become one of the dominant technologies in modelling and managing uncertain information in the last thirty years. Despite its success, it possesses several inherent limitations that make Bayesian network unsuitable for certain domains and environments. First, as a probabilistic model, BN relies on the Closed

Algorithm 13 Approximate Inference with Likelihood Weighting.

Require: A Bayesian network N; evidence $\mathbf{e}$ and n sample size.
Ensure: An estimation of $Pr(\mathbf{e})$ and $Pr(x|\mathbf{e})$ for each value x of variable X.

let $\widetilde{N}$ be the likelihood weight network for N with evidence $\mathbf{e}$.
let $P = 0$ (estimate for $Pr(\mathbf{e})$).
let $P[x] = 0$ for each value x of variable X. (estimate for $Pr(x, \mathbf{e})$).
for $i = 1$ to n **do**
    $\mathbf{x} =$ SIMULATEBN(N)
    $W = \prod \theta_{e|\mathbf{u}}$, where $E \in \mathbf{E}$ and $e\mathbf{u} \sim \mathbf{x}$.
    $P = P + W$.
    $P[x] = P[x] + W$ for each variable X and its values x consistent with $\mathbf{x}$.
end for
return P/n and $P[x]/P$.

World Assumption. It assumes complete knowledge of the domain in terms of its variables and their associated data. Various techniques have been developed (such as the method described in Section 4.4.4.2) for dealing with missing values and incorporation of latent variables for dealing with unknown variables, Bayesian network cannot explicitly capture and represent ignorance. For example[20], we may have good knowledge on the probability of bronchitis B caused by dyspnoea D, hence the conditional probability $Pr(B|D)$. However, $Pr(B|\bar{D})$ is not well-defined and its nature is unknown to us. $Pr(B|\bar{D})$ is not a suitable representation for the situations that bronchitis is caused by factors other than dyspnoea. Therefore, BN is not well suited for domains that are either not well defined or well understood. Furthermore, BN is not conductive to knowledge revision. Refinements of a Bayesian network, e.g. (Buntine 1991) only improve the probabilistic model by refining the network structure and parameters with additional data. However, it cannot cope with introduction of new variables or deletion of existing variables in the system.

Second, BN is restricted to probabilities as its inputs. Many complex environments have information that is difficult to formulate into probability (i.e. does not satisfy the axioms of probability). Proponents of BN often insist that BN "is not

[20]This is an example that appeared in the reviewer feedback section of Lauritzen (1988) paper made by Smets.

about numbers or probability" but a non-monotonic inference mechanism based on conditional probabilities. However, in a wide range of practical domains that have qualitative information or a mixture of qualitative and quantitative information, BN construction heavily relies on the inputs from domain experts. Without a clear general guideline for formulating probability for variables that lack quantitative data support, domain experts are often reluctant to give inputs or give incorrect inputs (due to misunderstanding of BN)[21] to the required prior and conditional probabilities.

Third, automated network construction (learning) and its refinements require considerable amounts of meaningful data (Zuk, Margel & Domany 2006). In many environments, such as the robotic soccer for the Ball Passing Problem, obtaining sufficient quantitative data is infeasible since opportunities of competing with a realistic opponent are rare. This makes automated model construction impossible. Finally, it is difficult to directly and fully integrate classical logics with probabilistic model such as BNs, although some steps have been taken in that direction (Richardson & Domingos 2006).

I conclude this brief survey on BN with a view that Bayesian probabilistic model is an important and well-developed technique for managing uncertainty information and performing non-monotonic reasoning. I adopt BN as one of the major components in my risk management framework. However, due its inherent limitations, BN cannot fully satisfy the framework requirements (Section 3.2) that the new framework intends to address. I need to further explore alternative methods that can be utilised in combination with BN in order to cover a wide range of problem domains.

4.5 Transferable Belief Model

The Transferable Belief Model (TBM) is a model for representing quantified beliefs for uncertainty (Smets 1998). Beliefs could be quantified by credibility, subjective

[21] Anecdotal evidence based on the spontaneous discussion occurred in the Bayesian network workshop at UAI 2009. In the workshop, many participants shared their personal experiences that majority BN users did not know how to use BN properly.

support or strength of opinions. Beliefs are represented with probability functions or belief functions (Shafer 1990). The core of TBM is two representation levels for beliefs: the *credal* level where beliefs are entertained, and the *pignistic* level where beliefs are considered and used to make decisions. Both levels could use probability functions for quantified beliefs. However, use of probability functions is only required when a decision or betting is involved (Ramsey 1931b). Therefore, in TBM, probability functions are usually used at the pignistic level for making decisions, whereas belief functions are used to quantify uncertainty at the credal level. An important advantage of using belief functions at the credal level is that normalisation of beliefs is less of a concern. Translation between the two levels is achieved through the use of pignistic transformation functions (Smets 1990). In the following section, we give a formal definition of belief functions and related concepts. We provide some details on the combination rules for belief functions and a generalised Bayesian theorem for conditional belief functions. Finally, we discuss the necessary pignistic transformation function for translating the belief function at credal level to the probability functions used in the pignistic level. TBM with its combination rules and transformation functions will form the basis of the quantitative risk management approach presented in Chapter 7.

4.5.1 Belief Function

Shafer first used belief functions to model uncertain knowledge with evidence theory (1976). From the possible world perspective: the set of all possible worlds Ω is the frame of discernment, which is a finite set of mutually exclusive elements. Let $\mathcal{R}$ be the power set of Ω consists of 2^{Ω} atoms of Ω. A belief function is a function *bel* that maps $\mathcal{R}$ to [0,1] (Smets 1998) such that:

1. $bel(\emptyset) = 0$;
2. $bel(A_1 \cup A_2 \cup ... A_n) \geq \sum_i bel(A_i) - \sum_{i>j} bel(A_i \cap A_j) .. - (-1)^n bel(A_1 \cap A_2 \cap ... A_n)$, for all A_1, A_2,...,$A_n \in \mathcal{R}$.

Another key concept in TBM is the *basic belief assignment*(bba) which is related to the belief functions such that:

$$m(A) = \sum_{B:B\in\mathcal{R},\phi\neq B\subseteq A} (-1)^{|A|-|B|} bel(B) \tag{4.21}$$

and

$$\sum_{A\in\mathcal{R}} m(A) = 1 \tag{4.22}$$

$$bel(A) = \sum_{B:B\in\mathcal{R},\phi\neq B\subseteq A} m(B) \tag{4.23}$$

for all $A \in \mathcal{R}, A \neq \phi$; $|A|$ is the number of atoms of $\mathcal{R}$ in A.

The non-negative value of $m(A)$ for $A \in \mathcal{R}$ is called the *basic belief mass*(bbm) at A. The total amount of specific support for A is provided by the sum of all bbm given to subsets of A. The *focal elements* of a bba m^Ω are the subsets A of Ω such that $m^\Omega(A)$ is positive. A *categorical belief function* on Ω focused on $A^* \subseteq \Omega$ is a belief function with bba m^Ω that satisfies

$$m^\Omega(A) = \begin{cases} 1 & \text{if } A = A^*, \\ 0 & otherwise. \end{cases}$$

A categorical belief function is equivalent to a classical propositional formula; when all bbas are categorical, the TBM is equivalent to propositional logic. Furthermore, we can define a plausibility function $pl : 2^\Omega \to [0,1]$ and commonality functions $q : 2^\Omega \to [0,1]$ from belief functions such that (Smets 1998)

$$\begin{aligned} pl(A) &= bel(\Omega) - bel(\neg A) \text{ and } pl(\emptyset) = 0 \\ q(A) &= \sum_{A\subseteq B} m(B) \text{ and } q(\emptyset) = 1, \end{aligned}$$

where $\neg A$ is the complement of A with respect to Ω. TBM also adopts the principle of minimal commitment such that subsets of Ω do not receive more support (bbm) than is justified. In other words, if there are many belief functions that satisfy constraints in a domain, the belief function that gives the smallest degree of belief to every subsets of Ω is adopted.

Based on these basic concepts, we introduce several TBM concepts that will play crucial roles in the new framework when dealing with (subjective) quantitative domain information.

4.5.2 Simple Support Function

A simplified version of a belief function is called a simple support function. It consists of only two non-null basic belief masses. One gives support to Ω; the other gives support to a specific subset of Ω, A. We let A^x to be the simple support function with belief mass of $m(A) = 1 - x$ and $m(\Omega) = x$. That is, proposition A has belief support $1-x$; and x support for all other possible worlds that do not include A. Note that, categorical belief functions are a special class of simple support functions.

4.5.3 Vacuous Belief Function

In TBM, we can represent total ignorance with a vacuous belief function. A vacuous belief function, usually denoted as T, is a belief function that $m(\Omega) = 1$, hence, $bel(A) = 0 \; \forall A \in \mathcal{R}, A \neq \Omega$ and $bel(\Omega) = 1$. This means, belief of all possible worlds is true and anything is possible. However, there is no specific belief in any subset of worlds, i.e. the vacuous belief function is a categorical belief function focused on Ω. The vacuous belief function is the least committed belief function among all belief functions on Ω.

4.5.4 Latent Belief Structure

Another key feature of TBM is its ability to represent the conflicting belief "there are some reasons to believe A" and "there are some reasons *not* to believe A" simultaneously. TBM uses a latent belief structure to represent such a state of belief. A latent belief consists of a pair of belief functions (C, D) where C and D represent a confidence and a diffidence component respectively. The classical belief state can be represented by (C, T) where T is the vacuous belief function. It has only the confidence component. On the other hand, (T, D) has only the diffidence component which means we only have reasons *not* to believe A. These pure belief states can also be represented using simple support functions such as (A^x, T) and (T, A^y), where $x, y \in [0, 1]$.

4.5.5 The Λ Operator and Apparent Belief

Finally, we also need to ensure that, as the overall effect, the confidence component and diffidence component can effectively counter-balance each other. When they are numerically equal, they cancel each other out completely and we end up with an apparent vacuous belief function T. Consequently, a Λ operator is introduced to transform a latent belief structure into an apparent belief function[22] such that $\Lambda(A^x, A^x) = T$. The apparent belief function is effectively a normal belief function. In Section 7.1.1.5, we propose a modification of the Λ operator: $\dot{\Lambda}$ which computes the relative strength of a causal connection.

4.5.6 Rules of Combination

In most environments, it is frequently required that information from different sources are to be combined together. A number of combination rules have been proposed to provide different strategies for combining disparate evidences. The most well-known combination rule is so-called Dempster's rule of combination defined as follows:

Definition 3. *Given two bbas m_1^Ω and m_2^Ω, the result from the Dempster's rule of*

[22]This transformation operator is limited to cover transformation into valid belief function space.

combination of m_1^Ω *and* m_2^Ω *is*

$$m_{12}(A) = \sum_{X \cap Y = A \neq \emptyset} \frac{m_1^\Omega(X) m_2^\Omega(Y)}{1 - K}, \forall A \subseteq \Omega, \tag{4.24}$$

where

$$K = \sum_{X \cap Y = \emptyset} m_1^\Omega(X) m_2^\Omega(Y)$$

and $m_{12}(\emptyset) = 0$.

Note that $1 - K$ is a normalisation constant that ensures combination results remain less than or equal to 1. In particular, K is a measure of the amount of conflict between two bbas (or two pieces of evidence). This means that any conflict between two pieces of evidence is discounted from the combination result. Consequently, the Dempster's rule of combination produces counterintuitive results when it is used to combine two pieces of highly conflicting information (Zadeh 1979). This issue reflects the underlying closed world assumption adopted by the combination rule as $m(\emptyset) = 0$, i.e. "nothing" is unknown. In TBM, the Dempster's rule of combination is replaced with an alternative combination rule, the so-called TBM conjunctive combination rule (Smets 2007), to accommodate possible contradictory information:

Definition 4. *Given two bbas* m_1^Ω *and* m_2^Ω*, the result from the TBM conjunctive combination rule of* m_1^Ω *and* m_2^Ω *is*

$$m_{12}(A) = \sum_{X \cap Y = A} m_1^\Omega(X) m_2^\Omega(Y), \forall A \subseteq \Omega. \tag{4.25}$$

TBM conjunctive combination rule is an unnormalised version of the Dempster's rule of combination. Instead of normalising the combination result, TBM conjunctive combination rule adopts the Open World assumption and allows $m(\emptyset) > 0$. In addition to conflicting evidences, $m(\emptyset)$ could also be caused by the frame of discernment is not exhaustive or there is a positive belief that the two bbas m_1^Ω and m_2^Ω do not correspond to the same object. A positive mass on the empty set indicates that additional analysis and knowledge is required for the domain and some evidences should be either discounted or rejected entirely. In addition to the TBM conjunctive combination rule, a disjunctive combination rule was also introduced in TBM (Smets 1993):

Definition 5. *Given two bbas m_1^Ω and m_2^Ω, the result from the TBM disjunctive combination rule of m_1^Ω and m_2^Ω is*

$$m_{12}(A) = \sum_{X \cup Y = A} m_1^\Omega(X) m_2^\Omega(Y), \forall A \subseteq \Omega. \tag{4.26}$$

This rule is used to combine separate pieces of evidence when it is known that either piece holds true.

4.5.7 Conditional Belief Function and Rules of Conditioning

Analogous to conditional probability, a conditional belief function $bel(B \mid A)$ can be defined to represent the degree of belief in B given a context in which A holds (Smets 1993):

$$\begin{aligned} m(B \mid A) &= \sum_{X \subseteq \neg A} m(B \cup X), \text{ if } B \subseteq A \subseteq \Omega \\ &= 0, \text{ otherwise} \\ bel(B \mid A) &= bel(B \cup \neg A) - bel(\neg A), \ \forall B \subseteq \Omega \\ pl(B \mid A) &= pl(A \cap B), \ \forall B \subseteq \Omega. \end{aligned} \tag{4.27}$$

Equation 4.27 represents an unnormalised conditioning process in which supporting mass $m(B)$ given to B is *transferred* to $A \cap B$ by conditioning on A. A normalised version of this rule is called Dempster's rule of conditioning. Under TBM, this redistribution of belief mass is called *specialisation*. Belief mass $m(B)$ is transformed with a specialisation matrix S_A on A, i.e. $m_A(B) = S_A \cdot m_0(B)$ where m_0 is the original bbm. In fact, it has been shown that rules of conditioning are the least committed specialisation (Klawonn & Smets 1992). Combination rules discussed in the previous section are also cases of specialisation (Smets 1998).

4.5.8 Generalised Bayesian Theorem

In TBM, the Bayesian theorem of conditioning is generalised for (conditional) belief functions. Given three frames of discernment X, Y and Θ and our knowledge of X, Y and Θ is represented by $bel_x(.\mathrel{\text{¦}}\theta_i)$ and $bel_y(.\mathrel{\text{¦}}\theta_i)$ $\forall\theta_i \in \Theta$. X and Y are conditionally independent given $\theta_i, \forall\theta_i \in \Theta$ and

$$bel_\Theta(.\mathrel{\text{¦}}x, y) = bel_\Theta(.\mathrel{\text{¦}}x) \mathbin{\bigcirc\!\!\!\!\!\cap} bel_\Theta(.\mathrel{\text{¦}}y),$$

where $\mathbin{\bigcirc\!\!\!\!\!\cap}$ denotes the TBM conjunctive combination operation. We can deduce the following with $bel_X(X|\theta_i) = 1$ and $bel_Y(Y|\theta_i) = 1, \forall x \subseteq X, \forall\theta \subseteq \Theta$ (Smets 1993):

$$\begin{aligned}
bel_\Theta(\theta\mathrel{\text{¦}}x) &= \prod_{\theta_i\in\neg\theta} bel_X(\neg x|\theta_i) - \prod_{\theta_i\in\Theta} bel_X(\neg x|\theta_i) && (4.28)\\
bel_\Theta(\theta|x) &= K \cdot bel_\Theta(\theta\mathrel{\text{¦}}x) \\
pl_\Theta(\theta\mathrel{\text{¦}}x) &= 1 - \prod_{\theta_i\in\theta}(1 - pl_X(x|\theta_i)) && (4.29)\\
pl_\Theta(\theta|x) &= K \cdot pl_\Theta(\theta\mathrel{\text{¦}}x) \\
q_\Theta(\theta\mathrel{\text{¦}}x) &= \prod_{\theta_i\in\Theta} pl_X(x|\theta_i) && (4.30)\\
q_\Theta(\theta|x) &= K \cdot q_\Theta(\theta\mathrel{\text{¦}}x)
\end{aligned}$$

Note that $|$ and ¦ represent normalised conditional and unnormalised conditional respectively. They are connected through a normalisation constant K that

$$K^{-1} = 1 - \prod_{\theta_i\in\Theta} bel_X(\neg x|\theta_i) = 1 - \prod_{\theta_i\in\Theta}(1 - pl_X(x|\theta_i)).$$

The equations above form the so-called *normalised* Generalised Bayesian Theorem(GBT). A more general form of GBT removes the normalisation constant K and becomes:

$$\begin{aligned}
bel_\Theta(\theta\mathrel{\text{¦}}x) &= \prod_{\theta_i\in\neg\theta} bel_X(\neg x\mathrel{\text{¦}}\theta_i) - \prod_{\theta_i\in\Theta} bel_X(\emptyset\mathrel{\text{¦}}\theta_i) && (4.31)\\
pl_\Theta(\theta\mathrel{\text{¦}}x) &= 1 - \prod_{\theta_i\in\theta}(1 - pl_X(x\mathrel{\text{¦}}\theta_i)) && (4.32)\\
q_\Theta(\theta\mathrel{\text{¦}}x) &= \prod_{\theta_i\in\Theta} pl_X(x\mathrel{\text{¦}}\theta_i) && (4.33)
\end{aligned}$$

Furthermore, if every $bel_X(x \mid \theta_i)$ is replaced with a probability function $P(.|\theta_i)$ and a priori belief is provided with a probability function $P_0(\theta)$, the normalised GBT is reduced to the standard Bayesian theorem.

4.5.9 Evidential Network

A direct application of conditional belief function and GBT is that we can construct a graphical model similar to a Bayesian network for modelling uncertainty in a domain. This is the so-called evidential network in which nodes represent domain variables and edges between nodes represent the correlation between the variables. Instead of using conditional probabilities, an evidential network uses conditional belief functions; and conditional probability tables are replaced by tables that contain bbms from the conditional belief functions. Table 4.7 shows an example of a conditional belief function (extracted from Xu et al. (1994)). Suppose A and B are two variables with frames $\Theta_A = \{a, \neg a\}$ and $\Theta_B = \{b, \neg b\}$ respectively. $\Theta = \{ab, a\neg b, \neg ab, \neg a\neg b\}$ is denoted as $\{1, 2, 3, 4\}$ so that subset $\{ab, a\neg b\}$ can be denoted as 12.

	a	$\neg a$	Θ_A
b	$m(14) + m(134) =$ $0.1 + 0.1 = 0.2$	$m(23) + m(123) =$ $0.1 + 0.1 = 0.2$	0
$\neg b$	$m(23) + m(234) =$ $0.1 + 0.1 = 0.2$	$m(14) + m(124) =$ $0.1 + 0.1 = 0.2$	0
Θ_B	$m(123) + m(124)+$ $m(1234) = 0.6$	$m(134) + m(234)+$ $m(1234) = 0.6$	$m(14) + m(23) + m(123)+$ $m(124) + m(134)+$ $m(234) + m(1234) = 1$

Table 4.7: An example of a conditional belief function represented in a table of bbms.

Evidential networks can be divided two subcategories, namely networks with undirected edges and networks with directed edges. Reasoning in undirected networks uses a valuation-based system (Shenoy & Shafer 1990). Inferences are drawn through joint valuation by combining all the valuations on the joint product space and/or marginal valuation on a subdomain by variable elimination. This generalised inference approach appears as the jointree algorithm in Bayesian network. For directed evidential networks, a more computationally efficient algorithm was developed based on GBT and the TBM disjunctive combination rule (Yaghlane & Mellouli 2008). In practice, modelling reasoning is frequently accompany by decision making. It is more useful to convert an evidential network into a probabilistic model and utilise inference algorithms developed for Bayesian network, since those algorithms are often more mature and computationally efficient. In the next section, we discuss the transformation from belief functions to probabilities.

4.5.10 Pignistic Transformation

When we are required to make decisions that take risk into consideration, beliefs at the credal level must be transformed into a probability functions so that they can be used to select the best decision. Such a transformation is called a *pignistic transformation*. The final probability function, the so-called pignistic probability function, is a function of the original belief function (Smets 1990), denoted as $BetP$:

$$BetP(w) = \sum_{A:w \subseteq A \in \mathcal{R}} \frac{m(A)}{|A|(1 - m(\emptyset))} \text{ for any atom } w \text{ of } \mathcal{R}, \tag{4.34}$$

where $|A|$ is the number of atoms of $\mathcal{R}$ in A and

$$BetP(A) = \sum_{w:w \in At(\mathcal{R}), w \subseteq A \in \mathcal{R}} BetP(w) \ \forall A \in R.$$

The pignistic transformation depends on the structure of the frame from which the (betting) decision is made. The *betting frame* A is a set of atoms in $\mathcal{R}$ on which stakes are assigned. A stake given to each atom is independent of the stakes given to the other atoms. A betting frame $\mathcal{R}$ is defined and an appropriate stake is assigned to each atom of $\mathcal{R}$ in order to define a pignistic transformation for a belief model at the credal level. Different betting frames will lead to different pignistic transformation functions and betting frames should be defined according to some utility function.

Note that, as a special case, if probability is used as the belief function on the credal level, the pignistic transformation is an identity transformation (Smets 1998). For details of pignistic transformation, justifications for having a two level model and implications of the pignistic transformation, we point readers to the works of Smets (1990, 1988, 2005).

4.5.11 *TBM versus Probabilistic Model

Transferable Belief Models are closely related to probabilistic models, i.e. Bayesian network. In particular, directed evidential networks closely resemble BN graphically. However, they are significantly different in several key areas:

- BN provides one level modelling solutions that solely rely on (conditional) probabilities, whereas TBM is a two-level solution that uses belief function for modelling and probability for decision making. In other words, TBM separates the modelling and decision making process.

- Because of the use of probability, the fundamental assumption that BN uses is the Closed World assumption; it cannot model ignorance appropriately. TBM, on the other hand, is based on the open world assumption and can model both partial and complete ignorance. Being able to model ignorance leads to a more sensible solution under some circumstances (Smets 1988). In fact, we regard this as the key advantage that TBM has over the BN, since we do not assume an agent has a complete knowledge of the domain (Framework Requirement 2 in Section 3.2).

- Models in BN can be either constructed manually with inputs from human experts or constructed automatically from domain data. Automated model construction relies on abundant quantitative domain data whereas the manual construction relies on subjective probabilities generated from experts, i.e. it is assumed that human expert can generate probability reliably. TBM do not have algorithms for automated model construction directly from data. Instead, TBM focuses on utilising subjective degrees of belief, i.e. belief function, normally acquired from human experts. The use of belief functions puts less demands on human experts and can handle unknowns and conflicting inputs from different sources through appropriate use of combination rules. More importantly, TBM offers a systematic way to transform belief functions to subjective probabilities. Bayesian networks lack a similar formal strategy to capture knowledge from human experts.

Compared with TBM, Bayesian probabilistic models have received significantly more research attentions in the past three decades. Accordingly, its modelling and reasoning algorithms are mature and efficient; its related software tools are well developed. However, BN should be not applied without careful consideration of the respective domain causal relationships are modelled. TBM, as a generalised belief modelling framework, is best for modelling in domains that have characteristics unsuited for BN. Specifically, domains that are not well understood, and/or do not have sufficient quantitative information and the main sources of domain knowledge are human experts with different opinions. We conclude this section with a view that TBM and probabilistic models complements each other. In the risk management framework, we shall adopt appropriate modelling strategies based on the characteristics of problem domains.

4.6 Possibility Theory

The previously discussed approaches for managing incomplete and uncertain knowledge are all specialised in dealing with either qualitative or quantitative information or beliefs. None of them is designed to take account of both qualitative and quantitative knowledge. Possibility theory, initially introduced by Zadeh (1978) and further developed by Dubois and Prade (1988), straddles between the qualitative/quantitative divide and can be utilised to model problems in complex domains. In fact, the overall possibility theory consists of two separate theories, one is qualitative and the other is quantitative. Both theories are built upon a common notion of *possibility* and share the same kind of set-functions. However, their operations, namely, conditioning and combination mechanisms, are distinctively different. The qualitative possibility theory is closely related to the non-monotonic reasoning, whereas the quantitative possibility theory can be regarded as a special case of belief function and probability in general. Therefore, we still need to select appropriate possibility theory for the purpose of modelling uncertainty in risk, depending on the problem domain. In this section, we review the key concepts of the possibility theory and discuss their implications and possible usage in the new risk management framework.

4.6.1 Possibility Distributions

A key concept in possibility theory is a possibility distribution, commonly denoted as π, which is defined as a mapping from a set of possible situations U to a bounded total ordered scale $(L, <)$. The referential set U is a finite set of mutually exclusive situations which is the same as the frame of discernment in belief functions[23]. The total ordered scale $(L, <)$ could be on any scale interval that has maximum and minimum operations, and an order-reversing map function[24]. Typically, a unit interval $[0, 1]$ is used as the ordered scale. A possibility distribution is generally associated with a random variable x ranging within U, denoted as π_x, that represents the knowledge about the state of affairs distinguishing what is plausible from what is less plausible. Specifically, $\pi_x(u) = 0$ means $x = u$ is impossible, whereas $\pi_x(u) = 1$ means $x = u$ is totally possible. The degree of possibility for $x = u$ is reflected in the values of $\pi_x(u)$ between 0 and 1. A possibility distribution is *normalised*, if $\exists u, \pi_x(u) = 1$. This means, the referential set U represents a complete range for x such that at least one element of U is fully possible.

Using a possibility distribution, we can represent a state of complete knowledge

[23] In this thesis, we only consider the finite cases.

[24] The order-reversing operation ensures the ordering among the possibility degree is meaningful.

by $E = u_0$ for some u such that $\pi_x(u_0) = 1$ and $\pi_x(u) = 0$ for $u \neq u_0$, i.e. only $x = u_0$ is possible. On the other end of the scale, a state of complete ignorance can be represented by $E = U$ and $\pi_x(u) = 1$, $\forall u$, i.e. every u is totally possible. Furthermore, given two possibility distributions, π_x and π'_x, for the same variable x such that $\pi_x < \pi'_x$, then π_x is more informed about values of x and hence puts more restrictions on the plausible values of x than π'_x. π_x is said to be more *specific* than π'_x, and π_x makes π'_x is redundant. Based on this notion, a principle of minimal specificity was introduced (Dubois & Prade 1987). It states that for a given set of constraints, the best representation of the state of knowledge is the the least specific possibility distribution that is compatible with the constraints. This distribution assigns the maximum possibility degrees to the elementary states of affairs in agreement with the constraints. This principle of minimal specificity plays a crucial role in the combination and conditioning of beliefs in the possibility theory, in the same spirit as the principle of minimal changes in Belief Revision.

4.6.2 Possibility and Necessity Measures

In possibility theory, the uncertainty or incompleteness of information is captured by means of set-functions and assigning degrees of confidences to the occurrence of the associated events. The extent of the information described by π_x that is consistent with the existing knowledge is given by a possibility measure defined as:

$$\Pi(A) = max_{u \in A} \pi_x(u) \text{ (for finite cases).}$$

The value of $\Pi(A)$ equals to the greatest possibility degree one element of A can obtain according to π_x. This possibility measure conforms to the maxitivity axiom (similar to probability measure conforms to the axioms of probability)

$$\Pi(A \cup B) = max(\Pi(A), \Pi(B)), \tag{4.35}$$

along with $\Pi(\emptyset) = 0$ and $\Pi(U) = 1$. A counterpart to a possibility measure is a necessity measure which represents the degree of certainty of A. The necessity measure is defined with possibility measure of 'not A' so that

$$N(A) = 1 - \Pi(\neg A),$$

since $A \cup \neg A = U$ and if the occurrence of 'not A' is not possible then A is certain. This duality relationship between possibility and necessity is a graded version of possibility and necessity found in modal logics, i.e. $\Box A = \neg\Diamond\neg A$. It represents a weak relationship between $\Pi(A)$ and $\Pi(\neg A)$ that is markedly different from the strong correlation between $Pr(A)$ and $Pr(\neg A)$ in probability theory. That is, while $N(A) = 1$ means that A is certainly true; $N(A) = 0$ means A is not certain but could still be possible, i.e. $\Pi(A)$ can be greater than zero. On the other hand, $\Pi(A) = 1$ expresses A is entirely possible but says nothing about $\Pi(\neg A)$. The strong $Pr(A)+Pr(\neg A) = 1$ relation found in probability theory reflect the 'closed world' and additive nature of the theory. Possibility theory does not share the same characteristics. It is also easy to show that $N(A) > 0$ implies $\Pi(A) = 1$ and $\Pi(A) \geq N(A)$. Necessity measures also satisfy the dual of the maxitivity axiom, i.e.

$$N(A \cap B) = min(N(A), N(B)).$$

4.6.2.1 Comparative Possibility Relation

In qualitative possibility theory, the uncertainty relations among events is represented using the so-called comparative possibility relation $\geq_\Pi$ on a set of events A, B etc. $A \geq_\Pi B$ means A is more plausible than B, i.e. A is more consistent with the available knowledge than B. Comparative possibility relation is functionally equivalent to the likelihood order we use to encode uncertainty relation among a set of possible worlds described in section 2.3.2. Technically, comparative possibility relation can be defined by the following axioms:

1. $A \geq_\Pi B$ or $B \geq_\Pi A$. (Completeness)

2. $A \geq_\Pi B$ and $B \geq_\Pi C$, then $A \geq_\Pi C$. (Transitivity)

3. $U >_\Pi \emptyset$, where $>_\Pi$ is the strict part of $\geq_\Pi$ ordering.

4. $U \geq_\Pi A \geq_\Pi \emptyset$.

5. $A \geq_\Pi B \rightarrow C \cup A \geq_\Pi C \cup B$.

Note that axiom 5 means if A is at least consistent as B with the available knowledge, it implies that A or C is at least consistent in relation to B or C. It has been shown that the only set-functions that satisfy the comparable possibility relation axioms are possibility measures (Dubois 1986).

A counterpart to the comparable possibility relation is the qualitative necessity relation defined by duality as $A \geq_N B \equiv \neg B \geq_\Pi \neg A$. Qualitative necessity relations are closely related to the epistemic entrenchment relation in AGM based Belief Revision (Dubois & Prade 1991). Information with higher necessity is less easily given up than the knowledge with less necessity. In parallel, qualitative possibility relation is closely related to revision functions. Suppose $K' = K * A$, both A and B are in K' if and only if $A \wedge B >_\Pi A \wedge \neg B$. In other words, B is in K' if B is more consistent with input A than $\neg B$. Belief change operations in AGM based BR correspond to possibility theory operated on integers (Dubois, Moral & Prade 1998). An Ordinal Epistemic Function is essentially a necessity measure and adjustment process can be cast into possibility framework by representing an epistemic state as a finite set of weighted formulas with a lower bound of necessity degree (Benferhat, Dubois, Prade & Williams 2002). Furthermore, under the qualitative possibility theory, a System of Spheres (Grove 1988) that represents a nested set of possible worlds can be reinterpreted as the level cuts $\{u|\mu_W(u) \geq \alpha\} = W_\alpha$, $\alpha \in L - \{0\}$, of a fuzzy set W of possible worlds based on the available knowledge, where μ_W is the member function of W. We can associate a possibility distribution π with a total preorder $\geq_\pi$ such that $u \geq_\pi u' \equiv \pi(u) \geq \pi(u')$. This induces a well-order partition of U, namely, $\{E_1, ..., E_{n+1}\}$ with the following property,

$$\forall u \in E_i, \forall u' \in E_j, \pi(u) > \pi(u') \text{ iff } i < j \text{ for } i < n+1, j > 1.$$

E_{n+1} is a subset of impossible states with $\pi(u) = 0$.

The strong connection between the qualitative possibility theory and Belief Revision suggests that Belief Revision, particularly when it is cast into the possibility framework, should play an important role in managing uncertainty caused by incomplete domain knowledge.

4.6.2.2 Conditioning and Combination in Qualitative Possibility

A conditional possibility measure is defined with a set function $\Pi(\cdot|A)$ in which

$$\forall B, B \cap A \neq \emptyset, \Pi(A \cap B) = min(\Pi(B|A), \Pi(A)).$$

With the principle of minimal specificity of the possibility theory, we have the least specific solution, when $B \cap A \neq \emptyset$, as

$$\begin{aligned} \Pi(B|A) &= 1, \text{ if } \Pi(A \cap B) = \Pi(A) \\ &= \Pi(A \cap B), \text{ otherwise.} \end{aligned} \tag{4.36}$$

Conditional possibility measures closely resemble conditional probability without the renormalisation process. Instead, conditional possibility measures simply move the most plausible elements of A to nominal 1. The corresponding possibility distribution is defined as

$$\begin{aligned} \pi(u|A) &= 1, \text{ if } \pi(u) = \Pi(A), u \in A \\ &= \pi(u), \text{ if } \pi(u) < \Pi(A), u \in A \\ &= 0, \text{ if } u \notin A. \end{aligned} \tag{4.37}$$

By duality, conditional necessity measure is defined as $N(B|A) = 1 - \Pi(\neg B|A)$. A direct result from this definition is that $\Pi(A) > 0, N(B|A) > 0 \leftrightarrow \Pi(A \cap B) > \Pi(A \cap \neg B)$. This means, B can only be accepted according to A if and only if B is more plausible than $\neg B$ when A is true. This possibility conditionalisation is extended to fuzzy sets (A, α) as uncertain inputs where α is a degree of necessity, for Belief Revision (Dubois & Prade 1997). In particular, it has been shown that adjustment (Williams 1994b) can be reformulated in the form of possibilistic conditioning

(Benferhat, Dubois, Prade & Williams 2010). The combination of qualitative possibility distributions for a variable requires a possibility scale common to all possibility distributions. Combining comparative possibility distribution is not possible because of the impossibility of combining comparative relation (due to Arrow's impossibility theorem (Arrow 1950)). There are two modes of combination of possibility distribution on a common scale L, namely conjunctive and disjunctive modes. They are equivalent to an intersection and union of fuzzy sets respectively. Specifically, for two possibility distributions π and π', we have

$$\begin{aligned} \pi^{\cap} &= min(\pi, \pi')(conjunctive) \\ \pi^{\cup} &= max(\pi, \pi')(disjunctive) \end{aligned}$$

The conjunctive mode corresponds to the pooling together of fully reliable information from two sources, while removing elements not considered to be possible in one source. The conjunctive combination rule leaves the resultant π unnormalised if the sources are in conflict. In this case, a qualitative normalisation process is required to move the possibility values that have greatest possibility degree 1. The disjunctive combination rule simply combines all available information. This potentially leaves faulty information hidden within the result. The disjunctive combination rule is the only one that is consistent with the axiom of possibility theory. Conjunctive and disjunctive combination rules closely resemble the conjunctive and disjunctive combination rules for belief functions.

4.6.2.3 Quantitative Possibility Measure

In quantitative possibility theory, a numerical possibility measure (and the corresponding possibility distribution) is the same as the possibility measure defined in qualitative theory if U is finite. When U becomes infinite, the equivalence breaks down since the numerical possibility measure no longer conforms to the finite maxitivity axiom Equation 4.35. An infinite maxitivity axiom must be adopted instead:

$$\Pi(\bigcup_{i \in I} A_i) = sup_{i \in I} \Pi(A_i) \tag{4.38}$$

Therefore, a possibility distribution π exists on U such that $\Pi(A) = sup_{u \in A}\pi(u)$, and $N(A) = 1 - \Pi(\neg A)$. The meaning of numbers in numerical possibility measures differ significantly to the qualitative case. In the qualitative theory, the numbers have no meanings up to a monotone transformation; whereas the numbers in the numerical possibility measure have meanings in absolute numerical values. The interpretation of the numbers in quantitative possibility measures can take several different forms. In particular, a quantitative necessity measure is interpreted as a belief function that gives minimum support in TBM (Smets & Kennes 1994); a numerical possibility measure can be viewed as a upper bound of an unknown probability in generalised probability theories. However, we should note that a possibility measure is significantly different from a probability measure in various aspects so it cannot be subsumed under probability. Most notably, possibility measures are non-additive. The necessity measure and the possibility measure are weakly related through $N(A) = 1 - \Pi(\neg A)$; whereas $1 - Pr(\neg A)$ is $Pr(A)$ in probability theory. Possibility theory makes a clear distinction between what is believed (necessity measure) and what is possible (possibility measure); whereas probability theory does not. These discrepancies exist between the quantitative possibility theory and probability theory are due to the closed world assumption on which probability theory is based. Quantitative possibility theory is more closely connected with belief functions and TBM in terms of handling incomplete domain information.

4.6.2.4 Conditioning and Combination in Quantitative Possibility

The key differences between qualitative and quantitative possibility theories are in their respective conditioning and combination operations. The qualitative conditioning is based on the *min* operation which is problematic in the infinite case due to lack of continuity, as Equation 4.36 implies that $a \to b = 1$ if $a < b$ and $a \to b = b$ if $a > b$. Instead, the quantitative conditioning is a product-based conditioning that closely resembles the conditioning in probability theory:

$$\forall B, B \cap A \neq \emptyset, \Pi(B|A) = \frac{\Pi(A \cap B)}{\Pi(A)}, \tag{4.39}$$

if $\Pi(A) \neq 0$. The corresponding conditional possibility distribution is

$$\begin{aligned} \pi(u|A) &= \frac{\pi(u)}{\Pi(A)}, \forall u \in A \\ &= 0, \text{ otherwise.} \end{aligned} \tag{4.40}$$

The dual conditional necessity measure is $N(B|A) = 1 - \Pi(\neg B|A)$. This corresponds to Dempster's rule of conditioning specialised for possibility measures (Shafer 1976).

In terms of combination operations for numerical possibility measures, the basics of the conjunctive and disjunctive combination rules used in qualitative case still apply with an additional normalisation. If π and π' are two possibility distributions to be combined to form $\pi^{\cap}$, the the normalised conjunctive rule is

$$\pi^{\cap} = \frac{min(\pi, \pi')}{sup_{u \in U} min(\pi(u), \pi'(u))}.$$

This rule can be considered as an extension of Dempster's rule of conditioning which can be recovered with $\pi' = u_A$. The dual disjunctive combination rule can be expressed in the form of $1 - (1 - a) * (1 - b)$ where $*$ is the conjunctive combination operation. It can be used without normalisation for merging information from two unreliable sources.

4.6.3 *Connecting Qualitative and Quantitative Uncertainty

From the previous sections, the possibility framework provides two tightly connected techniques for both qualitative uncertainty modelling and quantitative uncertainty modelling based on a common concept of possibility measure (and corresponding distribution). It demonstrates that a set of possible worlds in a comparative possibility relation, or a functionally equivalent likelihood preorder is a viable solution for encoding uncertainty qualitatively. Belief Revision is functionally important for managing uncertainty in a knowledge base with incomplete domain information. On the quantitative side, the numerical possibility measure can be viewed as a special case of belief function which is subsumed under the transferable belief model. Possibility theory offers the crucial theoretical insight into how we may bridge qualitative and quantitative uncertainty, and therefore risk, modelling. Now I turn my attention to how we may capture and model causal relations found in a domain.

4.7 Capturing Causal Connections

One of the key requirements for the risk management framework is to capture the causal connections among the domain factors/variables associated with risks in a target domain. Therefore, we need effective method(s) to capture the causal structure within the domain. Causality and causal relations have been long studied and debated by philosophers and AI theoreticians (Davidson 1967, Spohn 1983, Pearl 2000, Bonnefon, Neves, Dubois & Prade 2008). I do not intend to add to this debate, but to review two theoretical methods that may be used to distill causally based connections between domain variables from the knowledge of human experts and quantitative data respectively.

4.7.1 Ramsey Test

The Ramsey Test is a test for conditionals derived from the work of F.P. Ramsey (1931a). It gives a criterion for the rational acceptance of epistemic conditionals "if A then B":

> A conditional "if A then B", denoted as $A > B$, is rationally accepted in a given state of belief K if and only if B is accepted in K after the K is revised with A as a new piece of information, i.e. $K * A$.

The epistemic conditional of the hypothetical modification of K with A must *precede* the inclusion of B in K captures the belief that A is an *epistemic reason* for B. It is only accepted with the hypothetical modifications of beliefs about the world, i.e. $B \in K * A$ (assume K is a belief set, A and B are sentences). In other words, the Ramsey Test can be used to capture the subjective *belief* that A is a cause of B. It fits naturally with our goal of capturing the subjective knowledge (or beliefs) of the causally based inference relationships from the domain experts for the purpose of modelling risk. Furthermore, the Ramsey Test is such an intuitive criterion that it can be translated into direct queries for experts to obtain domain knowledge (see Section 7.1.1.1 for example).

4.7.1.1 Compatibility with Belief Revision

The formulation of the Ramsey Test relies on the revision of a given belief set with the antecedent of the conditional. However, the standard Ramsey Test has be shown to violate certain AGM Belief Revision postulates (Gärdenfors 1988). Under the AGM paradigm, the Ramsey Test takes a following form:

$$\text{For every belief set } K,\ A > B \in K \text{ if only if } B \in K * A.$$

This implies that Belief Revision has to be a monotonic function. That is, if a belief set G is included in another belief set K, then $G * A$ is also included in $K * A$. This monotonicity is in direct conflict with the combined effect of the expansion postulates $(K+1)$, $(K+2)$ and $(K+3)$ (Section 4.3.1.2) and leads to the so-called impossibility theorem (Gärdenfors 1988). A number of approaches have been proposed to resolve this inconsistency between the Ramsey test and Belief Revision through either weakening the Ramsey test (Lindström & Rabinowicz 1992) or weakening the AGM postulates (Lindström & Rabinowicz 1998). A third approach is to investigate the validity of the underlying assumption used in the impossibility theorem. That is, the assumption that an epistemic conditional such as $A > B$ should be a member of the belief set K. According to Levi (1988), the evaluation of the consequent of an epistemic conditional $A > B$, i.e. B, is not based on the current belief set K; but on the *potential revision* of K with the antecedent A. Therefore, the conditional should not belong to the belief set K but belong to an associated corpus $RL(K)$ of epistemic appraisals. Levi defines $RL(K)$ to be the smallest set of sentences that satisfies the following conditions:

1. $K \subseteq RL(K)$;
2. if $B \in K * A$, then $A > B \in RL(K)$;
3. if $B \notin K * A$, then $\neg(A > B) \in RL(K)$;
4. $RL(K)$ is closed under truth-functional consequence.

$RL(K)$ contains all the beliefs of an agent together with the epistemic conditionals that represent the evaluation of the agent's beliefs. Based on $RL(K)$, a pair of modified Ramsey Tests were developed.

(Levi's Ramsey Test) For any sentence A, B and for any belief set K,
$A > B$ is accepted in $RL(K)$ if and only if $B \in K * A$.

(Levi's Negative Ramsey Test) For any sentence A, B and for any belief set K,
$\neg(A > B)$ is accepted in $RL(K)$ if and only if $B \notin K * A$.

The standard monotonicity is not derivable from the Levi's Ramsey Test. Even when we introduce a weaken version of monotonicity that is permitted by the Levi's Ramsey Test[25], the situation which gives rise to the impossibility theorem is (itself) impossible. The Levi's version of Ramsey Test solves the incompatibility problem between the standard Ramsey Test and AGM Belief Revision postulates by providing a non-propositional interpretation to epistemic conditionals and separating them from the classic belief sets. I adopt the same approach in capturing experts' causal domain knowledge in our knowledge engineering risk modelling process (see Chapter 6). That is, I treat the epistemic conditionals that represent the experts' beliefs of the domain causal inference relations separately from the factual domain knowledge in the risk modelling process. In fact, the risk model in the new framework is largely built upon epistemic conditionals; and maintained through the manipulation of these conditionals. I will provide detailed discussions in the following chapters.

[25]That is, for all consistent belief sets G,K and sentence A, if $RL(G) \subseteq RL(K)$ then $G*A \subseteq K*A$.

4.7.2 Inductive Causation

Instead of solely relying on subjective causal knowledge from human experts, it is possible to extract causal structures from the available statistical data through a so-called inductive causation (Pearl 2000) process under the probabilistic/Bayesian network modelling framework. A causal structure is a DAG in which each link represents a direct functional relationship between two domain variables. How each variable is causally inferred by its parents is captured in a form of linear structural equation (Pearl 2000, p.27) such that changes in the parent(s) has deterministic effect on the variable.

Such a causal structure can be inferred from a *minimal* and *stable* probabilistic model and reconstruct the structure of the corresponding DAG. A minimal model is the model with the simplest structure consistent with the statistical domain data. Stable means that any changes in the probability distribution does not introduce extraneous conditional independences in the network. It turns out that there is an unique minimal causal structure for every distribution[26] (Pearl 2000). A partial directed DAG, or a pattern, can be generated from a stable probability distribution using a so-called IC algorithm (Algorithm 14) based on the concept of *intervention*. That is, a causal relationship between two variables can be inferred through a third variable that acts as a virtual external control to influence or manipulate the behaviour of the two variables. The IC algorithm provides a systematic way to find the variables that qualify as virtual controls. Note that, Spirtes et al. (1991) provides a systematic way of searching for the set S_{ab} in the second step of Algorithm 14 and an improved IC algorithm that handles latent variables is listed in Pearl (2000, p.53). The partial directed graph generated from the application of IC algorithm reveals the underlying causal structure of an existing probabilistic model. Specifically, edges that are unidirectional are genuine causation; bidirectional edges are spurious associations; whereas undirected edges are relationships with their causation relations

[26]Up to d-separation equivalence and assuming there is no hidden variables.

undetermined. Therefore, we can utilise the IC algorithm to extract causal structures from probabilistic models or validate an existing Bayesian network is indeed a causal model by comparing the causal structure generated from IC algorithm with the existing structure.

Finally, I would like to reiterate the key difference between the Ramsey Test and IC algorithm: causal knowledge distilled from experts using the Ramsey Test represents the subjective belief of a causal relationship between two variables; whereas the IC algorithm extracts factual causal knowledge from probabilistic models that derived from statistical data. Both methods shall be adopted in the new risk management framework only in the situations that are compatible with their inherent natures.

4.8 *Summary

In this chapter, I have reviewed a number of important theoretical techniques in modelling knowledge and managing (numerical) uncertainty in a problem domain. Classical logic such as the propositional/first order logic uses a symbolic system to represent conceptual domain knowledge. They provide the basic building block facility for construction of a formal domain model. However, classical logic lacks the capability to revise (and update) knowledge bases consistently. I then reviewed three prominent non-monotonic logics, namely default logic, autoepistemic logic and circumscription. All three approaches have their individual merits and shortcomings. One common problem they all share is the high computational cost, particularly when there are inconsistencies in the acquired domain knowledge. They are also not conductive to frequent, iterated knowledge revision/update and do not meet the critical requirement of our risk modelling framework. In light of this analysis, I gave a detailed review of a fourth approach and showed that Belief Revision is an intuitive and relatively simple mechanism that can be used to construct and maintain a consistent knowledge model of risk for an intelligent agent. Therefore, Belief Revision will play a critical part in the qualitative risk modelling and management for the new framework

Algorithm 14 Inductive Causation (IC) Algorithm

Require: A stable probability distribution $\hat{P}$ on a set of variables V.
Ensure: A partial directed graph, i.e. pattern, compatible with $\hat{P}$.

for all pairs of variables in $\{(a, b)|a, b \in V, a \neq b\}$ **do**
    search for a set S_{ab} such that a and b is conditionally independent with respect to S_{ab}.
    if S_{ab} is not found **then**
        connect a and b with an edge in an undirected graph G.
    end if
end for
for all pairs of nonadjacent a and b in G **do**
    if a and b has a common neighbour c **then**
        if $c \notin S_{ab}$ **then**
            add arrow head at c as $a \to c \leftarrow b$.
        end if
    end if
end for
for all undirected edges a — b in graph G **do**
    if there is a $c \to a$ such that c and b are not adjacent **then**
        change the edge to $a \to b$.
    end if
    if there is a chain $a \to c \to b$ **then**
        change the edge to $a \to b$.
    end if
    if there are two chains a—$c \to b$ and a—$d \to b$ such that c and d is not adjacent **then**
        change the edge to $a \to b$.
    end if
    if there are two chains a—$c \to d$ and $c \to d \to b$ such that c and d is not adjacent and a and d are adjacent **then**
        change the edge to $a \to b$.
    end if
end for

that will be discussed in the next chapter.

I reviewed two important and related knowledge modelling and reasoning techniques, i.e. Bayesian Network and Transferable Belief Model. Both techniques model numerical uncertainty. BN relies on (conditional) probability and works under Closed World assumption, i.e. it assumes a complete domain knowledge; whereas TBM uses mainly belief function and can handle ignorance. Models built in TBM can be translated to a BN model with a formal transformation function, the pignistic transformation function. I use both techniques in the quantitative part of the new framework. I also reviewed possibility theory. It provides two separate schemes for modelling uncertainty either qualitatively or quantitatively based on a common concept of possibility (distribution). Possibility theory is not adopted directly in the current form of new framework because the convenience of using integer in BR is preferred and readily usable (pignistic) transformation function available in TBM. Nevertheless, possibility theory does provide a strong theoretical support to the hybrid approach used in the framework.

Finally, I reviewed two techniques that can be used to distill causal knowledge from domain experts or probabilistic domain models. I use the Ramsey test to capture experts' beliefs in the causal inference relations through simple queries. On the other hand, the inductive causation algorithm provides a systematic way to recover causal structures from Bayesian probabilistic models through structural analysis of the models. Both methods ensure the final product generated from the risk modelling process is a causally connected model that can be used to assess and reason about the domain risks. I am now ready to present an overview of HiRMA and give a broad discussion of the techniques I utilise for the new risk management framework.

Chapter 5

An Overview of HiRMA

The in-depth analysis of risk as a formal concept for intelligent agents and the extensive review of the key knowledge representation and management methods have prepared the solid foundation for me to develop a generalised risk management framework for agents. In this chapter, I give a top-level introduction of the Hybrid, iterative Risk Management framework for Agents, HiRMA. I will first present the architecture of the framework. A high level discussion on the theoretical techniques adopted at three abstraction layers and the theoretical interrelationships between these methods will be provided. Next, I will give an overview of a typical iterative risk modelling and management process under HiRMA. This overview prepares us for the detailed formal presentation of HiRMA beginning with an introduction of a simple domain risk analysis technique based on the generalised risk concept discussed in Chapter 2. This knowledge engineering process is the first step in the risk management process. My technique is significantly influenced by the analysis techniques developed by Aven(2008) and the recommendations from ISO 31000 (2009a). I will use the two benchmark problems to illustrate the analysis technique. The final section of this chapter is the formal theoretical assumptions that underpin the specific risk modelling and management approaches developed in this thesis. Details of these methods will be presented in Chapter 6 and 7 respectively.

5.1 Three-level Framework Architecture

In order to provide a general solution for disparate domains, we need to deal with domain knowledge that is either qualitative or quantitative in nature (Requirement 4 Section 3.2). Under HiRMA framework, domains are grouped under three abstractions based on the characteristics of their domain knowledge. Corresponding modelling methodologies are organised accordingly, as shown in the following table.

Abstract Level	**Theoretical Foundation**	**Model Revision**	**Knowledge Belief Value**	**Model Type**	**Application Level**
High	Propositional/ Possibilistic Logic	Belief Revision	Qualitative	Deterministic Semi-deterministic	Strategic Level
Medium	Transferable Belief Model	Rank Revision	Semi-Qualitative/ Quantitative	Semi-deterministic/ Probabilistic	Tactical management Operational level
Low	Bayesian (causal) Network	Model Selection	Quantitative	Probabilistic	Operational level

Table 5.1: A high-level theoretical architecture of HiRMA.

High Level Abstraction: Relevant domain knowledge, e.g. domain environment, is categorical and cannot be easily quantified. Consequently, the corresponding domain model for risk must be qualitative in nature. The domain information is mostly distilled from the abstract knowledge from domain experts. The resultant model is used for qualitative risk assessments and strategic decision making in which no specific "numbers" are involved. A good example is risk modelling and assessment work required for an international company prior to its decision on making strategic investments or entering into a new market in a foreign country. Modelling and assessment of the country's political and commercial environments are highly qualitative and the final assessment and decision would be categorical. The approach developed in Chapter 6 is specifically designed for this type of problem domain.

Medium Level Abstraction: At this level, domain knowledge is a mixture of qualitative information and quantified opinions from domain experts. The resultant model may be used for either qualitative or quantitative risk assessment and related decision making. The ball passing problem is a representative problem falls under this category. The Transferable Belief Model (TBM) based modelling approach presented in Chapter 7 is designed for domains at this abstraction level.

Low Level Abstraction: Domain information is raw numerical data received directly from the sensors of an agent or other data collection mechanisms. Statistical analysis can be performed on the data. The framework uses modelling and reasoning techniques developed in Bayesian network and produces probabilistic (causal) model for risk evaluation and related decision making.

Table 5.1 gives a high level summary of the key domain characteristics and the corresponding techniques used in HiRMA organised in three knowledge abstraction levels.

Figure. 5.1 shows the theoretical inter-relational mapping of the chosen methods under the context of HiRMA. Specifically, HiRMA framework accommodates a wide spectrum of problem domains. Risk models developed for domains at the high abstraction end are open world qualitative models. The underlying uncertainty representation for qualitative risk is a System of Spheres of possible domain models that are captured using classical logic. Belief Revision is the main mechanism for risk model construction (and modifications). At the other end of the spectrum, i.e. domains in low level abstraction, risk models are closed world quantitative models. I use probability to represent the underlying uncertainty for quantitative risks. Techniques developed for probabilistic Bayesian models are used for risk modelling and management. For domains in the medium abstraction, HiRMA uses belief functions

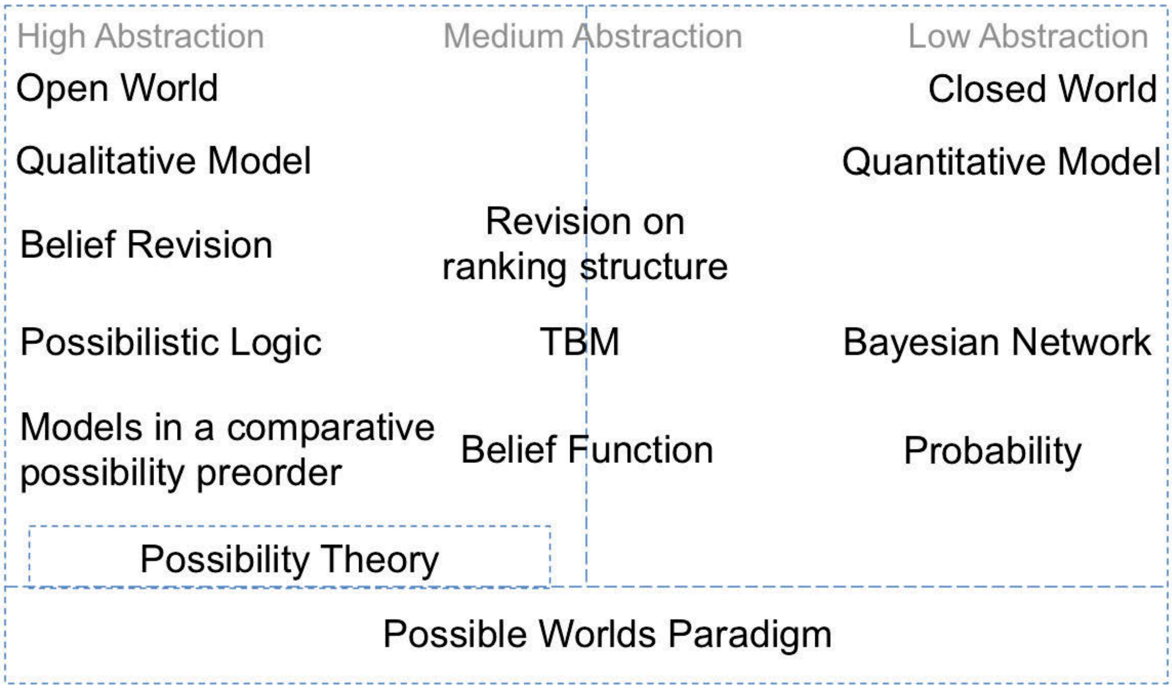


Figure 5.1: High level theoretical relationships between qualitative and quantitative modelling approach for risk modelling and management.

to represent uncertainty for domain risk. The Transferable Belief Model is the basis for risk model at the abstraction level. I developed a corresponding rank revision mechanism for risk model construction and revisions (see Chapter 7). A TBM based risk model can be transformed into a qualitative model or quantitative model[1]. This provides the crucial linkage between the risk models in high abstraction and the models in the low abstraction. Although, possibility theory is not directly utilised in HiRMA, it provides a strong theoretical connection for the methods used in high and medium level abstractions. Finally, every knowledge representation and management method utilised in HiRMA can trace its root to Possible Worlds paradigm and is in accordance to the fundamental analysis of risk in Chapter 2.

[1]It may require additional knowledge inputs.

5.2 Iterative Risk Modelling Process

At the heart of the HiRMA framework is an iterative risk modelling process, Figure 5.2, that can integrate new information with the existing domain knowledge while maintaining the consistency of the knowledge base. This modelling process ensures that an agent can continuously revise and improve its knowledge of the domain. This process represents the overarching strategy of improving an agent's risk management capability in HiRMA. It reflects my fundamental thinking in dealing with risks, more specifically the uncertainties associated with risks. That is, risks arise from imperfect understanding of a problem domain. The ultimate goal of risk management is to improve an agent's ability to minimise risks in its task domain. By improving the agent's knowledge of the problem domain, the uncertainties associated with risks is inherently reduced. The agent can make more accurate assessments and make better decision(s) in dealing with the risks. This becomes particularly important for complex domain environments that evolve with time. To deal with changing domain environment, there are often separations between model revision, learning and update in uncertainty modelling and management literature. The iterative modelling process (and algorithms) in HiRMA essentially unifies model learning and revision together. A risk model in HiRMA framework captures the causally based relations that relevant for a domain based on the latest domain information. As for environmental changes due to actions, HiRMA takes a "passive" approach and does not model actions explicitly. In fact, actions are treated as part of an initial context and risk analysis and modelling are carried out in the initial context (see the following section and Chapter 6 and 7). New domain environment changes will trigger a reanalysis of domain and new initial contexts will be created. This follows the conventional approach in the mainstream risk analysis literature.

Separate risk modelling and management mechanisms are developed in correspondence to the domain knowledge abstractions discussed in the previous section. In particular, the iterative modelling mechanisms work particularly well at medium and high abstraction levels of the framework under the Open World Assumption. The

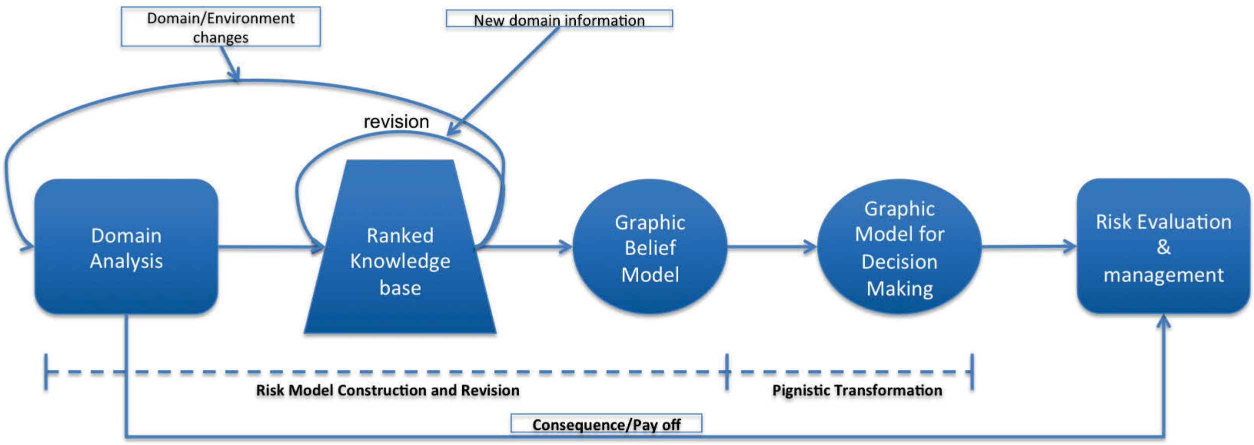


Figure 5.2: A typical risk modelling and management process in HiRMA (at medium domain knowledge abstraction).

Belief Revision based model revision algorithm and rank revision with combination of belief functions (discussed in the following chapters) can be used for initial model construction and subsequent model modifications. A clear (and natural) separation of domain modelling and decision making in which domain models are developed and maintained in an open world setting that permits missing knowledge and the model is "collapsed" down to a closed world setting through formal model translation mechanisms when decision making is required. This open to closed model transformation process captures another fundamental thinking in dealing with risk in a complex evolving environment. That is, we recognise that an agent does not have complete knowledge of its domain under most circumstances. The agent shall keep an open mind and improve its knowledge continuously. However, when it is time to take a concrete action, i.e. doing a risk assessment or making decisions, it shall base its actions on *what it knows best*. The model translation from an open world to a closed world setting is the key that connects modelling process and decision making process together. In particular, the pignistic transformation from TBM (Section 4.5.10) enables the connection between the medium abstraction and the low abstraction in the framework. I will present the details of these mechanisms in the following chapters.

5.3 A Knowledge Description Template for Risk Analysis

In order to understand a problem domain and model its associated risks, we must be able to develop an unambiguous description of the domain and a clear understanding of the domain environment in which an intelligent agent operates. We also need to have a clear description of the design objectives of the agent. Within the domain environment, we analyse and identify plausible scenarios that result from some initial "triggering" context and evaluate the scenarios against the agent's objectives. To assist a knowledge engineer to systematically analyse and model a task domain, I provide a simple domain knowledge description template (Table 5.2) in which the domain is analysed and described in terms of *Objective*, the *Environment*, the *Initial Context*, all possible *Scenarios* and all other *Associated Factors*. This analysis technique closely resembles the descriptive risk analysis technique developed in Aven (2008) with respect to the generalised risk definition introduced in the previous chapter. Concepts described under these five categories form the basic building blocks of a risk model for an agent. Further analysis and additional information helps make causal connections among these concepts and form an ever increasingly complete model that an agent uses to build up its domain knowledge in relation to risk.

In the following sections, I will reformulate the two benchmark problems (described in Chapter 3) using the knowledge description template. Note that, both benchmark problems have been simplified so as to focus the main discussion on the framework without being lost in specific domain details. In later chapters, I mainly use the first problem for explanation and illustration of HiRMA framework; while using the second problem to discuss some interesting issues in developing a risk modelling and management framework.

CATEGORY	DESCRIPTION
Objective	A specific objective an agent intends to accomplish.
Environment	A general description of the working environment that the agent operates for achieving the objective.
Initial Context	An initial triggering event or context for achieving the goal.
Scenarios	All plausible events resulted from the initial context/event.
Associated Factors	Factors within the environment that have direct or indirect influences towards occurrence of the scenarios.

Table 5.2: A domain knowledge description template for knowledge engineer undertaking risk analysis.

The Ball Passing Problem

Objective: Pass a ball between two NAO robots.

Environment: A RoboCup NAO soccer match with two opposing teams and each team is comprised of several identical robots.

Initial Context: a NAO robot R_a is in control of a ball and attempts to kick the ball towards one of its teammates, R_b.

Scenarios: Possible results from the initial context are summarised in the Table 5.3, which lists both qualitative and quantitative consequences (i.e. payoff) of each scenario is calculated using the approach described in Section 2.4.1 with subjective inputs from human experts. For quantitative consequences, I use a pre-determined payoff range of $[-1, 1]$; assign positive values to the preferred scenarios. $S1$ represents the most preferred result with respect to the stated objective; whereas $S2$ represents an exactly opposite outcome. Therefore, I assign 1 and -1 to $S1$ and $S2$ respectively. $S3$ and $S4$, although, not preferred results, are not as undesirable as $S2$. They are assigned with smaller negative values.

SCENARIO	DESCRIPTION	PAYOFF	
$S1$	Ball kicked and caught by R_b.	Excellent	1
$S2$	Ball kicked and intercepted (by an opposition robot).	Severe	-1
$S3$	Failed to kick the ball.	Minor	-0.2
$S4$	Ball kicked and did not reach R_b (no interception)	Bad	-0.5

Table 5.3: A simplified analysis of possible scenarios.

Associated Factors/Variables:

- Distance (D): Distance between R_a and R_b is less than twenty centimetres (in this example analysis),
- Nearby Robots (NR): Any nearby robots (either friendly or hostile excluding R_b) could possibly intercept the ball.

The above description represents a simplified snapshot of ball passing problem which allows us to highlight important features of my risk analysis and modelling approach. Clearly, in a more detailed analysis, many other variables such as *kicking power* of the robot could be added into the model.

The Foreign Exchange Problem[2]

Objective: Maintain a neutral foreign exchange position.

Environment: Australia-US dollar exchange rate fluctuates 0.5% on weekly basis. The firm imports large quantities of electronic goods that take one to two months to manufacture and two weeks for shipment. Payments for the goods could be paid in single or multiple instalments.

[2] **An extended version of this problem is presented in the appendix.**

Initial Context: The firm makes a large order and the payment for the goods will be made in two separate instalments.

Scenarios: Possible foreign exchange positions are summarised in the Table 5.4. Note that, currency hedging refers to various strategies to mitigate financial loss

SCENARIO	DESCRIPTION	PAYOFF	
$S1$	Overseas manufacturer willing to absorb the risk.	Excellent	0.9
$S2$	No currency hedging. Financial losses due to lowering AUD$.	Severe	-1
$S3$	No currency hedging. Financial gain due to rise of AUD$.	Good	0.7
$S4$	100% currency hedging. No net losses and cost of hedging.	Minor	0.2

Table 5.4: A simplified analysis of possible scenarios for FX risk in Australian Dollars.

due to currency fluctuations. A good example of currency hedging is keeping a certain size of foreign currency reserve in banking institutions. Any hedging method normally has some cost associated with it. Similar to the previous benchmark problem, I assign appropriate qualitative and quantitative payoff to the scenarios listed in Table. 5.4 according to their monetary cost to the importer.

Associated Factors/Variables:

- Official Interest Rate (I),
- Inflation (IF),
- Global manufacturing demand (G),
- Currency hedging costs (CH),
- And many more, some known and some unknown.

5.4 Theoretical Assumptions

The base assumption of HiRMA framework is that we can capture and model domain knowledge using classical logic. Specifically, I assume a finite language $\mathcal{L}$ with the usual representations of tautology and contradiction, closed under usual boolean connectives. Let Ω be a set of worlds that corresponds to the interpretation of $\mathcal{L}$. I assume that the conceptual knowledge described under the five categories of the domain knowledge description template (Section 5.3), such as those in the Ball Passing Problem, are *well defined* using language $\mathcal{L}$. That is, there is no conceptual overlap between the knowledge described under these categories and they are self-consistent. In other words, Ω is well *partitioned* for the task domain. I will denote atoms that form the partitions of Ω by capital Roman letters. The language should also be sufficient to describe all possible relationships between the concepts. An agent possesses the domain knowledge K in correspondence to a (set of) worlds in Ω. Furthermore, I adopt the Open World Assumption under qualitative and TBM based quantitative risk modelling. That is, I assume the agent does not have a complete knowledge of the task domain. The agent, however, is able to continuously acquire new information about the domain.

Chapter 6

Risk Management with Qualitative Knowledge

In this chapter, I start the detailed discussion on HiRMA, focusing on how to model and manage risks associated with qualitative domain knowledge. Chapter 7 will discuss risk modelling and management with semi-quantitative and numerical information. In Chapter 8, I will bring all parts together to form a unified and coherent framework. I will discuss how risk models developed at the qualitative level may be linked with models developed at the quantitative level and how a qualitative model may be transformed into a quantitative one, and vice versa.

I build a qualitative risk model based largely on classical logic in the following sections. A qualitative risk model can be viewed a set of plausible (world) models in a comparative likelihood order (Section 2.3.2) or a System of Spheres (Section 4.3.2.2). With the availability of additional domain information, the qualitative risk model can be iteratively revised with methods developed in Belief Revision (Section 4.3). It also can be easily cast as a model within the qualitative possibility framework. When an agent is required to make risk management related decisions, the entire risk model collapses to a single self-consistent and most plausible domain model for reasoning and decision making. Throughout this chapter, I use the Ball Passing Problem (Section 3.1.1) to illustrate the qualitative risk modelling and management process in HiRMA.

6.1 Qualitative Epistemic Reasoning Model

The first task for developing a model of risk is to design an appropriate language to describe and represent specific knowledge associated with risks in a domain. A risk model must be able to formally represent the notions captured through the knowledge engineering process described in Section 5.3. Specifically, the model should be able to represent the end scenarios, and the initial context with various factors within the domain environment that can lead to these scenarios. The causal connections among the factors and scenarios must be represented so that an ontological map of these relationships can be developed to be used in reasoning and managing the factors associated with the domain risks. In the following sections, I will first present an epistemic reasoning model schema for model construction, and introduce the key operator that captures the causal connections between two domain variables. I then give a set of additional postulates that allow the construction of a qualitative graphical risk model based on Directed Acyclic Graph (DAG).

6.1.1 Epistemic Reasoning Model Schema

The qualitative risk management in HiRMA uses an Epistemic Knowledge Repository that consists of four sets of sentences $< \mathbb{C}, \mathbb{F}, \mathbb{S}, \mathbb{R} >$ defined below. The first three sets contain well-formed sentences that represent conceptual knowledge directly derived from the domain analysis process discussed in Section 5.3. Specifically,

$\mathbb{C}$ is a finite set of sentences called *initial contexts*, $\{C_1, ..., C_i\}$, where i is the number of initial contexts. An initial context is a sentence that describes the condition under which interaction between various domain factors will eventually lead to a resultant event, i.e. a *scenario*. For example, in the study of the Ball Passing Problem, robot R_a may come into a state in which it decides to kick the ball in its possession[1]. We capture this state as an initial context which can be represented by $KickBall(R_a) \wedge BallPossession(R_a) \in \mathbb{C}$. This represents all

[1] I assume R_a always acts on its decisions.

the essentials of a particular circumstance under which a corresponding risk model can be developed. Note that, the HiRMA framework does not model *actions* explicitly.

$\mathbb{F}$ is a finite set of sentences that describe K domain *factor*s as $\{F_1, ..., F_k\}$. In practice, a domain factor has direct or indirect influence on the scenario from an initial context. In the Ball Passing Problem, the kicking distance between R_a (when it kicks the ball) and R_b can be an important factor that influences the resultant event. In the model, we can have a factor such as $Distance(R_a, R_b, 20) \in \mathbb{F}$, where $Distance(R_a, R_b, 20)$ is a sentence that expresses that a maximal kicking distance (from R_a to R_b) of 20 centimetres. Factors are reified such that we can have F_1 for $\{Distance(R_a, R_b, x), x < 20 \wedge x > 0\}$[2].

$\mathbb{S}$ is a finite set of sentences called *scenario*s, $\{S_1, ..., S_j\}$, that describes j potential resultant events with direct impact on the agent's objective(s). For example, a scenario in which R_b successfully receives a ball can be represented as $\neg BallPossession(R_a) \wedge BallPossession(R_b) \in \mathbb{S}$; whereas a scenario in which the ball is intercepted[3] by a third party R_c and does not reach R_b would be $InterceptedBall(R_c) \wedge \neg BallPossession(R_a) \wedge \neg BallPossession(R_b) \in \mathbb{S}$.

Note that the elements in $\mathbb{C}$, $\mathbb{F}$ and $\mathbb{S}$ must be consistent with each other. The fourth set $\mathbb{R}$ consists of sentences that are made using the $ReasonFor$[4] schema based on Levi's interpretation of the Ramsey Test (Section 4.7.1). It contains epistemic conditionals between either two factors, or a factor and a scenario. It captures the domain knowledge that a factor is an epistemic reason for another factor or scenario under an initial context. For example, when robot R_a makes a decision to kick the ball in its possession, i.e. under initial context C_1, the (short) kick distance between

[2]For simplicity, I only consider kicking distance less than 20 centimetres in the analysis and modelling of the ball passing problem.

[3]Note that, ball interception does not necessarily mean the interceptor possesses the ball in the end.

[4]We sometimes use *Reason* as a shorthand for this predicate when it is a more appropriate word to use in the following discussion.

R_a and R_b (of $< 20cm$), or F_1, is (one of) the reason(s) for R_b to have the ball fall in its possession, i.e. S_1. We capture this connection as $ReasonFor(F_1, S_1)$. Formally, $ReasonFor$ and its negation are constructed using the Ramsey Test in the following fashion:

- $Cn(\mathbb{C} \cup \mathbb{F} \cup \mathbb{S}) = K$;
- $ReasonFor(\alpha, \beta) \in \mathbb{R}$, iff $\beta \in K * \alpha$;
- $\neg ReasonFor(\alpha, \beta) \in \mathbb{R}$, iff $\beta \notin K * \alpha$.

Thus, $ReasonFor(\alpha, \beta)$ connects a factor, say α, with another factor or scenario, β. It means α is a *reason for* β (Spohn 1983). Compared with the normal implication in classic logic, $ReasonFor$ is a stronger conditional that puts emphasis on the *precedence* of antecedent over consequent. $ReasonFor$ also differs from conditional assertion (Belnap 1970), since it involves hypothetical modification of an existing knowledge base K with the antecedent and does not have a semantic concept of whether the antecedent is true. The conditional will not be rejected simply because the antecedent turns out to be not true. On the other hand, $\neg ReasonFor(\alpha, \beta)$ means α is **not** a reason for β. That is, the consequent cannot coexist with the antecedent in the knowledge base K.

$ReasonFor$ represents a key structural element in the construction of a qualitative epistemic reasoning model for modelling domain risks. $\mathbb{R}$ is the central repository that stores the "causal" knowledge of the inference relationships *under the presence* of a consistent knowledge base K which contains both the antecedents and consequents. This differs from the conditional knowledge bases used in cumulative reasoning models (Kraus, Lehmann & Magidor 1990) that have no additional dependency. The construction and maintenance of $\mathbb{R}$ is at the centre of HiRMA's iterative qualitative risk modelling process. For now, we have several immediate results that follow directly from the definition of $ReasonFor$.

Lemma 1.

$ReasonFor(\alpha, \alpha) \in \mathbb{R}$, (Self Reference)

$\neg ReasonFor(\alpha, \neg\alpha) \in \mathbb{R}$. (Self Consistency)

Proof. From the definition of $ReasonFor$, we have $\alpha \in K * \alpha$ which is the second BR revision postulate (Section 4.3.1.3), therefore, $ReasonFor(\alpha, \alpha) \in \mathbb{R}$. On the other hand, we have $K * \alpha$ must be consistent from the fifth postulate of BR revision. Therefore, $\neg\alpha \notin K * \alpha$. □

Lemma 2. *If α, β are consistent with K, and $ReasonFor(\alpha, \beta) \in \mathbb{R}$, then we have $K * \beta \subseteq K * \alpha$.*

Proof. From the definition of $ReasonFor$, we have $\beta \in K * \alpha$ from $ReasonFor(\alpha, \beta)$. Because both α and β do not contradict with the original K, we have $K * \alpha = K + \alpha$ and $K * \beta = K + \beta$, using the (K*3) and (K*4) postulates of BR revision. This means β is already incorporated in $K + \alpha$ as the additional knowledge with $ReasonFor(\alpha, \beta)$ and no knowledge is removed. Therefore, $K + \beta \subseteq K + \alpha = K * \beta \subseteq K * \alpha$. □

From the lemma above, I can derive another useful result as the following:

Lemma 3. *Given α, β are consistent with K, if both $ReasonFor(\alpha, \beta)$ and $ReasonFor(\beta, \gamma)$ are in $\mathbb{R}$, then we have $ReasonFor(\alpha, \gamma) \in \mathbb{R}$. (Chain of Reasons - transitivity)*

Proof. $ReasonFor(\beta, \gamma)$ means $\gamma \in K*\beta$. From Lemma 2, we also have $K*\beta \subseteq K*\alpha$, then $\gamma \in K * \alpha$. Applying the definition of $ReasonFor$, we get $ReasonFor(\alpha, \gamma)$. □

Lemma 3 gives a transitive property of $ReasonFor$ as long as α and β are consistent with K. This consistency requirement is satisfied under HiRMA assumptions that insist domain knowledge derived from the domain analysis do not overlap each other. Using this transitivity property, an epistemic reasoning chain formed from a number of existing $ReasonFor$ formulae collapse to a simple $ReasonFor$ formula. Hence, this property can be used to boost the performance of the model reasoning process by consolidating the model.

Lemma 4. *Given $\beta \vdash \gamma$, if $ReasonFor(\alpha, \beta)$ then $ReasonFor(\alpha, \gamma)$. (Chain of Reasons - derivativity)*

Proof. We have $\beta \in K * \alpha$ from $ReasonFor(\alpha, \beta)$. Since γ is a logical consequence of β, we have $\gamma \in K * \alpha$. Hence, $ReasonFor(\alpha, \gamma)$ by applying the definition of $ReasonFor$ again. □

Lemma 4 extends the chain of reasoning to the derivatives of existing consequents of $ReasonFor$.

Lemma 5. *If $ReasonFor(\alpha, \alpha \to \beta)$, then $ReasonFor(\alpha, \beta)$.*

Proof. $\alpha \to \beta$ is equivalent to $\neg\alpha \vee \beta$. Hence, $(\neg\alpha \vee \beta) \in K * \alpha$, according to the $ReasonFor$ definition. However, due to the consistency constraint of (K*5) postulate of BR revision, only $\beta \in K * \alpha$ is possible. Therefore, we have $ReasonFor(\alpha, \beta)$. □

From an Epistemic Knowledge Repository, I can define a qualitative *Epistemic Reasoning Model* that satisfies the following postulates RM1 to RM5. These postulates ensure the epistemic reasoning model is consistent with the assumptions of HiRMA. Furthermore, such a model has a graphical representation similar to probabilistic Bayesian models.

(RM1) $Cn(\mathbb{C} \cup \mathbb{F} \cup \mathbb{S}) = K$ (Well Structured)

(RM2) $\mathbb{C}$ and $\mathbb{S}$ are nonempty sets.

(RM3) If $\forall \alpha, \beta \in K$, $ReasonFor(\alpha, \beta) \in \mathbb{R}$ or $\neg ReasonFor(\alpha, \beta) \in \mathbb{R}$, then $\alpha \in \mathbb{F}$; and either $\beta \in \mathbb{F}$ or $\beta \in \mathbb{S}$.

(RM4) $\forall \alpha, \beta \in K, \alpha \nleftrightarrow \beta$, iff $ReasonFor(\alpha, \beta) \wedge ReasonFor(\beta, \alpha) \in \mathbb{R}$. (Mutual Causation Exclusion)

(RM5) $ReasonFor(\alpha, \beta) \wedge ReasonFor(\alpha, \gamma) \rightarrow ReasonFor(\alpha, \beta \wedge \gamma)$. (Common Reason)

Postulate RM1 states that an epistemic reasoning model has domain variables in K, which is a combination of mutually exclusive initial contexts, factors and scenarios. RM2 ensures that K must contain at least one initial context and one scenario. RM3 constrains the element types of the two terms in $ReasonFor$ formula. That is, the head term α must be a factor whereas the tail term β could be either a factor or a scenario. RM4 is a weak form of mutual causation prevention. That is, the situation that α is a reason for β and at the same time β is a reason for α cannot exist in $\mathbb{R}$ if α and β are not logically equivalent. In other words, I assume that an epistemic reasoning model contains only acyclic inference relations[5]. RM5 states that if α is a reason for β and γ, then it is a reason for the combination of β and γ.

6.2 Qualitative Graphical Risk Model

Our primary visual representation of an epistemic reasoning model is a DAG grouped under an initial context[6](Figure 6.1). Within a directed graph, an oval shaped node represents a domain factor, a node with rounded square is a scenario. A directed edge $\rightarrow$ between two nodes represents a direct inference relation between the two variables which captures the notion of epistemic reason. That is, a directed edge that starts from a node A and ends at a node B means that factor A is a direct reason for B, i.e. $ReasonFor(A, B)$. An edge of $A \nrightarrow B$ means that A is **not** a reason for B,

[5]A future extension of the framework may have this postulate removed.

[6]An initial context is not directly represented in the directed graph.

i.e. $\neg ReasonFor(A, B)$. Postulate RM4 ensures the directed graph remains acyclic. Table 6.1 gives a summary of graphical symbols used in the graphical representation and their corresponding meanings. Under the qualitative risk modelling in HiRMA, a graphical epistemic reasoning model for a domain represents one of many possible models of the domain. In the following sections, I will use the Ball Passing Problem to illustrate the qualitative risk modelling process.

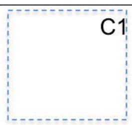
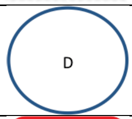
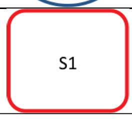
Graphical Symbol	**Meaning**
	An initial context containing an epistemic reasoning model.
	A domain factor.
	A scenario.
	A $ReasonFor$ formula.
	A $\neg ReasonFor$ formula.

Table 6.1: Graphical symbols and their corresponding meanings.

6.2.1 Capturing Knowledge in the Ball Passing Problem

Figure 6.1 shows two simple graphical epistemic reasoning models for the Ball Passing Problem. These models are direct results from the knowledge engineering process (Section 3.1.1) with inputs from domain experts. In this example, I have two initial contexts C_1 and C_2. C_1 is an initial context where R_a has made a decision to kick the soccer ball to R_b, whereas C_2 represents an initial context in which R_a has made no such decision. I have only one factor and two scenarios presented in these simple models. D represents the distance between R_a and R_b is less than 20 centimetres; S_1 and S_2 are the two corresponding scenarios described in Table 5.3. Note that, for a

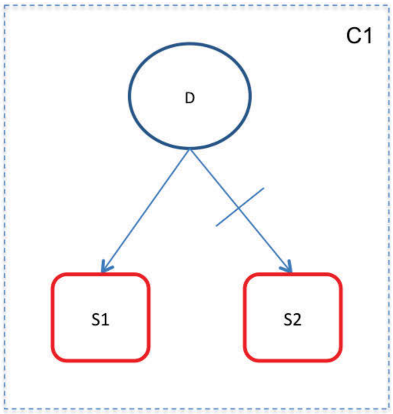


Figure 6.1: Epistemic reasoning models for ball-passing under two initial domain contexts.

clearer and more intuitive visual representation, I have eraised the subscripts of all domain variables and use D to represent the distance factor F_1. Under C_1, the graph shows that the soccer ball kicked by R_a at a distance of less than 20 centimetres will reach and captured by R_b. If R_a does not attempt to kick the ball towards R_b as in C_2, neither S_1 and S_2 result from D. Initial contexts C_1 and C_2 need to be treated separately. Clearly, C_1 is in line with the agent's objective, and therefore, I will continue the discussion focusing mainly on the C_1 model in the following sections.

Syntactically, the graphical models are represented in first order sentences as following:

$C_1 = KickBall(R_a) \wedge BallPossession(R_a)$,

$C_2 = \neg KickBall(R_a) \wedge BallPossession(R_a)$,

$D = F_1 = \{Distance(R_a, R_b, x), x < 20 \wedge x > 0\}$,

$S_1 = \neg BallPossession(R_a) \wedge BallPossession(R_b)$,

$S_2 = \neg BallPossession(R_a) \wedge BallIntercepted.$

$C_1, C_2 \in \mathbb{C}, F_1 \in \mathbb{F}$ and $S_1, S_2 \in \mathbb{S}$.

For C_1 I have the following reasons (captured from the experts) in $\mathbb{R}$:

$ReasonFor(F_1, S_1)$.

$\neg ReasonFor(F_1, S_2)$;

whereas for C_2 I have:

$\neg ReasonFor(F_1, S_1)$,

$\neg ReasonFor(F_1, S_2)$.

6.2.2 Risk Modelling in a System of Spheres

The model presented above represents only a small epistemic reasoning model for determining possible results in association with risk for the problem domain. Results from this static model are deterministic and it does not capture any information in relation to uncertainty which is crucial in modelling risk. The model in this form can be viewed as one of many possible models for the problem domain. As in Section 2.3.2, domain uncertainty is represented qualitatively in a likelihood preorder of all possible (worlds of) causal inference structures for the domain (Figure 2.1). It is easy to construct a System of Spheres structure (Section 4.3.2.2) in which all possible epistemic reasoning models are nested together where the most plausible models stay within the inner most sphere and the outer most sphere contains the least plausible model(s) as shown in Figure 6.2. Syntactically, I use partial epistemic entrenchment ranking structure $\mathbb{E}$ with an Ordinal Epistemic Function E (Section 4.3.4.3). I store all current *reason* formulas in $\mathbb{E}$ according to their individual ranking. Compared with a static epistemic reasoning model in which consistent *reason* formulas are stored in a plain set $\mathbb{R}$, I add additional degree of belief for every *reason* and store them in $\mathbb{E}$ to represent uncertainty in the system. Furthermore, I can have conflicting reasons (with different rankings) coexist in $\mathbb{E}$. I continue to use K as the repository for storing conceptual domain knowledge. Therefore, I have domain model for risk defined as the following:

Definition 6. *In HiRMA, a qualitative domain model for risk* $\mathbb{K}$ *is a structure of* $< K, \mathbb{E} >$*, where* K *is* $Cn(\mathbb{C} \cup \mathbb{F} \cup \mathbb{S})$ *and* $\mathbb{E}$ *is an partial epistemic entrenchment ranking order that stores* $ReasonFor$ *formulae with their degrees of belief.*

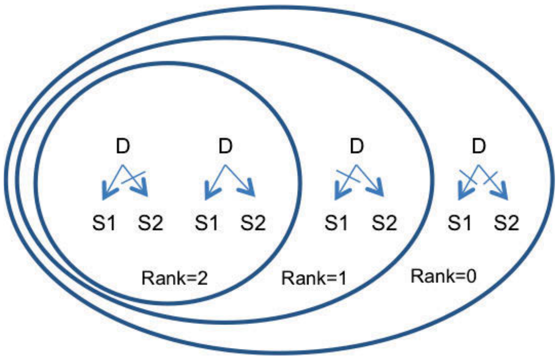


Figure 6.2: A ball passing risk model with all possible epistemic reasoning models captured in a System of Spheres.

6.2.2.1 Epistemic Reasoning Models within a Sphere

It is important to note that neither the likelihood preorder structure nor the corresponding System of Spheres prohibits multiple epistemic reasoning models with the same likelihood or ranking. In other words, we can have more than one model residing in one ring. It simply means an agent has equal degrees of belief for these models. Furthermore, models within one sphere are not required to be consistent with each other. That is, under the qualitative risk modelling framework, it is perfectly fine to have conflicting epistemic reasoning models with equal beliefs in the same sphere as shown in Figure 6.2. In such a case, we have the same degrees of belief for $ReasonFor(D, S_2)$ and $\neg ReasonFor(D, S_2)$. Semantically, this means an agent is unsure (or ignorant) of whether D is a real reason for S_2 since the agent has equal degrees of belief in $D \rightarrow S_2$ and $D \nrightarrow S_2$.

Definition 7. *A* ***vacuous reason*** *is the coupling of a $ReasonFor$ formula and its exact negation with an equal degree of belief (or rank).*

Therefore, the coupling of $ReasonFor(D, S_2)$ and $\neg ReasonFor(D, S_2)$ at rank 2 represents D is a vacuous reason for S_2 at rank 2. Being able to capture and model ignorance is a key requirement for HiRMA (Section 3.2). Epistemic reasoning models of equal ranking must be *combined* into a single self-consistent model when an agent is to make risk management decision based on the domain knowledge captured in these models. I further discuss this topic in Section 6.4.

6.3 Risk Model Construction and Revision

One of the key requirements for the risk modelling and management framework is to construct risk models and revise them according to the latest available domain information so that an agent is able to adapt to the changes in the problem domain. In the qualitative risk modelling, risk model construction and revision are the same process. That is, an agent is able to bootstrap a risk model and gradually build up the model using the same iterative model revision process. This iterative model construction and revision process is largely based on a Belief Revision mechanism. Revision of a risk model consists of two types of model revision, namely revision of domain variables and revision of inference relations among domain variables. I discuss these two types of modifications separately in the following.

6.3.1 Revision of Domain Variables

Change in domain variables typically occurs when there is a change in plausible scenarios (due to changes in agent's objectives) or risk factors (due to changes in domain environment). Revision of domain variables involves either adding new domain variables or removing existing variables from the system. Addition of a new variable, either a risk factor or scenario, means introducing a new node into all epistemic reasoning models in the System of Spheres of a risk model. However, an inference relationship between a newly introduced variable and existing variables in the model remains unknown untill further information of the inference relation is acquired. That is, when a variable is initially introduced into the model, an agent is ignorant of its relationship with other variables in the model.

Proposition 3. *For a risk model $\mathbb{K}$, addition of a domain variable, either a factor or a scenario, satisfies the postulates for expansion in Belief Revision (Section 4.3.1.2).*

Proof. Both factor and scenario are sentences. Addition of a sentence to a knowledge repository K consisting only of sentences does not change the nature of K and modified K is still a knowledge base with sentences. Hence (K+1) is satisfied. (K+2) and (K+3) are satisfied by definition of adding a new variable to K. If a variable α is already in a K, addition of a duplicate does not anything new due to compactness of language $\mathcal{L}$, therefore (K+4) is satisfied. By the same token, (K+5) and (K+6) are also satisfied since a variable represents an atomic piece of knowledge. Addition of a variable has no effect on $\mathbb{E}$ in $\mathbb{K}$, i.e. no $ReasonFor$ is added, removed or modified in $\mathbb{E}$ syntactically. □

I should note that, domain variables produced from the risk analysis process are well-defined and there is no partial overlap between the variables according to the framework assumptions (Section 5.4). This means, the new variable should not "interfere" with any existing variables in K and there should not be any inconsistency between K and the new variable.

6.3.1.1 Semantics of Variable Addition

Addition of a new variable in K is semantically equivalent to introducing a vacuous reason into the system. Specifically, if the new variable is a factor f in $\mathbb{F}$, the existing epistemic reasoning models at every sphere are duplicated $4n + 2m$ times, where n is the number of existing factors (in $\mathbb{F}$) and m is the number of scenarios (in $\mathbb{S}$); if the new variable is a scenario, $2n$ model duplications are required due to postulate RM3. A vacuous reason from x to f, i.e. $ReasonFor(x, f)$ and $\neg ReasonFor(x, f)$ where x is an existing factor in the model, is attached to every pair of duplicated models and forms a new model pair that represent agent's ignorance that x is a direct reason for f. This pairing procedure is repeated for all n factors in the model. In addition, we also need a pair of domain models that capture the ignorance that f is a direct reason for x in the system and x could be either a factor or a scenario[7]. These second model pairs for $ReasonFor(f, x)$ and $\neg ReasonFor(f, x)$ are not required if f is a scenario node due to postulate RM3. I will give a concrete example of this process in Section 6.3.3.1. It must be noted that this model duplication and modification process with *vacuous reasons* is a *virtual* operation that offers a formal semantic interpretation of variable addition within the qualitative risk modelling. It plays an important role in understanding the model revision process. However, there is no specific syntactic construct for vacuous reasons under qualitative risk modelling.

6.3.1.2 Removal of a Domain Variable

Removing an existing variable from a risk model is a relatively simple operation. We simply remove the variable from K and its associated $ReasonFor$ formulas from $\mathbb{E}$. All corresponding arcs in the graphical model are also removed. The rest of knowledge repository and structure of the System of Spheres remain unaffected due to the principle of minimal changes.

Proposition 4. *Given a qualitative risk model $\mathbb{K}$, removal of a domain variable, either a factor or a scenario, satisfies the postulates for contraction in Belief Revision (Section 4.3.1.4), apart from ($K\dot{-}5$).*

[7]We do not consider postulate RM4 which prevents mutual causation problematic in this case, since conflicting models are allowed to coexist at same rank level during the modelling process.

Proof. Removal of a domain variable and associated reason formulas from a K and $\mathbb{E}$ does not change the nature of K and $\mathbb{E}$. Modified K remains as a domain knowledge repository consists of sentences. $\mathbb{E}$ is also intact with remaining reasons unchanged. Hence, (K$\dot{-}$1) is satisfied. It is trivial to prove (K$\dot{-}$2) and (K$\dot{-}$3) are satisfied. All domain variables may be removed from the domain model along with their reasons, the knowledge repository K and $\mathbb{E}$ become empty. Therefore, (K$\dot{-}$4) is satisfied. (K$\dot{-}$5) is too restrictive since we cannot recover the reasons we have removed along with a variable simply by reintroducing the variable into the system. Every variable and its associated reasons have unique meanings within the domain context, a simple change of syntax of their formulas does not change the knowledge they represent. (K$\dot{-}$6) is also satisfied. According to the assumption (Section 5.4), all domain variables are mutually exclusive and there is not overlapping between variables, i.e. for two variable α and β, $\alpha \wedge \beta = \emptyset$. No variables (with their corresponding reasons) will be removed in $K\dot{-}(\alpha \wedge \beta)$. Therefore, both (K$\dot{-}$7) and (K$\dot{-}$8) are satisfied. □

6.3.2 Revision of Reasons

In qualitative risk modelling, the risk model as a System of Spheres captures all plausible (epistemic reasoning) model configurations with different combinations of inference relations (or reasons) among the domain variables. Revision of reasons is achieved through (expert) inputs on inference relationship between variables, i.e. the $ReasonFor$ formulas, and modifying their rankings in $\mathbb{E}$ accordingly. I employ the transmutation strategy, maxi-adjustment (Section 4.3.4.2), to perform revision of reasons. Intuitively, the model revision process essentially shuffles domain models that are consistent with input information to the inner spheres, while moving models that are inconsistent with the input to the outer spheres of the system. Using the ball passing problem as an example, suppose a robot player carries a risk model as Figure

6.2 and the agent receives new information that gives support to that a ball kicked from a distance (of 20 centimetres) D will be the reason for both scenario S_1 and S_2 with a degree of acceptance of 1, i.e. $E(ReasonFor(D, S_1) \wedge ReasonFor(D, S_2)) = 1$. Using adjustment, the risk model is revised to a configuration shown in Figure 6.3.

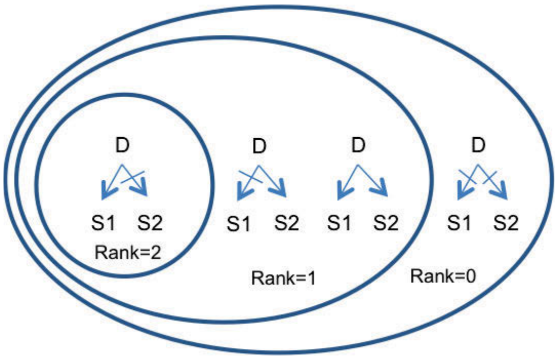


Figure 6.3: Revised ball passing risk model after adjustment with $E(ReasonFor(D, S_1) \wedge ReasonFor(D, S_2)) = 1$.

6.3.3 Model Construction and Revision Algorithms

By combining the revision procedures of domain variables and inference reasons, I can now construct an unified mechanism for qualitative risk modelling based on a modified version of maxi-adjustment transmutation algorithm (Algorithm 16). An agent can apply the same mechanism for risk model construction and model revision. For easy discussion, I separate the risk modelling process into a construction phase and a revision phase. However, there is little technical difference between the two modelling phases. Domain variables are not ranked and do not participate in the adjustment process, as they are in K. *Reason*s between domain variables are placed on the partial epistemic entrenchment rank structure $\mathbb{E}$ according their degrees of acceptance. The modified maxi-adjustment algorithm takes in new belief input for a $ReasonFor$ formula and revise its ranking within the $\mathbb{E}$ structure. The algoritm also ensures that postulate RM4 is enforced by moving down the rank of the formula in the opposite direction of the current reason with new information.

Algorithm 15 Qualitative Risk Model Construction

Require: Risk analysis for the problem domain. Relevant domain variables, i.e. risk factors and scenarios with respect to an initial context form a set $\mathcal{V}$. A predefined maximum ordinal number $\mathcal{O}$. An empty risk model $\mathbb{K} =< K, \mathbb{E} >$.
Ensure: A populated qualitative risk model $\mathbb{K}' =< K', \mathbb{E}' >$.
Initialise $\mathbb{E}$ with $dom(E) = \emptyset$; $max\_degree = 0$.
for all $N \in \mathcal{V}$ **do**
    add N to knowledge repository K.
    if N is a factor **then**
        for $X \in (\mathcal{V} \setminus \{N\})$ **do**
            Solicit inputs on plausible $ReasonFor(N, X)$ (or combination with other reasons) from domain experts.
            if $ReasonFor(N, X)$ should be accepted with a degree i **then**
                $\mathbb{E}' =$ MOD-MAXI-ADJUSTMENT($ReasonFor(N, X), i, \mathbb{E}$).
            end if
        end for
    end if
end for

Algorithm 15 can be used by an agent to build a risk model for a task domain based on an initial domain analysis. It assumes an empty risk model $\mathbb{K}$ that consists of an empty K, an empty $\mathbb{E}$ with an initial Ordinal Epistemic Function E and a predefined maximum ordinal value $\mathcal{O}$ for E. Semantically, $\mathcal{O}$ defines an initial number of nested spheres of the System of Spheres of the risk model. The algorithm goes through every known risk factor and seeks information on plausible reasons (or plausible combinations of reasons) from the factor to the rest of the system from domain experts. Once the agent acquires the relevant knowledge (with a degree of acceptance or confidence), it adds the formula into its knowledge base $\mathbb{E}$ and ranks it using the modified maxi-adjustment (Algorithm 16).

Using the same maxi-adjustment process, Algorithm 17 provides a general procedure for revising an existing risk model. When a new domain variable is introduced into a model, an agent seeks for information on all plausible reasons for this variable

Algorithm 16 Modified Maxi-adjustment Algorithm: to ensure acyclicity in risk model

Require: A $ReasonFor(N, X)$ with a degree of acceptance i, $\mathbb{E}$ with OEF E.
Ensure: A revised $\mathbb{E}'$ with OEF E'.

if $ReasonFor(N, X) \in dom(E)$ **then**
    $degree_{ReasonFor(N,X)} = E(ReasonFor(N, X))$.
else
    $degree_{ReasonFor(N,X)} = \textsc{Degree}(E, ReasonFor(N, X))$.
    Add $ReasonFor(N, X)$ into $\mathbb{E}$ with $degree_{ReasonFor(N,X)}$.
end if
$max\_degree = max(E(dom(E))$.
if $degree_{ReasonFor(N,X)} > i$ **then**
    $E' = \textsc{MoveDown}(ReasonFor(N, X), degree_{ReasonFor(N,X)}, i, E)$
else if $degree_{ReasonFor(N,X)} < i$ **then**
    if $degree_{ReasonFor(N,X)} = 0$ **then**
        $degree_{\neg ReasonFor(N,X)} = \textsc{Degree}(E, \neg ReasonFor(N, X))$.
        $degree_{ReasonFor(X,N)} = \textsc{Degree}(E, ReasonFor(X, N))$.
    else
        $degree_{\neg ReasonFor(N,X)} = 0$.
        $degree_{ReasonFor(X,N)} = 0$.
    end if
    if $degree_{\neg ReasonFor(N,X)} > 0$ **then**
        $E' = \textsc{MoveDown}(ReasonFor(N, X), degree_{\neg ReasonFor(N,X)}, 0, E)$.
    end if
    if $degree_{ReasonFor(X,N)} > 0$ **then**
        $E' = \textsc{MoveDown}(\neg ReasonFor(X, N), degree_{ReasonFor(X,N)}, 0, E)$.
    end if
    $E' = E$.
    $E' = \textsc{MoveUp}(ReasonFor(N, X), i, E)$
else
    $E' = E$.
end if
return E'.

Algorithm 17 Qualitative Risk Model Revision

Require: A qualitative risk model $\mathbb{K} =< K, \mathbb{E} >$. A reanalysis of the domain.
Ensure: A modified model $\mathbb{K}' =< K', \mathbb{E}' >$.

if A new domain variable N is added **then**
    Add-Domain-Variable($N, K, \mathbb{E}$).
else if A $N \in K$ is removed from the system **then**
    Remove-Domain-Variable($N, K, \mathbb{E}$).
else if New information on a existing $ReasonFor(N, X)$ with degree i **then**
    Mod-Maxi-Adjustment($ReasonFor(N, X), i, \mathbb{E}$).
end if
procedure Add-Domain-Variable($N, K, \mathbb{E}$)
    if N is a factor **then**
        for $X \in K$ **do**
            Solicit inputs on plausible $ReasonFor(N, X)$ and combinations with other reasons from domain expert/analysis.
            if $ReasonFor(N, X)$ should be accepted with a degree i **then**
                $\mathbb{E}'$ = Mod-Maxi-Adjustment($ReasonFor(N, X), i, \mathbb{E}$).
            end if
        end for
    end if
    for all $X \in K$ **do**
        if X is a factor **then**
            Solicit inputs on plausible $ReasonFor(X, N)$ and combinations with other reasons from domain expert/analysis.
            if $ReasonFor(X, N)$ should be accepted with a degree i **then**
                $\mathbb{E}'$ = Mod-Maxi-Adjustment($ReasonFor(X, N), i, \mathbb{E}$).
            end if
        end if
    end for
    Add N to domain knowledge base K.
end procedure
procedure Remove-Domain-Variable($N, K, \mathbb{E}$)
    for all $ReasonFor(X, Y)$ in $\mathbb{E}$ **do**
        if $X = N$ or $Y = N$ **then**
            Remove $ReasonFor(X, Y)$ and any combinations with other reasons from $\mathbb{E}$.
        end if
    end for
    Remove N from K.
end procedure

from the existing variables. If the newly added variable is a factor, the algorithm also attempts to acquire knowledge on plausible reason from the new variable to the rest of domain variables. Once relevant information on possible reasons is acquired, the agent adds the reasons into its knowledge base at their corresponding ranks. On the other hand, if a known variable is being eliminated from the system, the agent simply removes all reasons associated with the variable and then removes the variable itself. Revision of individual reason is an application of the modified maxi-adjustment. A useful property of the risk model construction and revision process is the following:

Proposition 5. *Within a ranking structure $\mathbb{E}$, reasons (or combinations of reasons) that are in **direct** conflict, e.g. $ReasonFor(\alpha, \beta)$ and $\neg ReasonFor(\alpha, \beta)$, should not be at the same or adjacent rank level.*

Proof. This is a direct result from the application of the modified maxi-adjustment algorithm. That is, adding a $ReasonFor(\alpha, \beta)$ will ensure its negation (and associated derivatives) being pushed to the lowest ranks. $\square$

The operational complexity of the risk model construction and revision algorithm at the top level is dependent on the number of domain variables in the system. More specifically, the complexity for model construction and general revision is $O(2n+m)$, where m is the number of factors and n is the number of scenarios. As model revision only involves node addition or removal, the complexity is reduced to $O(n)$. However, the real complexity of the algorithm lies within the maxi-adjustment it employs. The maxi-adjustment transmutation is dependent on logical inference used in calculating the known degree of the reason which is pending for modification. It is well-known that computation for logic resolution in first order language is undecidable. For a simple risk model such as the ball passing problem, logic inference used in degree calculation is still decidable and the overall complexity of the algorithm remains in polynomial time.

6.3.3.1 Modelling with the Ball Passing Problem

I now use the Ball Passing Problem to illustrate the qualitative risk modelling process step by step. Under an initial context of C_1, I start from a trivial model of only one variable D and gradually introduce additional domain variables and variable connecting reasons into the system. This risk model uses a partial epistemic entrenchment ranking structure $\mathbb{E}$ with a maximum ordinal of 2. The entire evolution of the risk model is displayed graphically in Figure 6.4. The semantic interpretation of the model evolution can be easily understood using the notions introduced in the previous sections. That is, addition of a variable introduces a vacuous reason to the existing epistemic reasoning model(s) at each sphere. This artificial construction represents an agent's lack of knowledge on the inference relationships between the new variable and the rest of the system. Acquisition of information on these relationships shifts the corresponding reasons (and associated models) apart within the System of Spheres according to the degree of acceptance of the new knowledge. Removal of a variable means all epistemic reasons associated with the variable are also taken out of the system.

The evolution of the risk model shown in Figure 6.4 (page 169) is a semantic graphical representation that corresponds to a series of syntactical operations on reasons in model $\mathbb{K}$ with an OEF E as following:

Initial: An initial model $\mathbb{K} =< K, \mathbb{E} >$ with $D \in K$. An empty $\mathbb{E}$ with an initial OEF E, $dom(E) = \emptyset$.

Step 1: Add a scenario S_1. $K = K \cup \{S_1\}$. Semantically, it means the introduction of a vacuous reason for $D \rightarrow S_1$ on the existing model.

Step 2: Acquired information that supports $ReasonFor(D, S_1)$ with a degree of acceptance 2. Add $(ReasonFor(D, S_1), 2)$ into $\mathbb{E}$ using Maxi-Adjustment resulted with $E(ReasonFor(D, S_1)) = 2$ and $E(\neg ReasonFor(D, S_1)) = 0$.

Step 3: Add variable S_2. $K = K \cup \{S_2\}$ and vacuous reason for $D \rightarrow S_2$.

Step 4: Add $(\neg ReasonFor(D, S_2), 2)$ into $\mathbb{E}$. $E(ReasonFor(D, S_2)) = 0$ and $E(\neg ReasonFor(D, S_2)) = 2$ after maxi-adjustment.

Step 5: Add $(\neg ReasonFor(D, S_1) \wedge \neg ReasonFor(D, S_2), 1)$ into $\mathbb{E}$. $E(\neg ReasonFor(D, S_1) \wedge \neg ReasonFor(D, S_2)) = 1$.

Step 6: Remove variable S_2. $K = K \setminus \{S_2\}$. Remove all reasons related to S_2, i.e. $\neg ReasonFor(D, S_1) \wedge \neg ReasonFor(D, S_2)$, $ReasonFor(D, S_2)$ and $\neg ReasonFor(D, S_2))$ from $\mathbb{E}$.

6.4 Decision Making under Qualitative Risk Model

In the previous sections, I have developed a mechanism for construction and modification of a qualitative risk model in a System of Spheres, or formally, a knowledge base $\mathbb{K}$ that consists of an equivalent epistemic entrenchment structure $\mathbb{E}$ for reason formulas and a belief set K for conceptual domain knowledge (i.e. domain variables). Theoretically, such a structure is rich enough to capture and represent all plausible model configurations according to their plausibilities based on the available domain knowledge. It is suitable for further risk analysis, modelling and evaluation in risk matrix form (Section 2.5.1) in an open world environment. In order to make appropriate decisions in relation to domain risks, it is necessary to reconstruct the model, following the principle of *making decisions based on the best of what we know*. In other words, an agent makes qualitative decisions based on the most plausible model summarised from its knowledge $\mathbb{K}$. For instance, if the risk model for the ball passing problem is as Figure 6.3, an agent can simply use the epistemic reasoning model in the inner-most sphere to help to reach a decision. As this particular model for ball passing is heavily simplified, it does not possess the following technical issues that exist in a typical risk model:

1. Multiple partially conflicting models with equal plausibility such as Figure 6.2;

2. The epistemic entrenchment ranking structure $\mathbb{E}$ is sufficiently large and domain knowledge in terms of epistemic reasons are finely spread across the entire ranking structure. This means a domain model constructed from reasons at any individual rank level is only a partial model.

The basic idea for addressing these two problems is merging the epistemic reasoning models from the most plausible world outwards with ones in the less plausible worlds up to a limit that corresponds to a confidence level. Syntactically, I introduce a merge operator $\mathcal{M}$ that collects epistemic reasons and the combinations of reasons that are consistent with each other from the top epistemic rank downwards, while removing all the reasons and the combinations of reasons that are in conflict with more and equally plausible reasons:

$$\mathcal{M}(\alpha_i, \mathbb{E})(R_i) = \begin{cases} R_i \cup \{\alpha_i\} & \text{if } C_i = \emptyset, \\ R_i \setminus C_i & \textit{otherwise}, \end{cases} \quad \text{if } i \geq j \tag{6.1}$$

where

$$C_i = \{\beta : \beta \in R_i \wedge (\beta \wedge \alpha_i \models \bot)\}$$

and R_i is a consistent set of epistemic reasons (or combinations of reasons) collected from the highest rank level $\mathcal{O}$ to a lower rank of i so far; α_i is a reason (or a combination of reasons) at the rank of i that is pending for merging into R_i; j is the cutoff rank that $0 \leq j \leq \mathcal{O}$. After repeated applications of $\mathcal{M}$ on all reasons (and combinations of reasons) from the highest rank to rank j, a consistent set of epistemic reasons (to rank j) is generated from $\mathbb{E}$. Formally,

Definition 8. *R_j is a consistent set of epistemic reasons, if for all $\alpha, \beta \in R_j$,*

- *both α and β are epistemic reasons or combinations of epistemic reasons;*
- *$\alpha \wedge \beta \not\models \bot$.*

The combination of conceptual knowledge K and R_j forms an epistemic reasoning domain model $\mathbb{K}_j$ with a confidence factor of $\frac{j}{\mathcal{O}}$. I call the repeated $\mathcal{M}$ operation a rank merging process. From Figure 6.2, the agent will end up with an epistemic reasoning model of a single causal connection from D to S_1 with a confidence factor of 1 after the rank merging process.

During this rank merging process, the epistemic reasoning models in the neighbouring spheres are effectively combined together from the inside of the System of Spheres. The merging process may cause reduction in the knowledge carried by the resultant $\mathbb{K}$, due the possible conflicts between the reasons in current R and the reasons that are pending for merge. We need to select an appropriate cutoff rank in order to ensure a good balance between having enough domain knowledge in the resultant $\mathbb{K}$ and maintaining a reasonable confidence level of $\mathbb{K}$, i.e. not to include reasons with very low rankings. I should note that, Proposition 5 ensures that we do not end up with an empty R due to a merge of adjacent ranks. However, in some degenerate cases in which we have a small ranking structure and conflicting reasons cannot be spread out across the structure clearly, performing the merging process across different ranks will yield trivial models that contain no useful knowledge. Therefore, the effectiveness of the rank merging process depends on how clearly structured agent's domain knowledge is.

After the rank merging process, an agent can apply any standard theorem provers and reason with the resultant $\mathbb{K}$. The logic conclusion from the model represents the most plausible scenario that could be reached from the initial context. An agent can be confident in the logical conclusion with a degree of confidence of $\frac{j}{\mathcal{O}}$. The agent can use this result in comparison with the results (of the same confidence) derived under other initial contexts to make a final decision. With the ball passing example, we end up with a result in Figure 6.1. Naturally, a robot soccer player will prefer the model under C_1. In other words, robot R_a concludes that if it does make a ball pass to R_b when the distance between them is less than 20 centimetres, the most likely result is the ball will be received by R_b. Therefore, it is safe to carry out the ball passing action under this condition.

6.5 Discussion

In this chapter, I have presented the details of a formal process of qualitative risk modelling and management for an intelligent agent using classic logic and Belief Revision. To develop an appropriate qualitative risk model for a domain in which the agent operates, The domain and its environment are first analysed using the knowledge engineering process described in Section 5.3. Knowledge acquired from the analysis can be represented using classical first order logic. I introduced a simple epistemic reasoning model schema with a collection of postulates that can be used to construct a graphical epistemic reasoning model for the domain. In particular, I focus on building a model that captures causal connections among various factors in the domain with respect to risks. To this end, I defined *ReasonFor* that embodies the Ramsey Test, in order to capture and represent inference relations among the relevant domain variables. An epistemic reasoning model constructed from the schema represents only one plausible model for the domain under an initial context. Uncertainty in the problem domain is captured in a System of Spheres that contains all plausible model configurations. Within this structure, the epistemic reasoning model that resides in the inner most sphere is the most plausible model corresponds to the reality; while models in the outer spheres are less plausible. Plausibility of these models declines monotonically with increasing sphere levels. Modelling qualitative uncertainty in a System of Spheres structure is a minor variation of the likelihood preorder structure discussed in Section 2.3.2. Combined with consequence analysis of the plausible scenarios, a full qualitative risk matrix can be constructed for risk assessment. When it is required to make risk management related decisions, the agent can employ the most plausible and consistent epistemic reasoning models that generated (up to a predefined confidence level) from a rank merging process, i.e. making decisions based on best of what the agent knows.

By using a System of Spheres structure in the risk modelling, I adopt approaches developed in Belief Revision directly as the core of the qualitative risk modelling process. As I have discussed in Section 2.3.1 and 4.6.2.2, there is a deep underlying connection between qualitative uncertainty modelling and Belief Revision. Uncertainty in a domain arises from an incomplete domain knowledge that an agent possesses. The agent is unable to determine what domain model corresponds to the actual state of the domain due to missing or inaccurate domain knowledge. A natural approach in minimising uncertainty is to acquire additional information and improve the agent's knowledge. Belief Revision provides solid mechanisms for incorporating new information and maintaining a consistent knowledge base. In the modelling process, I assume domain variables do not self-mutate. They are either associated with domain risks or irrelevant for the modelling purpose. Uncertainty resides in the inference relations among the relevant variables. Therefore, the model construction and revision centres around revision of reasons in the model. *Reason*s are placed in an epistemic entrenchment ranking structure $\mathbb{E}$, which is semantically equivalent to a System of Spheres, according to their degrees of acceptance. The revision algorithm used in the modelling process, i.e. maxi-adjustment, ensures a reason that receives strong support from domain information is promoted to a high rank, while its negative counterpart (and derivatives) is demoted to the lowest ranking. This algorithm also ensures the model constructed from the most plausible reasons for decision making is not a trivial model that has most its useful relations cancelled out by their negative counterparts of equal ranking. Furthermore, I must highlight an key feature of this modelling process: both (initial) model construction and model revision use the same Belief Revision mechanism. In other words, model construction and revision operationally are the same process. There are, however, several areas in which the qualitative risk modelling and management process may be improved and extended in the future. I briefly discuss two main areas in the following:

Revision Ranking with Risk Matrix: The ordering in the System of Spheres used in the risk modelling is organised based on the likelihood of domain models.

The degree of belief (or acceptance) for a causal inference relation between two domain variables is also based solely on its likelihood. This ordering/ranking mechanism is not adequate for domains where the plausibility of the domain model inversely correlates to its consequence. In other words, risk modelling based on the likelihood of the domain model will obscure unlikely causal events with extreme consequences. A possible variation for the current revision scheme is to replace likelihood with a proper risk measure number derived from a risk matrix that combines the plausibility of a causal relation and effect from the result of this causal relation (Section 2.5.1). Such an extension enriches the epistemic ranking structure with additional knowledge of effects of every causal relations and makes revision of reasons more meaningful in relation to risk. However, this extension requires exhaustive analysis of every inference relationship in terms of its effects. In addition, this modification changes the nature of resultant domain model. That is, the domain model is no longer a pure epistemic reasoning model and the existing postulates need to be modified accordingly.

Revision and Reasoning with Possibility Theory: As I have discussed in Section 4.6, the (qualitative) possibility theory is closely related to Belief Revision. Therefore, a BR based qualitative risk model could be recast under the possibility theory framework relatively easily. Specifically, the System of Spheres structure can be converted into a fuzzy set with all plausible epistemic reasoning models as members of the set, or equivalently, a fuzzy set of epistemic reasons. Model ranking can be expressed with the member function of the fuzzy set and model revision becomes conditioning of possibility distribution corresponding to the fuzzy set. Model reasoning can be carried out using the inference techniques developed for possibilistic logic.

A risk model under the possibility framework would be theoretically more elegant because the framework consists of qualitative and quantitative theories under the same concept of possibility. Such a risk model would have similar representations for qualitative and numerical domain information. Decision making can be done using works in Dubois et al. (1999, 2001) and Fargier et al. (2005).

In the next chapter, I investigate risk modelling and management when there is numerical information available for modelling domain risks. I develop a risk model construction and revision process that closely resembles the qualitative process presented in this chapter. In fact, there is a strong connection between the two risk modelling and management processes.

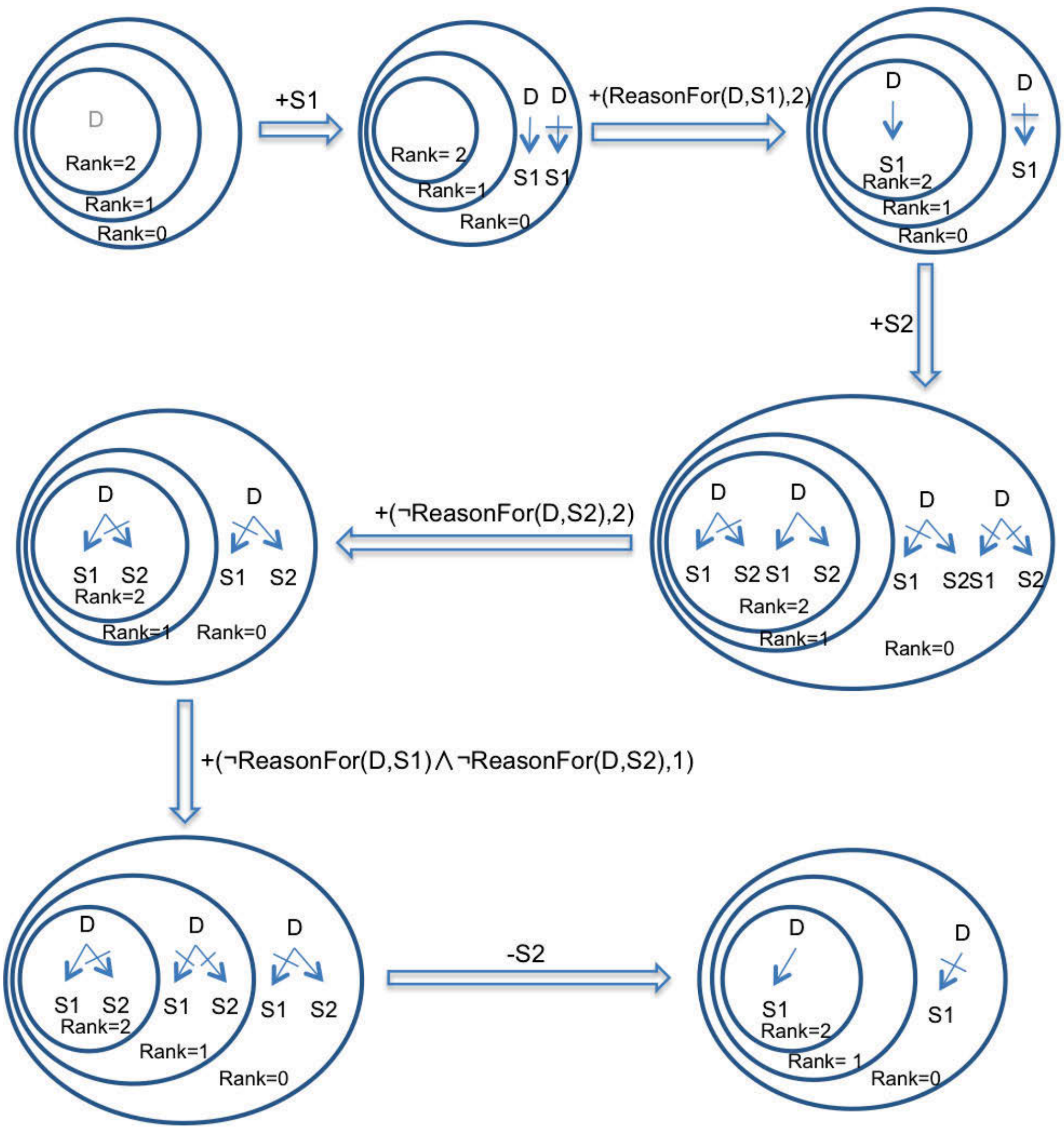


Figure 6.4: Evolution of a simple risk model for ball passing problem. '+' means addition and '-' means contraction. Vacuous reasons are shown in steps in which new variables are added for illustration purpose.

Chapter 7

Risk Management with Quantitative Knowledge

In this chapter, I develop the quantitative part of the HiRMA framework. This part of the framework is divided into two separate modelling and management approaches depending on the nature of the quantitative information available. Specifically,

1. information largely consists of abundant numerical data that capture statistical knowledge of a domain. For example, trading data from a foreign currency exchange market;

2. information largely consists of subjective opinions from experts. The quantified information reflects the degree of beliefs of the experts over (causal) relations among domain variables associated with the risks.

For problem domains falling under the first category, HiRMA employs the standard probabilistic Bayesian network approach with its well developed methods (discussed in Section 4.4.4.1, 4.4.4.2 and 4.4.4.3) to construct probabilistic models from available numerical data. The numeric information is obtained through the knowledge engineering process described in Chapter 5. That is, I first work out the relevant domain variables using the standard domain analysis procedure, and then collect quantitative data associated with these variables through various data collection mechanisms e.g. data feed from stock exchange gateways. Once the probabilistic model is constructed, I use algorithm(s) presented in Section 4.7.2 to recover the causal structure within the model. Subsequent reasoning on this causal structure and decision making with respect to risks can be carried out using the range of inference algorithms surveyed in Section 4.4.5. For further details on the Bayesian modelling approach, I refer readers to the above mentioned sections in Chapter 4 and the extended external materials referenced in these sections. In this chapter, I focus my attention on risk modelling for domains falling under the second category.

I will present an iterative approach for modelling risks based on the Transferable Belief Model, in a similar fashion to the qualitative risk modelling approach developed in the previous chapter. Specifically, a quantitative epistemic reasoning model is developed and maintained in the credal level. It has the similar graphical representation as a probabilistic Bayesian network. When it is necessary to make decisions in relation to risk, the model is translated into a probabilistic model through the so-called pignistic transformation process (Section 4.5.10). Algorithms developed for Bayesian probabilistic models can then be applied for reasoning and decision making. In the following sections, I will first discuss the key elements in this quantified risk modelling technique, in particular *quantified reason* based on the Ramsey Test and belief function. I then present the iterative model construction and revision process based on a real number based ranking structure for *quantified reasons*, in similar fashion as the qualitative modelling approach. I will also establish a direct connection between my risk model and the standard direct evidential network. I then discuss the pignistic transformation as a formal mechanism to translate a model at the credal level into a probabilistic model at the pignistic level for risk assessment and decision making. Finally, I compare the TBM based modelling approach with the probabilistic Bayesian approach. Their respective advantages and disadvantages will be discussed.

7.1 Quantitative Graphical Risk Model

Similar to the qualitative risk modelling, we need to perform the necessary domain risk analysis prior to the initial model construction. Initial contexts, factors and scenarios resulted from the analysis are also expressed in the classical first order language. I use the same directed acyclic graph as the graphical representation for quantitative risk models. That is, rounded square nodes (scenario nodes) represent scenarios and oval shaped nodes (normal nodes) represent factors. Nodes in the quantitative graphic model are typically single value variables. The arcs between nodes represent the epistemic reasons between the associated nodes. In addition, the arcs are coupled with belief functions which represent the degree of beliefs (from experts) in these inference relations. In other words, the arcs are the quantified epistemic reasons for the associated domain variables. For example, in Figure. 7.1 a directed arc from node D to node S_1 means that Distance of 20 centimetres is a direct *reason for* scenario S_1 with a causal strength (explained in later sections) of 0.2. Uncertainty in the risk model is captured in these inference relations. Also similar to the qualitative approach, I only allow arcs initiated from normal factor nodes to scenario nodes but not vice versa.

Central to the quantitative risk modelling is the way we construct and maintain a consistent knowledge base for risk through manipulation of quantified epistemic reasons. I employ a knowledge repository with a ranking structure RS. The quantified epistemic reasons between domain variables are stored in RS according to their relative strengths. Specifically, I map the known inference relations to the rank interval [0,1] according to their corresponding degrees of belief. Inference relations of rank 0 are implausible reasons in the domain; relations of rank 1 are most plausible reasons in the model. For conceptual knowledge, i.e. representation of domain variables, I group them in a belief set K in the same way as the qualitative case. Within the context of RS, K can be viewed as a set of sentences with rank of 1. In other words, I do not rank conceptual domain knowledge and "trust" them to be true implicitly. With the acquisition of new knowledge on the inference relations, I combine existing and new beliefs, and revise the rankings of the inference relations; consequently maintaining the consistency of the knowledge base. Furthermore, this ranking structure facilitates the generation of the final graphical model for risk reasoning, assessment and decision making. In the following sections, I will give a detailed technical discussion on various aspects of this quantitative risk modelling approach. Again, I will use the same ball passing problem to illustrate the entire modelling process.

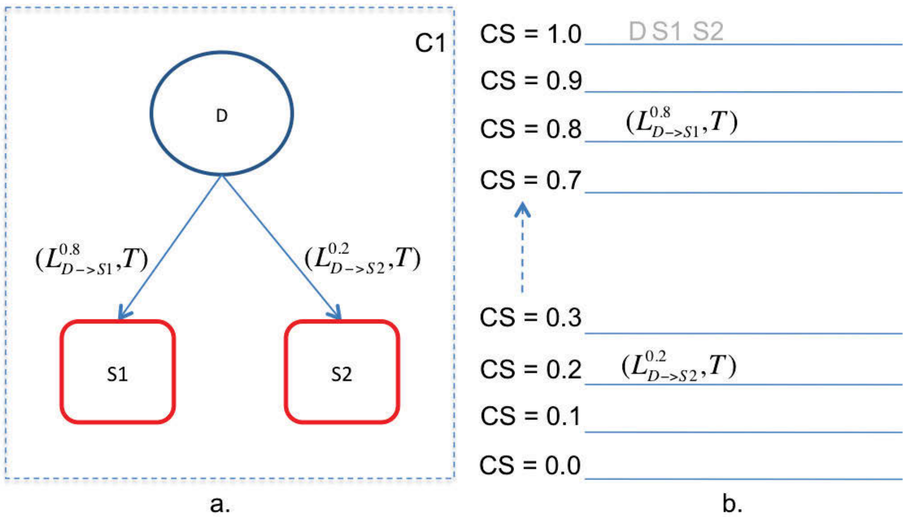


Figure 7.1: Evolution of a quantitative epistemic reasoning model for risk: initial model setup (a) and corresponding ranking structure (b).

7.1.1 Model Quantified Epistemic Reasons

In the quantitative risk modelling process, I focus on capturing the causally based inference relations among the relevant domain variables because such a model represents the stable ontological knowledge of the domain. More importantly, an agent can depend on such a model to take appropriate actions to minimise the consequences from the undesirable scenarios. However, under normal circumstances, an agent does not usually possess domain information with absolute certainty. That is, the agent may not be entirely certain that variable A is a definite reason for B based on the available domain information it possesses[1]. Furthermore, it may not have a full picture of the domain, i.e. some relevant variables and inference relations may be missing from its knowledge base. Therefore, it is natural to model uncertainty directly in the inference relations and taking possible ignorance also into consideration.

7.1.1.1 Lead – Quantified Epistemic Reason

To model domain inference relations with uncertainty, I again invoke the Ramsey Test (Section 4.7.1) to model inference relations as epistemic reasons, the same way I use it in the qualitative modelling approach. In addition, I couple an epistemic reason with a basic belief mass to represent the associated uncertainty. I call such a quantified epistemic reason a *Lead*, denoted as $L^m_{X \to Y}$, where X and Y are two domain variables. It means X is a reason for Y with a belief mass m. Formally, I define *Lead* as the following:

Definition 9. *For a knowledge base K, X and Y are two simple domain variables. We accept a **lead** $L^m_{X \to Y}$ in RS if and only if Y is accepted with a bbm $m(L_{X \to Y})$ in $K * X$, where $K * X$ denotes 'K revised by X'.*

This formal definition[2] can be illustrated through the following queries (to a domain expert): "Based on what you already know about a soccer match (K), if robot R_a kicked the soccer ball towards robot R_b at distance D, will you accept the belief that the ball will be caught by R_b in situation S_1?" and "What value (m) do you put on this belief?". The value one attributes might be related to his or her confidence in the belief or related beliefs. A lead $L^m_{D \to S_1}$ follows immediately with the answers. In fact, these questions can be readily used to capture quantified opinions from domain

[1] Otherwise, if the agent was certain of the causal connection between the variables, it can generate the final outcome deterministically given known factors, i.e. there is no uncertainty/risk to deal with.

[2] I deviate from the simple support function convention used in TBM that the actual support mass is $1 - x$, i.e. I use bbm value as it is.

experts. This definition can be regarded as a quantified version of $ReasonFor$ used in the qualitative risk modelling. A corollary definition is as follows:

Definition 10. *A* ***vacuous lead*** *is a lead* $L_{X\to Y}$ *with a bbm* $m(L_{X\to Y}) = 0$ *denoted as* $T_{X\to Y}$.

A vacuous lead $T_{X\to Y}$ means we are ignorant of whether there is a causal inference relation from node X to Y. It is semantically equivalent to the *vacuous reason* used in the qualitative model.

In terms of graphical representation, $L^m_{X\to Y}$ is represented as an arc starts from node X and ends at node Y. A vacuous lead $T_{X\to Y}$ is normally not shown in a graph. However, for illustration purpose, a vacuous lead is sometimes represented as a dashed arc. It should be noted that all postulates on epistemic reason (Section 6.1.1) are still applicable. In particular, no leads can be initiated from the scenario nodes to normal nodes or other scenario nodes[3]; scenario nodes can only have leads from normal nodes. There is no such a restriction for normal nodes. This ensures the final graphical model is a DAG.

7.1.1.2 Frame of Discernment Ω_X

We also need to define a frame of discernment of possible leads for every variable in a graphical model. For a normal node X, a frame of discernment, Ω_X, is a set of all possible leads initiated from X to the rest of the nodes. Since a scenario (leaf) node has no leads initiated from it, its frame of discernment is an empty set. Therefore, the frame size for a normal node is $i - 1 + j$ where i is the number of normal nodes and j is the number of scenario nodes and the frame size for a scenario node is zero. For example in Figure. 7.1(a), the frame of discernment for node D is $\Omega_D = \{L_{D\to S_1}, L_{D\to S_2}\}$ with $m(L_{D\to S_1}) = 0.8$ and $m(L_{D\to S_2}) = 0.2$. It literally means that we have reasons to believe (kicking from) the distance (of 20 centimetres) leads

[3]This reduces the number of leads we have in the system.

to scenario S_1 with a belief support of 0.8; we also have reasons to believe kicking from this distance will lead to S_2.

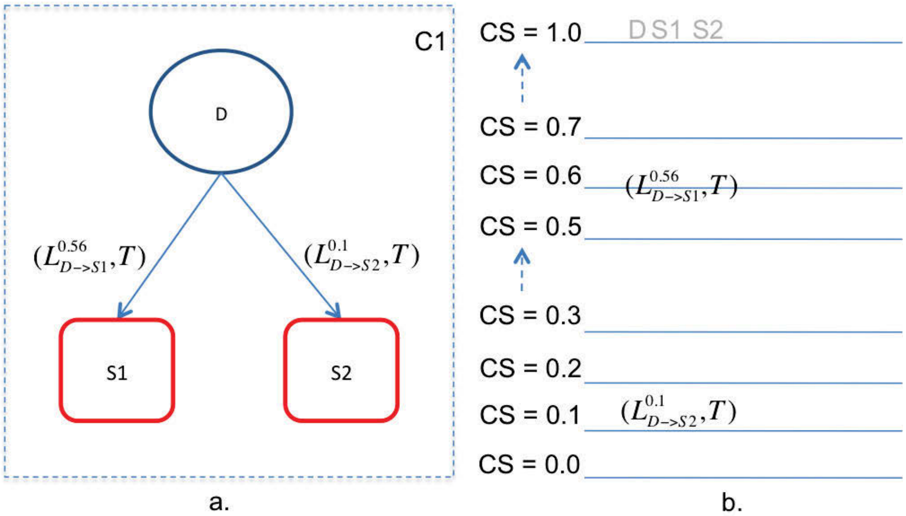


Figure 7.2: Evolution of a quantitative epistemic reasoning model for risk: Fuse additional lead information from another source with $m(L_{D\to S_1}) = 0.5$, $m(L_{D\to S_2}) = 0.3$ and $m(\Omega_D) = 0.2$.

7.1.1.3 Data Fusion in Lead

One key feature in the quantitative risk modelling approach is that leads are conductive to modifications and are able to incorporate new information on the inference relation. This means we are able to combine belief functions associated with leads from disparate sources. That is, we can fuse data that may or may not be consistent from different sources. We employ the TBM conjunctive combination rule (Section 4.5.6) for this data fusion task. Suppose we have an existing model for the ball passing problem as in Figure. 7.1. Specifically, we have bbm $m(L_{D\to S_1}) = 0.8$, $m(L_{D\to S_2}) = 0.2$ and $m(\Omega_D) = 0$. We receive new information on these leads from another source that $m(L_{D\to S_1}) = 0.5$, $m(L_{D\to S_2}) = 0.3$ and $m(\Omega_D) = 0.2$. Note that, a non-zero $m(\Omega_D)$ represents partial ignorance of the expert. After the application of the TBM conjunctive combination rule, the existing bbms become $m(L_{D\to S_1}) = 0.56$, $m(L_{D\to S_2}) = 0.1$ and $m(\emptyset) = 0.34$ (Figure. 7.2). Non-zero value of $m(\emptyset)$ is a measure of the amount of conflict between the existing knowledge on these leads and the new information received. In other words, the resultant leads are not normalised. This is perfectly acceptable in the credal level of TBM. When it comes to decision making or fusing with additional data, the inconsistencies are resolved in the following ways:

Decision Making: We can normalise leads using the pignistic transformation function (Section 4.5.10). Under normal circumstances, we choose the same betting frame as the frame of discernment Ω. After this transformation process, the bbms in the example become $m(L_{D\to S_1}) = 0.85$, $m(L_{D\to S_2}) = 0.15$. This normalised result is equivalent to the result we get from applying the Dempster's rule of combination instead of the TBM conjunctive combination rule.

Iterative Data Fusion: The fundamental assumption in the possible world paradigm is that a world must be self-consistent. Inconsistency that appears the knowledge base is a direct result of an agent's ignorance in the world. Therefore, we may transfer the mass allocated for the empty set to Ω to reflect this ignorance and hence renormalise the bbms. In fact, this redistribution of inconsistency mass produces the same result as if the Yager's combination rule was used in the earlier data fusion (Yager 1987).

7.1.1.4 Latent Lead

We can go one step further and employ a tuple structure similar to the latent belief structure in TBM so that we have an additional diffidence component in a latent lead. We can use this diffidence component to represent the cases that we have some reasons **not** to believe node X leads to Y. For example, $(L^{0.8}_{D\to S_1}, L^{0.5}_{D\not\to S_1})$ means we have reason to believe (with bbm 0.8) D will lead to scenario S_1; at the same time, we also have reasons to believe (with bbm 0.5) our D will **not** lead to scenario S_1. This is particularly useful, since incoming new information can reinforce or weaken or even dismiss the existing support for the leads between the nodes. With the latent lead structure, we can handle contradictory inputs from different knowledge sources. We also have a more powerful way to express the relationships between variables such as expressing "negative probabilities"[4] for the inference relations which is not possible under conventional probabilistic models. In addition, the latent lead structure provides a compact form for storing the inference relations in knowledge base. Note that, an information source may contribute support to either confidence component or diffidence component of a latent lead structure. However, one cannot give both confidence and diffidence supports simultaneously for a particular latent lead. That is, we assume a domain expert does not give contradictory inputs.

Under the latent lead structure, $(L_{X\rightarrow Y}, T_{X\nrightarrow Y})$ and $(T_{X\rightarrow Y}, L_{X\nrightarrow Y})$ (with bbm $m = 1$ omitted) represent the two special cases in which we have full confidence with zero diffidence and no confidence with full diffidence respectively. They become the categorical belief functions and semantically equivalent to $ReasonFor(X,Y)$ and $\neg ReasonFor(X,Y)$ (Section 6.1.1) respectively in the knowledge base. Furthermore, when we have a latent structure with equally weighted confidence and diffidence components, for example $(L^{0.2}_{NR\rightarrow S_1}, L^{0.2}_{NR\nrightarrow S_1})$, it means that we have collected inputs from experts with exactly opposite views, and the end result is that agent has no concrete reasons to believe NR would lead to S_1. In other words, the confidence and diffidence components in the latent structure cancel each other out. We are left with a vacuous lead $T_{NR\rightarrow S_1}$ and remain totally ignorant whether there should be any inference relationship from NR to S_1.

7.1.1.5 The $\dot{\Lambda}$ Operator and Causal Strength

I now introduce a transform operator $\dot{\Lambda}$, based on the similar transform operator Λ in TBM (Section 4.5.5), to map a latent lead structure into an ordinal that represents the *apparent causal strength* (or simply causal strength CS) based on the current belief an agent holds with respect to the corresponding inference relations. We use causal

[4]Intuitively, this could be regarded as *resistance* towards having the causal inference relation.

strength as an overall measure for ranking the inference relation (see the following section). The $\dot{\Lambda}$ operator uses the belief supports of a (latent) lead as the inputs for calculating the apparent causal strength. The $\dot{\Lambda}$ operator has the following properties:

$$\begin{aligned} 0 \leq \dot{\Lambda}((L^u_{x \to y}, L^v_{x \nrightarrow y})) &\leq 1 \\ \dot{\Lambda}((L^u_{x \to y}, L^v_{x \nrightarrow y})) &= 0 \quad \text{if } u = v. \end{aligned} \tag{7.1}$$

Notice that, if a latent structure has equal supports for both confidence and diffidence, then its apparent causal strength is 0. We may have different $\dot{\Lambda}$ operators designed for different domain environments. For example, we may put different discounting factors on confidence and diffidence components. Without losing generality, we select a simple transform operator for the ball passing problem:

$$\dot{\Lambda}((L^u_{x \to y}, L^v_{x \nrightarrow y})) = \begin{cases} u - v & \text{if } u > v \\ 0 & otherwise, \end{cases} \tag{7.2}$$

where u, v are in range of [0,1].

7.1.1.6 The Ranking Structure *RS*

Another key element in the quantitative risk modelling approach is the use of a ranking structure *RS* for storing domain inference relations according to their apparent causal strengths as shown in Figure. 7.1(b). Theoretically, the risk knowledge repository captures all possible inference relations (of various causal strengths) among all relevant domain variables, i.e. latent leads. The ranking structure starts from rank 0 to a maximum rank of 1. Rank 1 is given to those sentences representing the inference relations that are *definitely* plausible to the task domain. Sentences that are the least plausible with respect to the domain, i.e. vacuous leads have the rank of 0. As far as we are concerned, they represent things we are totally ignorant. With this ranking structure, we have a clear picture of relative strengths of causal relationships between various risk factors and scenarios in the model. When it comes to risk assessment and decision making, it is not useful to include all inference relations in the final graphical model. The ranking system provides a facility to filter out all weak and unnecessary inference relations such as vacuous leads and leads with only diffidence.

Now we have all the necessary elements to embark on quantitative risk model construction and revisions. Please note, I assume the inputs are information with respect to either nodes or the inference relation i.e. leads between nodes.

7.2 Risk Model Construction and Revision

Quantitative risk modelling takes the similar iterative approach that is used in qualitative risk modelling. An agent is able to build a risk model from scratch and update the model continuously using the same model construction/revision process. In this section, I bring together ideas discussed in the previous sections and develop revision mechanisms for domain variable and inference relation respectively in similar fashion as in the qualitative case. I then discuss the model construction and revision process illustrated with the Ball Passing Problem.

7.2.1 Revision of Domain Variables

Adding or removing a variable can occur when a domain evolves, e.g. a change in the soccer rule that introduces an additional soccer player into soccer matches. Such a change requires a re-analysis of the domain. As a result, existing risk factors or scenarios may be no longer relevant and have to be removed; while new risk factors or scenario may be added. Under the quantitative risk model structure, addition of a new domain variable carries a subtle but important additional operation: adding a variable node means *adding all possible vacuous leads between existing nodes and the newly added node*. The frame of discernments of all variables in the system (including the newly added variable) are expanded to include these vacuous leads. For example, when node NR is added to a model of Figure 7.2(a), we automatically add four vacuous leads of $T_{D->NR}$, $T_{NR->D}$, $T_{NR->S_1}$ and $T_{NR->S_2}$, and we have $\Omega_D = \{L_{D\to S_1}, L_{D\to S_2}, T_{D\to NR}\}$ and $\Omega_{NR} = \{T_{NR\to S_1}, T_{NR\to S_2}, T_{NR\to D}\}$. Intuitively, it makes sense that when we get to know a relevant notion of *nearby robot* (NR) for the first time, we have no idea how NR is related to the existing variable D, S_1 and S_2. As evidence for these leads emerge, they may become normal leads as their ranks move above 0. Addition of a normal node will entail additional $2n+m$ vacuous leads where n is the number of existing normal nodes and m is the number of existing scenario nodes; whereas adding a scenario node will entail n vacuous leads. This is consistent with postulate RM3 (Section 6.1.1).

Removal a domain variable node means *all leads, regardless of whether they are vacuous leads or normal latent leads, between the retiring node and rest of nodes must also be removed*. Technically, it means the frames of discernment of the remaining nodes are contracted with lead elements associated with the retiring node removed. In practice, we need to transverse the whole ranking structure RS (in which leads are stored) to remove all associated leads (Algorithm 19).

Algorithm 18 Node Addition

Require: A quantitative epistemic reasoning model for risk $\mathbb{K} =< K, RS >$ A new domain variable N. $\mathcal{V}_n$ is a set of normal nodes, i.e. domain factors in K.
Ensure: A modified model $\mathbb{K}' =< K', RS' >$.

```
if N is a domain factor then
    for all X ∈ K do
        Add T_{N→X} in RS at rank 0.
    end for
end if
for all X in V_n do
    Add T_{X→N} in RS at rank 0.
end for
Add the node N to K.
```

Another important property of variable node revision is that addition or removal a node does not effect the rest of leads that not associated with the node. That is, no additional computation is required to adjust the remaining leads. In particular, removal of a node means that bbms associated with any leads from other nodes into

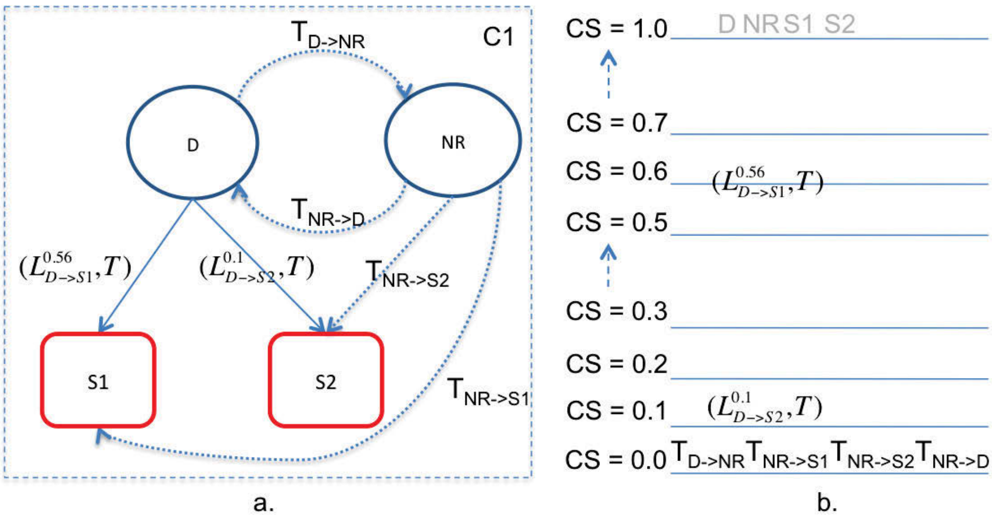


Figure 7.3: Evolution of a quantitative epistemic reasoning model for risk: addition of a risk factor NR (Nearby Robot).

the node are transferred to their respective $m(\Omega)$; other leads in the frame are not disturbed. Take the initial risk model (Figure. 7.1) as an example: removal of S_2 means $L_{D\to S_2}$ is also removed. The bbm associated with this lead is transferred to $m(\Omega_D)$ and the only remaining lead $L_{D\to S_1}$ in Ω_D is unchanged. This property is due to the minimum commitment principle used in TBM.

Algorithm 19 Node Removal

Require: A quantitative epistemic reasoning model for risk $\mathbb{K} =< K, RS >$. An existing variable node X pending for removal.

Ensure: A modified model $\mathbb{K}' =< K', RS' >$.

```
Remove X from K.
for each lead L in RS do
    if L ∈ Ω_X or L ∈ {L_{N→X}, ∀N ∈ K} then
        Remove L from RS.
    end if
end for
```

7.2.2 Revision of Leads

Revision of a quantified epistemic reason, i.e. a lead, simply means fusing the existing knowledge with new information for the inference relation (Section 7.1.1.3); recompute the apparent causal strength (Section 7.1.1.5) and then reshuffle the lead within the ranking structure RS. For example, in Figure 7.4(a), after addition of node NR (Figure. 7.3(a)), new information that gives support of 0.7 to the lead from NR to S_2 arrives. We combine the vacuous lead $T_{NR \rightarrow S_2}$ (automatically introduced with addition of NR) with a latent lead structure $(L^{0.7}_{NR \rightarrow S_2}, T_{NR \nrightarrow S_2})$ using the TBM conjunctive combination rule so that:

$$
\begin{aligned}
&(L^{0.3}_{NR \rightarrow S_2}, T_{NR \nrightarrow S_2}) \owedge T_{NR \rightarrow S_2} \\
&= (L^{0.3}_{NR \rightarrow S_2}, T_{NR \nrightarrow S_2}) \owedge (T_{NR \rightarrow S_2}, T_{NR \nrightarrow S_2}) \\
&= (L^{0.3}_{NR \rightarrow S_2}, T_{NR \nrightarrow S_2})
\end{aligned}
$$

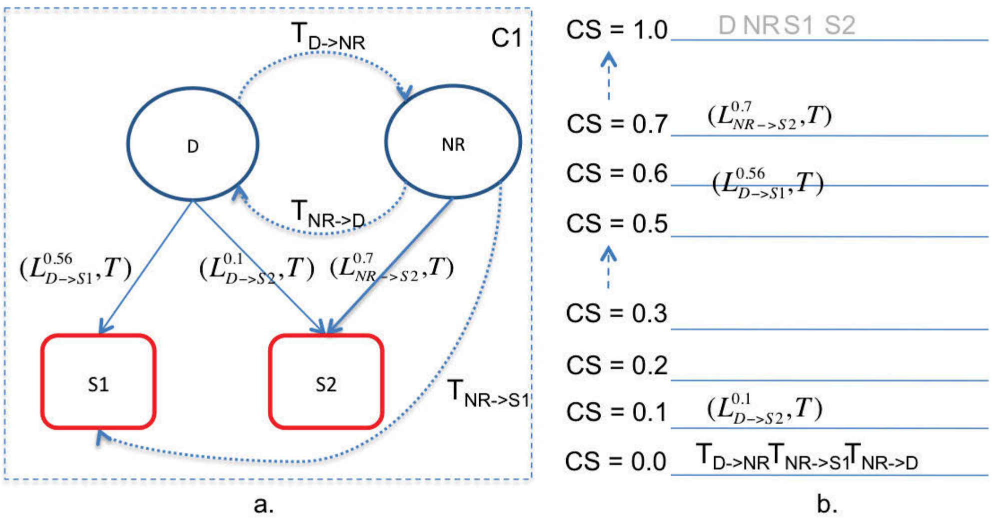


Figure 7.4: Evolution of a quantitative epistemic reasoning model for risk: update lead $L^{0.7}_{NR \rightarrow S_2}$.

Graphically, the ranking structure changes from Figure. 7.3(b) to Figure 7.4(b). This ranking adjustment process can visualised as the invisible arc from NR to S_2

becoming visible while the lead moves up in the ranking structure RS.

7.2.3 Model Construction and Revision Algorithms

Combining the revisions procedure for domain variables and inference relations of the variables, we can now provide an unified risk model construction and revision algorithm based on quantitative knowledge provided by domain experts. As in the qualitative modelling case, I separate the algorithm into two parts: initial model construction (Algorithm 20) and model revision (Algorithm 21), for convenience.

Algorithm 20 Quantitative Risk Model Construction

Require: Risk analysis for the problem domain that produces a set of domain variables $\mathcal{V}$. An empty quantitative epistemic reasoning model for risk $\mathbb{K} =< K, RS >$. One knowledge source that can provide information on the possible inference relations among the variables.
Ensure: A populated model $\mathbb{K}' =< K', RS' >$.

1: **for all** $N \in \mathcal{V}$ **do**
2:     Call Node Addition Algorithm 18.
3:     **if** N is a domain factor **then**
4:         **for all** possible $L_{N\to X} \in \Omega_N$ **do**
5:             Solicit input from the knowledge source for $L_{N\to X}$.
6:             **if** we have support bbm $m > 0$ for $L_{N\to X}$ **then**
7:                 $L_{N\to X} = L^m_{N\to X} \owedge T_{N\to X}$.
8:                 $CS = \dot{\Lambda}(L_{N\to X})$
9:                 Adjust the placement of $L_{N\to X}$ in RS according to CS.
10:             **end if**
11:         **end for**
12:     **else if** N is a scenario **then**
13:         do nothing.
14:     **end if**
15: **end for**

For simplicity, lines 4-10 in Algorithm 20 use simple lead for data fusion with the TBM combination rule. In fact, the operation is performed on the corresponding latent structure. It is also not necessary to add initial vacuous leads into the knowledge base in practice, since they have no physical effects on the actual model. They are implicitly assumed when we have the relevant variables stored in $\mathbb{K}$.

Algorithm 21 Quantitative Risk Model Revision

Require: A quantitative epistemic reasoning model for risk $\mathbb{K} =< K, RS >$. A reanalysis of the domain.
Ensure: A modified model $\mathbb{K}' =< K', RS' >$.
if a new domain variable N is introduced **then**
    Call Node Addition Algorithm 18.
else if an existing variable $N \in K$ is being removed **then**
    Call Node Removal Algorithm 19.
else if additional information on an existing $L_{N \to X}$ with support i **then**
    $L_{N \to X} = L^m_{N \to X} \oslash L^i_{N \to X}$.
    $CS = \dot{\Lambda}(L_{N \to X})$.
    Adjust the placement of $L_{N \to X}$ in RS according to CS.
end if

7.2.4 Graphical Probabilistic Model Generation

The model construction and revision process described above ensures the knowledge repository always remain in a consistent state at the credal level. However, such a model is not useful for risk assessment and decision making because of the two following issues:

- Due to the nature of the quantitative modelling process, i.e. introduction of vacuous leads when a new variable is added; allowing conflicting inputs from disparate sources, a typical quantitative epistemic reasoning model at the credal level contains a large number of vacuous leads and "weak" leads with low ranks.

- The model at the credal level accommodates ignorance; that is, it allows $m(\Omega) > 0$ (or $m(\emptyset) > 0$). However, the resultant model is not normalised such that it is not conductive to risk assessment and decision making.

The first issue can be easily resolved by model pruning and generating graphical models based on "what is important". Within the ranking system, I use a cutoff rank R_{cut} to exclude all vacuous and weak leads from the model and produce a graphical epistemic reasoning model that contains only leads with apparent causal strengths above a desirable level. For example, by excluding all vacuous leads, we end up with a graph in Figure. 7.4(a) *without* the dashed arcs. This filtering process effectively generates graphic models that are close approximations of actual knowledge repository. Depending on my attitude towards low probability risks, I could choose a different rank cut-off point for graph model generation depending on the desired risk profile. A corresponding directed graph can be generated at any time, once a cutoff rank is selected.

Formally, generation of a graph involves combining all relevant variable nodes and associated leads (with ranks above the cut-off point) using the TBM disjunctive combination rule. Note that, the frames of discernment of variables are disjoint, hence, the frame of discernment for the graph is set of all possible leads among the normal and scenario nodes. That is,

$$\Omega_{graph} = \bigcup \Omega_X, \forall X \in \mathcal{V}. \tag{7.3}$$

The frame size $|\Omega_{graph}| = |\mathcal{V}_n(2^{|\mathcal{V}_n - 1|} + \mathcal{V}_s)|$, where $\mathcal{V}_n$ is a set of the normal nodes, $\mathcal{V}_s$ is a set of the scenario nodes and $\mathcal{V} = \mathcal{V}_n \cup \mathcal{V}_s$.

Finally, in order to perform risk assessment and decision making, we need to transform the epistemic reasoning model for risk at the credal level into a normalised probabilistic model at the pignistic level. The pignistic transformation (Section 4.5.10) is applied to every lead in the model and associated bbm is translated into probability according to a betting frame generated from the pruned model with a cutoff R_{cut} . The result model normalisation procedure is listed in Algorithm 22. This procedure truthfully captures the intuition of making assessments or decisions *based on what we know are important.*

Algorithm 22 Quantitative Risk Model Normalisation

Require: A quantitative epistemic reasoning model for risk $\mathbb{K} =< K, RS >$, a cutoff rank R_{cut}, $\mathcal{V}$ is a set of domain variable in K.

Ensure: A normalised model $\mathbb{K}' =< K', RS' >$.

```
for all N ∈ V do
    Let the betting frame A = Ω_N.
    for all L_{N→X} ∈ Ω_N do
        if Λ̇(L_{N→X}) < R_cut then
            Discard L_{N→X} from the A and RS.
        end if
    end for
    for all L_{N→X} ∈ A do
        Apply the pignistic transformation to L_{N→X}.
    end for
end for
```

7.2.5 Modelling with Ball Passing Problem

I now summarise the quantitative risk modelling process illustrated with the ball passing problem step by step. The model construction and revision procedure has already been described in separate steps previously. Here, I pull them together to present a systematic and coherent view of the entire process. Under an initial context of C_1, I start from a trivial model of only three domain variables, one factor node and two scenario nodes. I gradually introduce additional knowledge on the inference relations between these nodes. A factor NR is introduced and later removed from the system. The entire evolution of the quantitative epistemic reasoning model for risk is shown in Figure 7.5 (page 195). The semantic interpretation of the model evolution can be easily understood using the formalism discussed in the previous sections. That is, addition of a variable introduces a set of associated vacuous leads. This artificial construction represents an agent's lack of knowledge on the plausible causal connections between the new variable and rest of the system. Additional information on these leads is combined with the known domain knowledge and forces a (re-)shuffle of the leads within the ranking structure RS according to their updated apparent causal strengths. Removal of a variable means all leads associated with the variable are also removed. When risk assessment is required, every lead in the model is normalised through the pignistic transformation after pruning vacuous and insignificant leads from the model. In other words, the quantitative epistemic reasoning model at the credal level is transformed into a probabilistic model for decision making. Note that, for a simple model for the ball passing problem, model pruning described in Section 7.2.4 is not necessary and I use the original Ω as the betting frame.

The evolution of the risk model shown in Figure 7.5 is listed as follows:

Initial: An initial model $\mathbb{K} =< K, RS >$ with $D, S_1, S_2 \in K$ and vacuous leads $T_{D\to S_1}, T_{D\to S_2}$ in RS.

Step 1: Introduce $L^{0.8}_{D\to S_1}$ and $L^{0.2}_{D\to S_2}$.

Step 2: New input $L^{0.5}_{D\to S_1}$ and $L^{0.3}_{D\to S_2}$.

Step 3: Add variable NR. $K = K \cup \{NR\}$; add $T_{D\to NR}$, $T_{NR\to D}$, $T_{NR\to S_1}$ and $T_{NR\to S_2}$ in RS.

Step 4: Introduce $L^{0.7}_{NR\to S_2}$.

Step 5: Remove NR and associated leads.

Step 6: Normalise $L^{0.56}_{D\to S_1}$ and $L^{0.1}_{D\to S_2}$.

7.3 Discussion

In this chapter, I have focused on the development of a quantitative domain modelling process that takes subjective opinions from domain experts as the main quantitative domain knowledge for modelling and assessing risk. This modelling process is based on the Transferable Belief Model and provides an intuitive and effective approach for generating graphical quantitative epistemic reasoning models for risk analysis and management. My approach attempts to formalise the process of capturing and modelling domain expert knowledge from a risk perspective in an iterative fashion. It requires the same risk analyses as the qualitative risk modelling and management process described in Chapter 6. In addition to identifications of the initial contexts, scenarios, relevant risk factors and epistemic reasons among domain variables, domain experts are required to provide their (degree of) beliefs on the inference relations. Graphically, all variables are represented in a directed graph as nodes and directed arcs between nodes represent the causal connections between the corresponding variables. Syntactically, variables and their causal inference relations are stored in a knowledge base as sentences. The inference relations are placed within a ranking structure RS according to their apparent causal strengths derived from the experts' inputs. The model revision is achieved by combining the existing domain knowledge with new information and adjusting the ranking of the inference relations. Consequently, the knowledge base remains in a consistent state at the credal level. With a graphical model generated from the knowledge repository, I can perform necessary risk analysis and reasoning task using the inference techniques developed for DEVN at credal level. When it is time for making final risk related assessments and decisions, the quantitative epistemic reasoning model can be transformed into a probabilistic model using pignistic transformation after pruning out vacuous and weak leads. Finally, combined with the consequence/pay off resulted from the domain analysis, an agent can make appropriate risk evaluation and take appropriate actions accordingly. This modelling and management process faithfully reflects the way of analysing, modelling and managing risks adopted by a rational being based on *what it knows best* in real life situations.

Most existing approaches for managing quantified uncertainty, and therefore risk, are based on the probabilistic Bayesian network approach. Classical Bayesian networks are deeply rooted in probability and statistical inference and their constructions and refinements (Buntine 1996, Murphy 2002) usually require a considerable amount of numeric data (Zuk et al. 2006) as sample inputs, and there are probabilities to easily work with (Shafer 1990) (Lauritzen & Spiegelhalter 1988)[5]. In many environments

[5]See comments made by Smets.

such as the ball passing problem, obtaining abundant and meaningful data is just not practically feasible. The alternative is to use experts' knowledge from various sources. However, there is no simple formalism to capture potentially conflicting knowledge from multiple experts and create a consistent causally based graphical model. The modelling process introduced in this chapter fills this gap. It is sufficiently general to be adapted for analysis and management uncertainty in a wide variety of domains that rely on the subjective belief/probability from domain experts (Twardy, Wright, Laskey, Levitt & Leister 2009). It also worth noting that my TBM based model does not preclude the use of data and probabilities theoretically; we can use (conditional) probabilities, if it is available, as the belief functions in the risk model.

Another key feature of my risk modelling process is that initial model construction and follow-on model modifications all use the same revision mechanism. It is possible to construct a risk model in one particular domain environment, and adapt the model to a similar environment with minimum modifications. Existing Bayesian based probabilistic models evolution are, in essence, model selections (Buntine 1991, Ramachandran 1998, Lam 1998) based on datasets. However, the reality is that data collected is often limited by what "we think we need". New discoveries, new correlations are often found when different sets of data are combined or data sets change, and solutions based on hidden variables have inherent limitations. All these issues are due to the fact that all probabilistic models are based upon the Closed World Assumption, whereas my TBM based solution use Open World Assumption. Specifically, my approach does not use hidden variables (or another other means) to emulate unknowns. I simply recognise that there are factors and causal relations may be missing in an agent's knowledge. My strategy is to continuously acquire new domain knowledge and revise the existing model. When a new variable is introduced into the system, the frame of discernment (of the system) is expanded to include new plausible causal connections, and any missing knowledge (about inference relations) is assigned to Ω. Conversely, removal of a variable will cause a contraction of the frame of discernment and the removal of all associated leads according to the principle of minimum commitment. Furthermore, the calculation on the degree of belief for an inference relation relies solely on the supporting evidences the agent has gathered. The transformation of a belief model into a probabilistic model is also based on the known evidence and the treatments of beliefs and ignorance. In short, no artificial constructs (e.g. no hidden variable or artificially setting variable values to 'off' state) are used in the entire modelling process. I believe this approach is more natural than the pure probabilistic ones under dynamic, uncertain and open environments that require constant updates/revisions with changing information.

My approach to model construction and revision is currently limited by the fact my input is restricted to processed information(either by human experts or machines) for variables and leads between variables and it cannot work directly with raw data. When there is an abundant amount of quantitative data available for a domain, for example in genome analysis, the probabilistic Bayesian network provides mature algorithm for model building. Therefore, my modelling approach complements the probabilistic modelling process. One needs to have a good understanding of a domain's quantitative nature before selecting a more appropriate approach to model the domain for risk analysis and management.

Finally, belief mass bbm is currently used for ranking evaluation in my method. It is possible to exclude a lead with low belief of occurrence but with a huge potential consequence when we are generating the graphical model. I need to use an appropriate measure for risk, and a better ranking system to reflect measure of risk. To this end, I need to analyse and study the $\dot{\Lambda}$ operator further and to develop more sophisticated transformation operators.

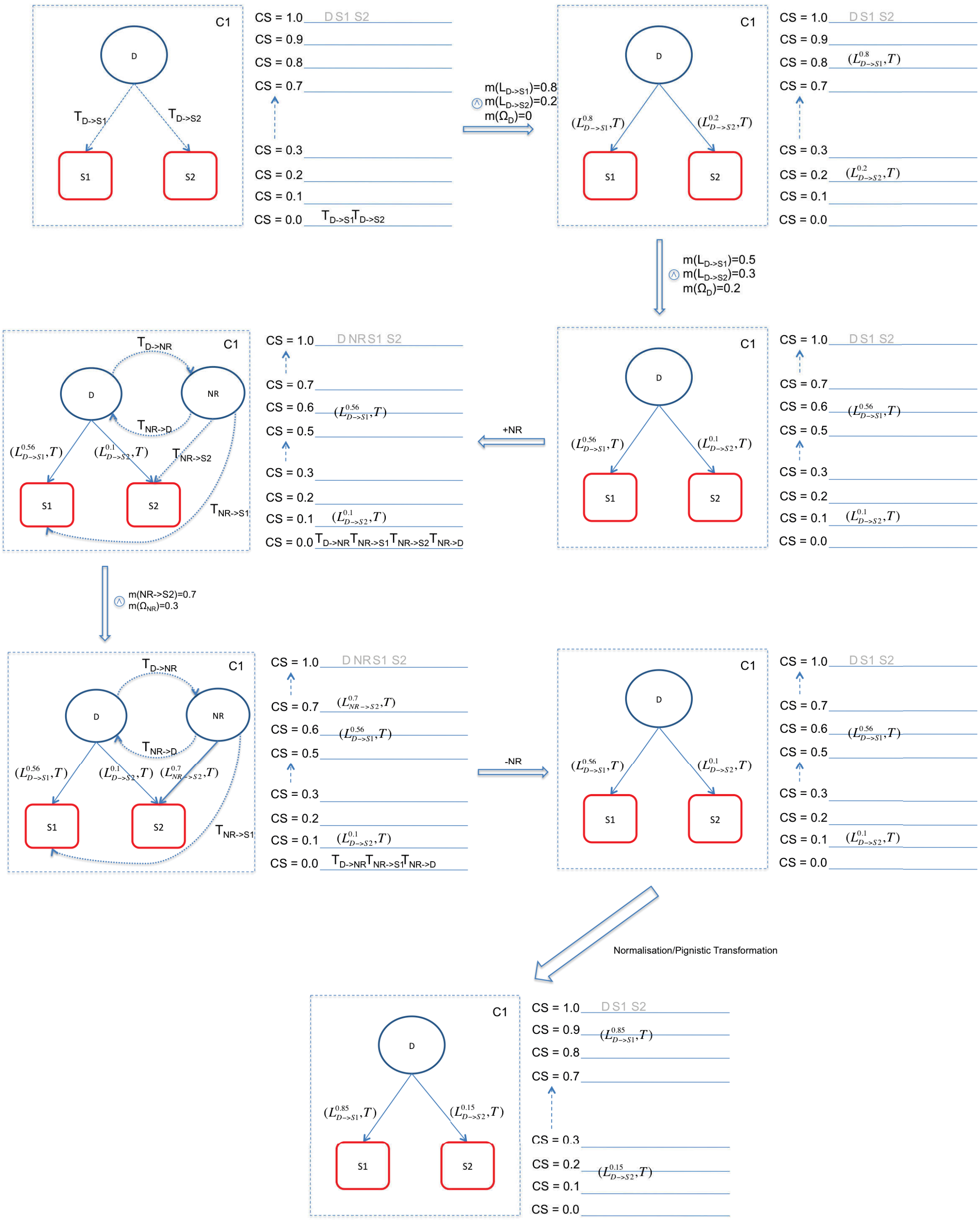


Figure 7.5: Evolution of a simple risk model for ball passing problem. '+' means addition and '-' means contraction. Ⓐ means data fusion with TBM conjunctive combination rule. Vacuous leads are shown in steps in which new variables are added for illustration purpose.

Chapter 8

A Unified Multi-level Iterative Framework

In Chapter 6, I have introduced a qualitative risk modelling and management process and a quantitative approach based on the Transferable Belief Model in Chapter 7. Albeit with their differences in dealing with different types of domain knowledge, these two modelling approaches share many fundamental features:

- Both approaches employ the same domain analysis technique for analysis and acquisition of relevant domain knowledge.
- Both approaches use Open World Assumption and are able to represent ignorance in domain knowledge.
- Both approaches are iterative processes that use the same mechanism for both model construction and revision.
- Both approaches generate intuitive graphical models for risk assessment and decision making.

In this chapter, I will summarise the two risk management approaches together under one unified framework, HiRMA, that provides a generalised risk modelling and management solution for intelligent agents. This generalised solution can be tailored and applied to a wide range of practical domains according to their individual characteristics. This chapter consists of two parts. In the first part, I will first recall the overall architecture of HiRMA presented in Chapter 5. The connection between the qualitative and quantitative modelling approaches will be established at the semantic level. Translations between the two model types will be also discussed. I will then discuss and evaluate HiRMA against the requirements I set out in Section 3.2. In the second part, I will present a framework implementation strategy to guide knowledge engineers to select the most appropriate mechanism and the associated algorithms to develop and manage domain models for agents in dealing with risks. Finally, I will propose a workable generic software architecture for implementation.

8.1 Bridging the Qualitative and Quantitative Divide in Risk Management

In order to accommodate different types of domain knowledge, HiRMA takes a hybrid approach and classify problem domains into three layers of data abstraction, high, medium and low. The high abstraction level is for domains in which only qualitative knowledge can be obtained; the low abstraction level corresponds to domains that contain massive numerical statistical information. The medium level is for domains with mixtures of qualitative and quantitative knowledge. Table 8.1 shows (again) the overall structure of HiRMA. This thesis has focussed on the development of risk modelling and management approaches in high and medium abstraction levels.

It is not difficult to observe that there is a strong connection between the qualitative models at high abstraction level and TBM based quantified models at medium abstraction level. A categorical belief function is equivalent to a first order sentence. A vacuous belief function is equivalent to a coexistence of a sentence with its negation at the same ranking. In other words, a $ReasonFor(A, B)$ is equivalent to a quantified

Abstract Level	**Theoretical Foundation**	**Model Revision**	**Knowledge Belief Value**	**Model Type**	**Application Level**
High	Propositional/ Possibilistic Logic	Belief Revision	Qualitative	Deterministic Semi-deterministic	Strategic Level
Medium	Transferable Belief Model	Rank Revision	Semi-Qualitative/ Quantitative	Semi-deterministic/ Probabilistic	Tactical management Operational level
Low	Bayesian (causal) Network	Model Selection	Quantitative	Probabilistic	Operational level

Table 8.1: A high-level theoretical architecture of HiRMA.

epistemic reason $L_{A \to B}$ with a bbm $m(L_{A \to B}) = 1$; $T_{A \to B}$ is equivalent to have a coupling of $ReasonFor(A, B)$ and $\neg ReasonFor(A, B)$ at an equal degree of belief. In TBM based risk modelling, the causal plausibility (or the apparent causal strength) of a lead is represented with a numerical ranking in the range of $[0, 1]$; whereas the (causal) plausibility of a qualitative epistemic reason is represented with an ordinal ranking number under the OEF structure. This means, the main syntactical difference between the qualitative risk model and the TBM based quantitative model is the ranking structure used to represent the plausibility of inference relations among the domain variables. Compared with qualitative epistemic reasons, the information captured in quantified epistemic reason with belief functions is more rich (e.g. bbm, latent structure) and their manipulations are more sophisticated (i.e. data fusion with combination rules, $\dot{\Lambda}$ operator).

One may appreciate the close connection between qualitative risk model and the TBM based quantitative risk model further from a more intuitive graphically based angle. Suppose we assign a weighting real number to every sphere (according to its likelihood) in the System of Spheres of a qualitative risk model (similar to Figure. 2.1) instead of an integer; every epistemic reason formula is associated with the weighting number of the most inner sphere in which it first appears. Merging all spheres together will result a single DAG model that is semantically equivalent to a TBM based graphical risk model. A cutoff rank used in rank merging process for qualitative risk model performs a similar function as a cutoff rank used in TBM based model for pruning vacuous/weak leads. Furthermore, from the perspective of possibility theory, the qualitative risk model and the TBM based quantitative model can be cast under the qualitative possibility framework and the quantitative possibility framework[1].

The use of belief function and TBM at the medium abstraction is crucial in constructing a pathway that connects the qualitative risk modelling with the probabilistic risk modelling at the low abstraction level. This is due to fact that probability cannot represent full/partial ignorance that is allowed in the qualitative open world risk model. The two level structure of TBM provides us with the transitional pathway for converting an open world model to a closed world model (and vice versa[2]). The pignistic transformation in TBM provides a solid mechanism for perform such a transition. Overall, a risk model developed at the medium abstraction can be transformed into a pure qualitative model (by categorising the leads) if qualitative risk assessment or decision making is required; at the same time it can also be translated into a probabilistic model (through pignistic transformation) if quantitative assessment or decision making is needed. Therefore, in a general sense, the medium level risk modelling approach is the most flexible method within the overall framework.

I would like to highlight that my unified framework rests on mature uncertainty modelling theories and techniques that share a common philosophical foundation of Possible Worlds Paradigm. Chapter 2 has provided a detailed discussion on System of Spheres and probability under the paradigm of possible worlds; while Section 4.5.1 gives a formal definition of belief function from possible worlds. Figure 5.1 gives a graphical overview of the modelling theories and techniques used in HiRMA. It shows the theoretical relationships between these techniques.

[1] I do not give detailed discussion on this particular topic here due to the limited scope of the report.

[2] Since probability is a special class of belief function, I can use the existing probability values as the initial belief value in the open world model.

8.2 Framework Evaluation

In Chapter 3, I have identified the three critical challenges for developing a generalised risk management framework. These challenges are domain *Complexity*, domain *Openness* and domain *Dynamics*. Using the benchmark problems from two disparate fields, the Ball Passing Problem and the FX problem, I have demonstrated these challenges are inherent characteristics of many real-world task domains. My risk management framework must address these issues in order to be practically useful for many real-world domains. Hence, I have set out a set of key framework requirements in direct response to these challenges; I have also discussed the limitations of the mainstream risk management methodologies with respect to these requirements.

The HiRMA framework I have presented in this thesis is designed to meet these requirements so that it is general enough to be adapted to as many problem domains as possible. I have shown in details how a specific risk analysis, modelling and management process may be developed for the Ball Passing Problem under HiRMA in the previous chapters. A similar solution developed for the FX problem is shown in the Appendix A. In the following section, I summarise all key framework features discussed previously in direct correspondence to the framework requirements listed in Section 3.2.

1. Through the multi-level hybrid architecture, HiRMA does not require complete knowledge is available for the problem domain under investigation. If the problem domain is poorly understood and/or with little quantified knowledge, e.g. the two benchmark problems, the high or medium level approaches with proper treatment for ignorance may be adopted for risk modelling and management; whereas, if the domain is well studied with abundant quantitative knowledge available, the low level probabilistic modelling approach should be utilised.

2. The iterative risk modelling process matches naturally with the assumption that agents can continuously acquire new domain knowledge. New knowledge is incorporated into agents' knowledge base through revision mechanisms listed

in Table 5.1. Furthermore, the risk modelling processes at the high and medium abstractions can incorporate potentially conflicting informations from disparate sources.

3. HiRMA is able to deal with both qualitative and quantitative domain knowledge. The previous Section 8.1 provides a detailed high-level discussion on this particular topic.

4. At the high and medium abstractions, the framework uses the notion of *epistemic reason*, which is based on the Ramsey Test, to capture the causally based inference relationships in a problem domain. At the low abstraction, inductive causation algorithm is used to produce and validate the causal structures resulted in the probabilistic models.

5. The evolution of domain knowledge is handled with the iterative modelling processes in the framework.

I conclude that HiRMA achieved the theoretical objectives I have set out in this thesis. In comparison, most of existing risk analysis and modelling tools only meet a subset of these requirements. Methods such as fault tree analysis and event tree analysis (Aven 2008) are born out of system analysis. They produce only qualitative graphical models for risk assessment that do not capture domain uncertainty inherently[3]. On the other hand, probabilistic based model (Bedford & Cooke 2001) typically requires conditional probabilities from domain experts who often have difficulty producing accurate and meaningful numbers (that sums to 1), while taking account of their own ignorances. In fact, no risk modelling methods (we surveyed so far) deal with ignorance explicitly. Furthermore, apart from Hierarchical Holographic Model (Haimes 1981) and HiRMA that allows for hybrid models, no other method can build qualitative and quantitative risk models at the same time. Finally, apart from probabilistic models, none of the existing techniques has built-in

[3]As an extension, probability was later added to the system to express uncertainty.

mechanisms to handle model revisions upon domain changes. There is no formalised and theoretically sound transformation mechanism to convert a qualitative model to a quantitative model, and vice versa. A comparison between HiRMA and several popular risk analysis and modelling techniques with respect to the framework requirements is summarised in Table 8.2.

Methodology	**Feature Requirements**				
	Openness	Continuous Knowledge Inputs	Iterative Model Revision	Qualitative + Quantitative	Causally Stable Model
HiRMA	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
Fault Tree				$\surd^*$	$\surd$
Event Tree				$\surd^*$	$\surd$
FMEA				$\surd^*$	$\surd$
Probabilistic		$\surd$	$\surd$		$\surd$
HHM				$\surd$	$\surd$

Note:
FMEA - Failure Modes and Effects Analysis.
HHM - Hierarchical Holographic Model.
$\surd^*$ - Only supports these features through extensions.

Table 8.2: A feature comparison between HiRMA and the existing risk management methodologies.

After completing this theoretical discussion on the HiRMA framework, I now turn my attention to the practical implementation of the framework. In the following sections, I provide prospective knowledge engineers and system developers with a high level guide for implementing HiRMA framework in intelligent agents.

8.3 Risk Modelling Strategies

Prior to the software design and implementation of HiRMA, the target domain and the capabilities of the intelligent agent operating in the domain should be fully analysed

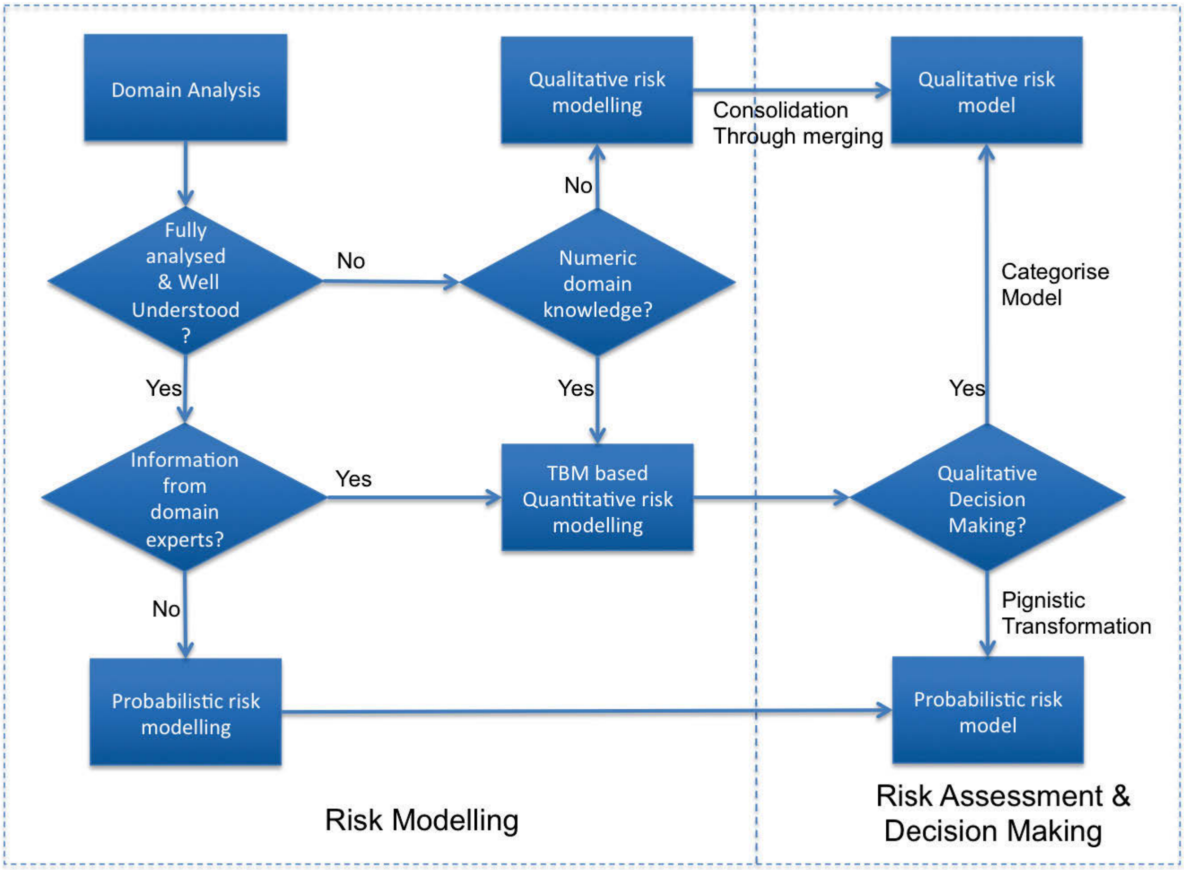


Figure 8.1: A flowchart for selecting appropriate risk modelling and management process.

by the knowledge engineers and system designers. More specifically, the fundamental natures of the domain, in terms of the complexity and openness of its domain knowledge, and dynamics of its environment should be fully investigated. The functions (or objectives) of the agent should be carefully defined. Possible scenarios and associated consequences should be fully analysed with respect to the agent's goals. Additional domain variables should also be studied against the specific domain environments. Section 5.3 provides a simple working template for such domain analyses. Once the initial analysis is completed, the system developer should select the most appropriate modelling techniques from HiRMA for the target domain. Figure 8.1 gives a simple flowchart for determining which modelling process should be adopted for implementation. Note that, the flowchart is partitioned into the risk modelling and the risk assessment/decision making sections in order to highlight the necessary model transformations required in the respective modelling and management approaches. In particular, a TBM based model should be converted into either a pure qualitative model or a probabilistic model depending on the type of assessment or decision is required. If the domain knowledge is largely comprised of quantified subjective opinions from domain experts, I recommend that the TBM based risk modelling approach should be adopted instead of the probabilistic modelling method. This is due to the fact that human experts are usually not good at producing meaningful probability and inputs from different experts are likely to be inconsistent with each other. The TBM based approach provides a formal conflict (in evidence) resolution and belief normalisation (pignistic transformation) mechanism for generating more meaningful probabilities for the risk model. A probabilistic model, on the other hand, simply assumes the subjective probability provided by the experts are meaningful and consistent.

8.4 A Generic Software Architecture

I now present a top-level software design for implementing the risk modelling and management framework. This design can be used for a full or a partial implementation of the framework. Figure. 8.2 gives a schematic overview of this generic software design. Each rounded rectangular box represents a top-level software module or subsystem. The system takes data input from either human domain experts or data feed from other computerised systems. Risk models are constructed and maintained using one of the modelling approaches described in previous chapters. They are stored physically in relational databases. In fact, two databases are used, one database contains the full risk model; the other contains the consolidated risk model ready for risk assessment and decision making. The conversion between the full risk model and

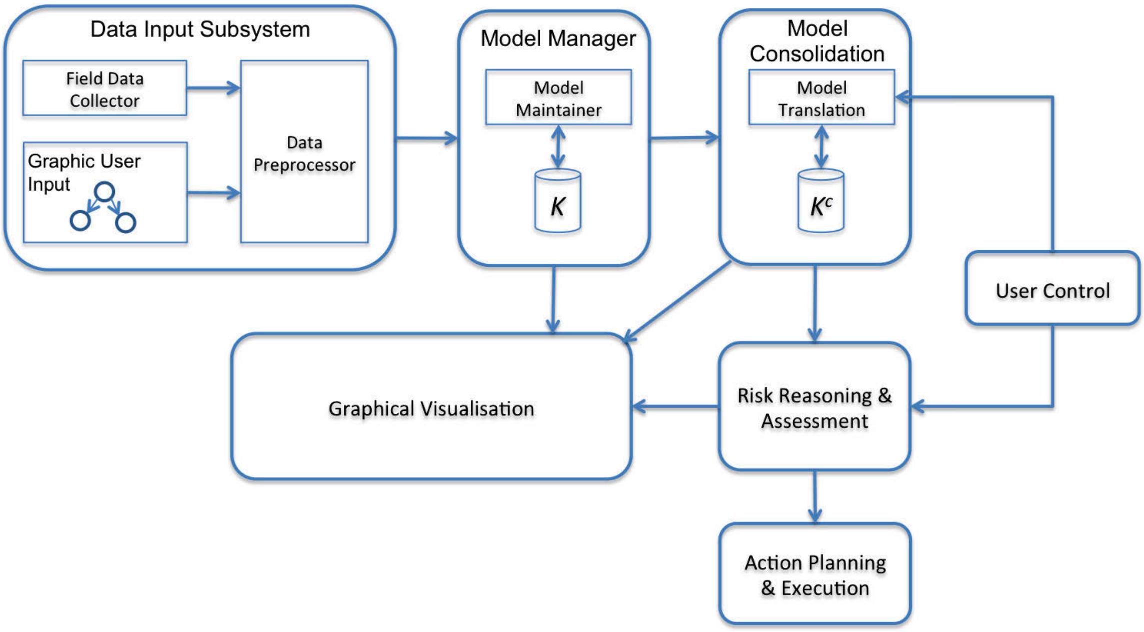


Figure 8.2: A schematic diagram of a generic software architecture for implementation of the HiRMA framework.

the consolidated risk model is done using the translation algorithm with respect to the selected modelling method. The conversion process can be controlled by users. Users may provide additional information for the conversion process. Intermediate and consolidated risk models can be visualised using a graphical presentation module and assessment results can be displayed using risk matrices or risk measure plot described in Section 2.5.1 and 2.5.2.2. This system should form an integral part of an intelligent agent or information system. I provide some additional comments for the key components of the system in the following:

Data Input Subsystem: When one implements a graphical user input interface for collecting domain information for domain experts, we should use the same graphical representation described in the earlier chapters. The user interface should prompt user with a series of questions (similar to questions in Section 7.1.1.1) to capture causally based domain inferencing information. Users should be able to edit directed edges from one domain node to another so to provide information interactively.

The data input subsystem should also have a data preprocessor unit to validate and cleanse data collected from user inputs and other data sources, e.g. ensure quantified data does not go beyond the predefined range, so that data we feed into the subsequent system module is valid and meaningful.

Model Manager: One should implement the selected risk model construction and revision algorithm in this module. Domain knowledge is stored in a relational database. I recommend that domain variables and inference/reasoning relations between domain variables are stored in separated tables. In particular, the inference relation table should contain references to the domain variable tables and have a data field containing the ranking information of the inference relation. The model construction and revision algorithms works mainly on this table, therefore, some optimisation techniques, e.g. data caching, may be considered for optimising system performance.

Model Consolidation: This module should convert the risk model provided from the Model Manager into a normalised model for risk assessment and decision making. It should maintain a separate database so that the "raw" knowledge base constructed from the Model Manager is not destroyed. Execution of this module should be controlled carefully for optimal system performance since model consolidation is only required when an agent needs to assess risk and make decisions. System users should have the control of the model consolidation process. For a performance critical system with minimum resources, e.g. the robot soccer system, we take the entire modelling and consolidation process offline and operate the risk reasoning system on a robot with the consolidated risk model generated from the offline process.

Risk Reasoning & Assessment: Respective reasoning/inference algorithms should be implemented in this module, i.e. automated theorem provers for qualitative model; various exact and approximated inference algorithms for probabilistic models. Results from the reasoning process is presented in qualitative risk matrix or quantified risk measure that can be used in further action planning and execution.

Graphical Visualisation: This module accesses the databases maintained by the Model Manager and Model Consolidation modules and present the respective risk models graphically. It should also be able to display the results from the Risk Reasoning & Assessment module in their appropriate forms. The visualisation module may also be integrated with the graphical user data input and system control subsystems as an integrated graphical system interface for users. Users can, therefore, have a clear picture of internal workings of the agent in terms of managing domain risks and have the key controls of the entire process.

Finally, in terms of actual implementation technology, conventional combination of data processing, communication and database technologies may be adequate for small scale systems. For large scale systems with high data volume, a large number of data source and distributed processing nodes, the recent streaming computing technology(Gedik, Andrade, Wu, Yu & Doo 2008) may be a better programming model for implementing the iterative data flow process found in my framework. This concludes the main presentation of this thesis.

Chapter 9

Conclusions

The goal of this thesis is to apply mature and proven knowledge representation, modelling theories and reasoning techniques developed from the field of Artificial Intelligence in the area of risk modelling and management. Specifically, I have focused on developing a practical risk modelling and managing framework for intelligent agents and information systems so that these systems can assess and manage risks in their respective domains autonomously and intelligently. Since the concept of risk has not been fully developed from the perspective of an intelligent agent, I have formalised a concept of risk and provided a generalised definition of risk for intelligent agents based on the fundamental notions of *uncertainty* and *consequence* (with respect to an agent's objectives). As uncertainty is also a poorly defined concept, I have investigated the notion of uncertainty from the first principles. I provided a formal interpretation and a number of representations of uncertainty for intelligent agents from the possible worlds paradigm. Specifically, uncertainty is represented qualitatively with a likelihood preorder structure of possible worlds; quantified uncertainty is represented using probability or belief function both of which are formulated from possible worlds. These uncertainty representations lead to

1. A modified version of risk matrix which is commonly used in risk management literature to represent qualitative risk. Compared with the standard risk matrix,

the modified version has a formal underlying paradigm of possible worlds and is more expressive in terms of representing possible domain models (in a likelihood order) directly in the table instead of probability bands.

2. A quantified risk measure that is based on the probability and expected utility theory. This risk measure captures an agent's attitude towards risk. It provides a more accurate way of assessing risk with respect to individual agent (or stakeholder).

Overall, the modified risk matrix and the quantified risk measure offer concrete, truthful and disparate representations of risk that are essential for intelligent agents to be able to analyse and assess risks in their respective domains.

My fundamental analysis of "what risk means" for an intelligent agent enables us to develop a simple but effective template for risk analysis based on five important categories, namely, the agent's tasks/objectives, domain environment, initial context, scenarios and associated (risk) factors. Using this template, a knowledge engineer, in collaboration with domain experts, can work out relevant variables associated with the domain risks. My analysis in "how risk can be represented" for an agent reveals disparate uncertainty representations (and risk representation) developed under the Possible World paradigm. This work essentially sets up a selection criteria for the knowledge representation and uncertainty management theories to be adopted in my risk modelling and management framework.

The risk modelling and management framework, i.e. HiRMA, developed in this thesis is intended to be a generalised solution that can be applied and implemented in a wide range of practical domains. Through the analysis of two practical benchmark problems (Section 5.3), namely, the ball passing problem in robotic soccer and risk management of foreign exchange for small enterprises, I observed three critical and challenging issues commonly exist in many real-world domains and environments, i.e. *complexity*, *openness* and *dynamics*. Accordingly, I developed a set of concrete requirements that the HiRMA framework must satisfy in order to address these challenges (Section 3.2). I used these requirements as a set of important criteria (in addition to be consistent with the possible worlds paradigm) to select appropriate knowledge representation and uncertainty management methodologies. Finally, these requirements are also used as the key metrics in the evaluation of the framework.

As a major part of this thesis, I have devoted considerable amount of effort in surveying a wide array of theoretical works in knowledge representation and uncertainty management developed in AI from the past fifty years. My survey ranged from the classical first order logic to non-monotonic logics; from qualitative knowledge representation with pure propositional language to quantitative representation with probability. I have focused on the well-known uncertainty management methodologies such as Bayesian network; discussed their merits and deficiencies with respect to the key challenges. I reached a firm conclusion that no single theory in knowledge representation and uncertainty management techniques can provide a complete and satisfactory solution to the issues of complexity, openness and dynamics. Each individual modelling and reasoning technique is applicable to a specific set of domains and environments. I concluded that a hybrid framework that is comprised of a number of techniques would offer a better generalised solution for covering a wide range of disparate domains. Techniques that ensure a causally connected domain model is developed for dealing with domain risks were also highlighted in the literature review.

The core of this thesis is a multi-level hybrid iterative risk modelling and management framework I developed based on an integration of the key knowledge representation and uncertainty management techniques of classical logics, Belief Revision, Transferable Belief Model and probabilistic Bayesian Network. I presented the framework in two separate parts, the qualitative and the quantitative risk modelling and management. Both parts adopt the same iterative approach towards risk modelling and use the knowledge representation in accordance to the nature of domain information available. Specifically, I use sentences in classical logic to represent domain variables; the causally based inference relations between domain variables are represented qualitatively with epistemic $ReasonFor$ formulas, or quantitatively with quantified epistemic reasons (or leads) with belief functions or probabilities. The uncertainty associated with domain risk is captured in possible domain models in a System of Spheres, conditional belief function or probability. In other words, my risk modelling framework focuses on capturing the domain uncertainty, or (more accurately) an agent's beliefs (based on the available domain knowledge) in the inference relationships within a domain. This modelling approach is based on the reasoning that if an agent is certain of the inference relations between the relevant domain variables, then it can deterministically derive the final outcome from a (set of) initial conditions and therefore no *risk* to speak of. To ensure the framework produces the required causally based domain models, I use the Ramsey Test to capture causal knowledge from domain experts and inductive causation algorithm to validate the final (probabilistic) models are causally connected. In summary, the HiRMA framework tackles the three key challenges in the following ways:

Complexity: By providing multiple risk modelling approaches that work at different data abstraction levels, system developers can select the most appropriate modelling and reasoning techniques from HiRMA for implementation in their target domains. Furthermore, a model constructed at one data abstraction level can be translated to a functionally equivalent model at another abstraction level through a formal conversion mechanism, e.g. the pignistic transformation mechanism.

Openness: By adopting modelling and management technologies based on the Open World Assumption, risk models are developed and maintained in an open world setting in which partial ignorances are allowed. Additional domain knowledge can be continuously integrated into the existing model. The "open" risk models are collapsed down to a closed world model for risk assessment and decision making.

Dynamics: Using an iterative process to continuously update model with newly acquired information, the new knowledge is integrated with the existing knowledge through a formal model revision mechanism such as the modified maxi-adjustment algorithm for rank maintenance. Risk models are being continuously modified and improved along with evolving domains.

The approaches adopted in HiRMA reflect my fundamental stance towards risk modelling and management that an agent is not presumed to possess a complete knowledge or data of its domain and environment. However, it (should) have the ability to acquire new information about the domain (through sensors or other means) and actively acquire such knowledge. The agent keeps an "open mind" and treats its (partial) ignorance about the domain during the entire risk modelling management process in a formal and consistent manner. The agent constantly maintains and updates its domain model according to the latest information to ensure the changes in the domain environment are taken into account. When the agent is required to carry out formal risk assessments and/or making decisions in relation to the domain risks, the agent makes the assessment and the decisions based only on *what it currently knows about the domain*. "Unknowns" are removed from the system and the risk model is normalised based on the existing knowledge and preferences. The resultant model is then used for reasoning and decision making. In other words, the agent maintains its domain model for risks on the open world credal level, and only translates the model to a closed world model when risk assessment and decision making is required. In this way, an agent is able to retain as much domain knowledge as possible, including conflicting information and possible ignorance, in its open model; at the same time only uses the logically consistent and most plausible domain knowledge in its decision makings. I should also note that, the HiRMA framework does not prohibit the development of closed world model. The framework uses the Bayesian probabilistic modelling and reasoning techniques when the target domain is well-understood and has abundant quantitative domain information available.

In summary, HiRMA integrates the well developed knowledge representation and reasoning techniques in the formal logics and the probabilistic methods in modelling and managing risks for a wide range of practical domains. The framework makes a clear separation between domain risk modelling and risk assessment/decision making. Formal translation mechanisms are employed to prepare the model developed in the modelling process for decision making based on the what the agent currently knows best, i.e. knowledge above a certain confidence level. Consequently, the entire risk modelling and management process can be easily implemented in a modularised fashion so that a variety of system requirements or limitations can be satisfied. For example, in a system with limited computational resources such as robotic soccer players, the modelling process have to be implemented offline and only a normalised model is running directly on robots for intelligent risk assessment and decision making.

The framework is flexible enough such that a system developer can select a partial implementation that is mostly suitable for the immediate problem domain. The algorithms incorporated in the framework are directly derived from the mature modelling and reasoning algorithms developed in the respective AI technologies. System developers can implement most of the algorithms with relative ease and can utilise many existing implementations available in the public domain, e.g. Bayesian network learning implementation in Weka[1]. Together with its solid theoretical foundations, the diverse representations for modelling risks in disparate environments, the iterative process for model improvement and the formal model transformation mechanisms for risk assessment and decision makings and the corresponding mature algorithms, I believe HiRMA provides a well balanced and workable solution for modelling and managing risk in intelligent agents from many domains.

With the research work done in this thesis, I have brought risk management and AI, two seemingly unrelated fields of study together by giving an alternative and more formalised approach to risk management, while extending the application of AI into a relatively unexplored area. More importantly, I have filled an important but often neglected area in the designing and implementation of intelligent systems.

[1] http://www.cs.waikato.ac.nz/ml/weka

In the next step in the evolution of my research work is to implement HiRMA framework in a real world oriented system situated in a more complex domain environment so that I can physically demonstrate the crucial benefits HiRMA brings. To this end, we are planning to apply HiRMA in the area of Human Robot Interaction (HRI) particularly on service robots such as PR2[2] because being able to interact with people safely will critically affect the adoption of these robots in the general community. I hope to develop quantifiable metrics to show the benefits of HiRMA. Finally, I hope to continue improving the framework and promote HiRMA for wide adoption in the development of intelligent agents.

[2]http://www.willowgarage.com/pages/pr2/overview.

Appendix A

An Extended FX Example

In this appendix, I extend the FX benchmark problem 3.1.2 to demonstrate a possible implementation of HiRMA at median and high abstraction level. I should note that this extended example remains a simplified model, and actual implementation for deployment in the real world environment will require further development of the model constructed below. Furthermore, I present the overall implementation and do not go through every step of the process.

A.1 Initial Risk Analysis

Objective: Maintain a neutral foreign exchange position for a median size electric goods importing firm in Australia (AU).

Environment: The Australia-US dollar exchange rate fluctuates 0.5% on weekly basis on average, may vary more violently ($> 5\%$) due to fast changing global environment. The firm imports large quantities of electronic goods that take one to two months to manufacture and two weeks for shipment. Payments for the goods could be paid in single or multiple instalments.

Initial Context: The firm makes a large order and the payment for the goods will be made in two separate instalments.

Scenarios: Possible foreign exchange positions are summarised below.

SCENARIO	DESCRIPTION	PAYOFF	
$S1$	Overseas manufacturer willing to absorb the risk.	Excellent	0.9
$S2$	No currency hedging. Financial losses due to lowering of AUD$.	Severe	-1
$S3$	No currency hedging. Financial gain due to rise of AUD$.	Good	0.7
$S4$	100% currency hedging. No net losses and cost of hedging.	Minor	0.2

Table A.1: An analysis of possible scenarios for FX risk in Australian Dollars (AUD).

Associated Factors: All factors listed below consider their corresponding variations in a period of two months.

- Variation in AUD above 5% (VC),
- Variation in AUD below 5% (MC),
- Differential Interest between US and AU (DI),
- Interest Rate in AU (IR),
- Inflation in AU (IF),
- Current Account deficit (CA),
- Public and private Debt in AU (PD),
- Demand in Resources (DR),
- Growth rate in Asia (GA),
- Manufacturer's Willingness to retain orders (MW),
- Global manufacturing Demand (GD),
- Europe/US Debt Crisis (DC),
- Currency Hedging costs (CH),
- Hedging in full(HG).

A.2 Analysis and Selection of Modelling Process

As small/median size company specialised in import, I may not have abundant amount of quantitative foreign exchange data available for analysis and modelling. Knowledge about FX are obtained mainly from experts from financial institutions, outlook guidances from government reports, economic news and the company's past operational experiences (e.g factor MW). It is unlikely that the firm has full knowledge of the FX market and most of existing knowledge are partially quantifiable. Hence, according to the methodology selection flowchart Figure 8.1, I adopt the TBM based modelling and management process for my implementation.

A.3 Knowledge Databases for Modelling FX risk

A.3.1 Database Structure for $\mathbb{K}$

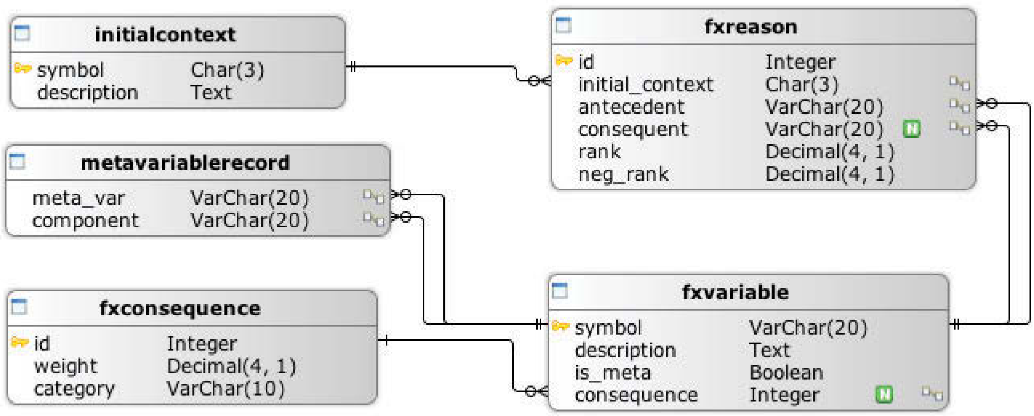


Figure A.1: Database schema for a FX risk model at credal level.

Figure A.1[1] shows a graphical database schema for the risk model. I store domain risk factors and scenarios in the `FXVariable` table. Normal factors have null *consequence* whereas scenarios have references to the `FXConsequence` table. The *is_meta* flag indicates whether the variable is a meta-variable that is a combination of concrete variables. Meta-variables are used to to capture epistemic reasons between a domain variable and a combination of other variables, e.g. *DC* is a reason for *GD and MW* at the same time. `MetaVariableRecord` is used to record the many-to-many relationships between meta-variables and associated concrete variables. Depending on the data type of *rank* and *neg_rank* attributes, an entry in `FXReason` table is either a *ReasonFor* formula with its negation, or a quantified epistemic reason, i.e. latent *Lead*. Integer *rank*s mean qualitative reasons, and Decimal *rank*s imply quantitative reasons. *neg_rank* is the ranking for the negation of a reason. Both *antecedent* and *consequent* are foreign reference keys from `FXVariable` table. Note, *antecedent* must not be `NULL`, whereas *consequent* can have a `NULL` value. In fact, a FX reason with a NULL *consequent* represents $\Omega_{antecedent}$. When I add a variable, e.g. *MW* into the `FXVariable` table, I should also have a corresponding Ω_{MW} with *rank* of 1.0 in the `FXReason` table. Listing A.1 shows the data definition in PostgreSQL[2] SQL syntax: `serial` is auto incremented integer and `boolean` data type is used for *is_scenario* flag.

```
create table FXConsequence (
  id serial primary key,
  weight decimal,
  category varchar(10)
);

create table FXVariable (
  symbol varchar(20) primary key,
  description text,
  is_meta boolean default false,
  consequence integer references FXConsequence on delete restrict
      default null
```

[1]Generated with dbwench2: www.dbwrench.com.

[2]http://www.postgresql.org

```
);

create table InitialContext (
  symbol char(3) primary key,
  description text
);

create table MetaVariableRecord (
  meta_var varchar(20) references FXVariable on delete cascade
     not null,
  component varchar(20) references FXVariable not null
);

create table FXReason (
  id serial primary key,
  initial_context char(3) references InitialContext on delete
     cascade not null,
  antecedent varchar(20) references FXVariable on delete cascade
     not null,
  consequent varchar(20) references FXVariable,
  rank decimal,
  neg_rank decimal
);
```

Listing A.1: Data schema for $\mathbb{K}$ in PostgreSQL.

A.3.2 Database Structure for $\mathbb{K}^c$

For assessing risks and decision making, I need to first normalise the risk model at credal level. I use a second database $\mathbb{K}^c$ to store the resultant model after the consolidation process. The schema for $\mathbb{K}^c$ shown in Figure A.2 differs slightly from $\mathbb{K}$ with `FXConsolidatedReason` without the *neg_rank* field. In addition, the *consequent* field in `FXConsolidatedReason` cannot be NULL. These schema changes are the results of normalisation of the model.

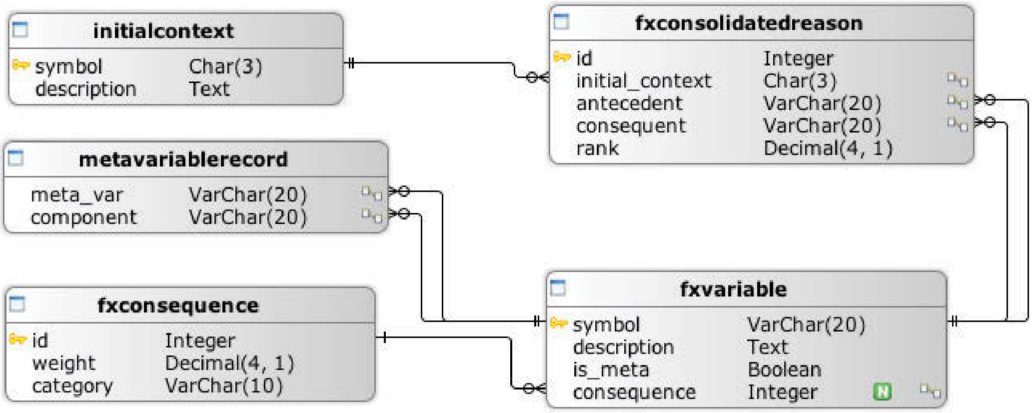


Figure A.2: Database schema for a consolidated FX risk model.

```
create table FXConsolidatedReason (
  id serial primary key,
  initial_context char(3) references InitialContext on delete
     cascade not null,
  antecedent varchar(20) references FXVariable on delete cascade
     not null,
  consequent varchar(20) references FXVariable not null,
  rank decimal
);
```

Listing A.2: Modified data schema $\mathbb{K}^c$ in PostgreSQL.

A.3.3 Populated Knowledge Database $\mathbb{K}$

I use a mixture of C/C++ and PL/pgSQL language to implement the entire TBM based risk modelling framework. The core algorithms 20 and 21 are implemented in PL/pgSQL (shown in the following section) to acquire and populate the database. The A "snapshot" of the model is shown graphically in Figure A.3 (for initial context $C1$). I use `add_variable` to insert domain factors and scenarios; use `revise_reason` to fuse and revise the quantitative epistemic reasons. Removal of variables and reasons are carried out with `remove_variable` and `remove_reason` respectively. Following are the actual data stored in a PostgreSQL database.

```
--- InitialContext
 symbol |                     description
--------+------------------------------------------------------------
 C1     | Payment in two instalments within two months and no hedging.
 C2     | Payment in two instalments within two months and fully hedged.

--- FXConsequence
 id | weight | category
----+--------+----------
  1 |    0.9 | Excellent
  2 |   -1.0 | Severe
  3 |    0.7 | Good
  4 |    0.2 | Minor

--- FXVariable
symbol |                    description                         | is_meta | consequence
-------+--------------------------------------------------------+---------+------------

 S1    | Overseas manufacturer willing to absorb the risk.       | f       | 1
 S2    | No currency hedging. Financial losses due to lowering of AUD.| f  | 2
 S3    | No currency hedging. Financial gain due to rise of AUD.  | f       | 3
 S4    | 100% currency hedging. No net losses and cost of hedging. | f      | 4
 VC    | Variation in AUD above 5%.                              | f       |
 MC    | Variation in AUD below 5%.                              | f       |
 DI    | Variation in Differential Interest between US and AU.   | f       |
 IR    | Variation in Interest Rate in AU.                       | f       |
 IF    | Variation in Inflation in AU.                           | f       |
 CA    | Variation in Current Account deficit.                   | f       |
 PD    | Variation in Public and private Debt in AU.             | f       |
 DR    | Variation in Demand in Resources.                       | f       |
 GA    | Variation in Growth rate in Asia.                       | f       |
 MW    | Variation in Manufacturer Willingness to retain orders. | f       |
 GD    | Variation in Global manufacturing Demand.               | f       |
 DC    | Europe/US Debt Crisis.                                  | f       |
 CH    | Currency Hedging costs.                                 | f       |
 HG    | Hedging in full.                                        | f       |
 GDMW  | Meta GD+MW                                              | t       |


--- FXReason
 id | initial_context | antecedent | consequent | rank | neg_rank
----+-----------------+------------+------------+------+---------

 13 | C1              | CH         |            |  1.0 |      1.0
 14 | C1              | HG         |            |  1.0 |      1.0
 23 | C1              | MC         | S1         |  0.3 |      0.0
 24 | C1              | MC         | S2         |  0.4 |      0.0
 25 | C1              | MC         | S3         |  0.3 |      0.0
  2 | C1              | MC         |            |  0.0 |      1.0
 26 | C1              | VC         | S2         |  0.8 |      0.0
 27 | C1              | VC         | S3         |  0.1 |      0.0
  1 | C1              | VC         |            |  0.1 |      1.0
 28 | C1              | MW         | S1         |  0.3 |      0.0
 10 | C1              | MW         |            |  0.7 |      1.0
 29 | C1              | DI         | VC         |  0.1 |      0.0
  3 | C1              | DI         |            |  0.9 |      1.0
```

```
30 | C1 | IF | VC   | 0.18 | 0.0
31 | C1 | IF | MC   | 0.67 | 0.0
 5 | C1 | IF |      | 0.15 | 1.0
32 | C1 | PD | IF   |  0.4 | 0.0
 7 | C1 | PD |      |  0.6 | 1.0
33 | C1 | IR | IF   |  0.5 | 0.0
34 | C1 | IR | DI   |  0.5 | 0.0
 4 | C1 | IR |      |  0.0 | 1.0
35 | C1 | CA | VC   | 0.05 | 0.0
 6 | C1 | CA |      | 0.95 | 1.0
36 | C1 | DR | VC   | 0.95 | 0.0
 8 | C1 | DR |      | 0.05 | 1.0
37 | C1 | GA | DR   |  0.9 | 0.0
 9 | C1 | GA |      |  0.1 | 1.0
38 | C1 | GD | GA   | 0.53 | 0.0
11 | C1 | GD |      | 0.47 | 1.0
39 | C1 | DC | GDMW | 0.75 | 0.0
12 | C1 | DC |      | 0.25 | 1.0

--- FXConsolidatedReason with cutoff 0.01
id  | initial_context | antecedent | consequent | rank
----+-----------------+------------+------------+-----------------------
 99 | C1 | GA | DR   | 1.00000000000000000000
 94 | C1 | IF | MC   | 0.74500000000000000000
 95 | C1 | IF | VC   | 0.25500000000000000000
 88 | C1 | MC | S3   | 0.30000000000000000000
 89 | C1 | MC | S2   | 0.40000000000000000000
 90 | C1 | MC | S1   | 0.30000000000000000000
102 | C1 | DC | GDMW | 1.00000000000000000000
 86 | C1 | VC | S3   | 0.15000000000000000000
 87 | C1 | VC | S2   | 0.85000000000000000000
100 | C1 | MW | S1   | 1.00000000000000000000
 91 | C1 | DI | VC   | 1.00000000000000000000
 92 | C1 | IR | DI   | 0.50000000000000000000
 93 | C1 | IR | IF   | 0.50000000000000000000
 98 | C1 | DR | VC   | 1.00000000000000000000
101 | C1 | GD | GA   | 1.00000000000000000000
 96 | C1 | CA | VC   | 1.00000000000000000000
 97 | C1 | PD | IF   | 1.00000000000000000000
```

Listing A.3: Data in the database $fxrisk$ in PostgreSQL.

A.4 Functional Code Snippets

This section contains a number of PLpg/SQL functions that implements the core algorithms in Chapter 6.

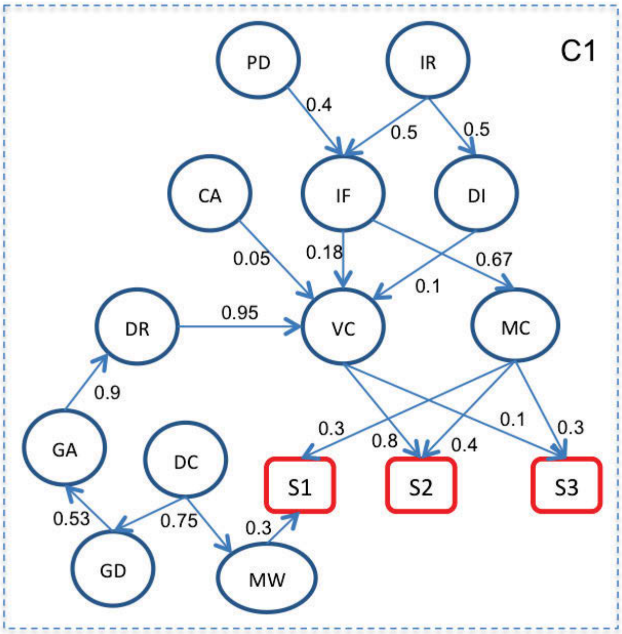


Figure A.3: A snapshot of the quantitative epistemic reasoning model for FX risks.

A.4.1 add_variable

`add_variable` creates a variable entry, either a factor or scenario in the `FXVariable` table. It also creates an entry in `FXReason` for $\Omega_{antecedent}$.

```
create or replace function add_variable(context char, sym varchar, descrip text, con
    varchar) returns boolean as $$
declare
  conq_rid FXConsequence.id%type default null;
  var_rid FXVariable.symbol%type default null;
  ic_rid InitialContext.symbol%type default null;
begin
  select symbol into var_rid from FXVariable where symbol=sym;
  if var_rid is not null then
    raise notice 'Variable % already exists', sym;
    return false;
  end if;
  if con = '' then
    insert into FXVariable(symbol,description) values(sym,descrip);
    select symbol into ic_rid from InitialContext where symbol=context;
    if not found then
      raise notice 'Unknown initial context %', context;
      return false;
    end if;
    -- I donot insert all plausible vacuous leads into RS.
```

```
    -- Instead I set m(Omega)=1.
    insert into FXReason(initial_context,antecedent,consequent,rank,neg_rank) values(
        ic_rid,sym,null,1.0,1.0);
  else
    select id into conq_rid from FXConsequence where category = con;
    if not found then
      raise notice 'No % type consequence is available.', con;
      return false;
    end if;
    insert into FXVariable(symbol,description,consequence) values(sym,descrip,
        conq_rid);
  end if;
  return true;
end;
$$ language plpgsql;
```

Listing A.4: add_variable function.

A.4.2 add_metavariable

`add_metavariable` creates a meta variable that a composite of several concrete domain variables. It creates entries both `FXVariable` (with *is_meta* set to true) and `MetaVariableRecord` table.

```
create or replace function add_metavariable(context char, sym varchar, descrip text,
    cons varchar array) returns boolean as $$
declare
  var_rid FXVariable.symbol%type default null;
begin
  select symbol into var_rid from FXVariable where symbol=sym;
  if var_rid is not null then
    raise notice 'Variable % already exists', sym;
    return false;
  end if;
  insert into FXVariable(symbol,description,is_meta) values(sym,descrip,true);
  begin
    for i in 1..array_length(cons, 1) loop
      insert into MetaVariableRecord(meta_var,component) values(sym,cons[i]);
    end loop;
  exception
    when invalid_foreign_key then
      raise notice 'Component variable % not found', cons[i];
      delete from FXVariable where symbol=sym;
      return false;
  end;
  return true;
end;
$$ language plpgsql;
```

Listing A.5: add_metavariable function.

A.4.3 remove_variable

`remove_variable` remove domain variables (including meta variables) from the database. It also remove all leads associated with the variable (using function `remove_associated_reason`). The bbms assigned to the removing leads are reassigned to the corresponding Ω leads.

```
create or replace function remove_variable(sym varchar) returns boolean as $$
declare
  meta_rec record;
begin
  -- remove associated meta variable
  for meta_rec in select * from MetaVariableRecord where component=sym loop
    execute remove_associated_reason(meta_rec.meta_var);
  end loop;
  delete from MetaVariableRecord where component=sym;

  execute remove_associated_reason(sym);
  -- delete the actual variable and its associated lead through cascade.
  delete from FXVariable where symbol=sym;
end;
$$ language plpgsql;

create or replace function remove_associated_reason(sym varchar) returns boolean as
    $$
declare
  del_rec record;
  null_rec record;
begin
  for del_rec in select * from FXReason where consequent=sym loop
    select * into null_rec from FXReason where antecedent=del_rec.antecedent and
        consequent is null;
    if found then
      update FXReason set rank=null_rec.rank+del_rec.rank where antecedent=del_rec.
          antecedent and consequent is null;
    end if;
  end loop;
  delete from FXReason where consequent=sym;
end;
$$ language plpgsql;
```

Listing A.6: remove_variable function.

A.4.4 remove_reason

`remove_reason` deletes individual epistemic reasons and reassigns their bbms to the corresponding Ω leads.

```
create or replace function remove_reason(context char,ant varchar,con varchar)
    returns boolean as $$
declare
  del_rec record;
  null_rec record;
begin
  select * into del_rec from FXReason where initial_context=context and antecedent=
      ant and consequent=con;
  if found then
    update FXReason set rank=rank+del_rec.rank where initial_context=context and
        antecedent=ant and consequent is null;
    delete from FXReason where initial_context=context and antecedent=ant and
        consequent=con;
  else
    raise notice 'Unable to find epistemic reason %->% in context %',ant,con,context;
  end if;
  return true;
end;
$$ language plpgsql;
```

Listing A.7: remove_reason function.

A.4.5 revise_reason

`revise_reason` revise and update quantitative epistemic reasons in `FXReason` table. It implements TBM rule of combination for fusing quantitative data from disparate sources. Note that, this function focuses on normal lead. It can be easily adapted to take in latent lead structure.

```
create or replace function revise_reason(context char, ant varchar, cons varchar
    array, bbms decimal array) returns boolean as $$
declare
  myrec record;
  metarec record;
  con_var record;
  mrec_c MetaVariableRecord.component%type;
  krec record;
  total_weight decimal default 0.0;
  omega_weight decimal;
  new_weight decimal;
begin
  -- check for the existance of Omega
  select * into myrec from FXReason where antecedent=ant and initial_context=context
      and consequent is null;
  if not found then
    raise notice 'Unknow antecedent variable %',ant;
    return false;
  end if;
  if array_length(cons, 1) <> array_length(bbms, 1) then
```

```
      raise notice 'size of con % is not the same as the size of bbm %',array_length(
          cons, 1), array_length(bbms, 1);
      return false;
    end if;
    for i in 1..array_length(bbms, 1) loop
      total_weight = total_weight + bbms[i];
    end loop;
    if total_weight > 1.0 then
      raise notice 'total weight of input bbmsi % is above 1.0',total_weight;
      return false;
    end if;
    omega_weight := 1.0-total_weight;
    -- implement TBM combination rule
    for i in 1..array_length(cons,1) loop
      select * into krec from FXReason where antecedent=ant and initial_context=context
          and consequent=cons[i];
      if not found then
        insert into FXReason(initial_context,antecedent,consequent,rank,neg_rank)
            values(context,ant,cons[i],bbms[i],0.0);
      else
        new_weight := bbms[i]*myrec.rank+bbms[i]*krec.rank+omega_weight*krec.rank;
        -- check for meta variables that contains current consequent cons[i]
        for metarec in select * from FXReason where antecedent=ant and initial_context=
            context and consequent<>cons[i] loop
          for mrec_c in select t.component from MetaVariableRecord t where t.meta_var=
              metarec.consequent loop
            if mrec_c = cons[i] then
              new_weight := new_weight + metarec.rank*bbms[i];
            end if;
          end loop;
        end loop;
        for j in 1..array_length(cons,1) loop
          select * into con_var from FXVariable where symbol=cons[j];
          if found and con_var.is_meta and cons[j]<>cons[i] then
            for mrec_c in select t.component from MetaVariableRecord where meta_var=
                cons[j] loop
              if mrec_c = cons[i] then
                new_weight := new_weight + krec.rank*bbms[j];
              end if;
            end loop;
          end if;
        end loop;
        update FXReason set rank = new_weight where antecedent=ant and initial_context=
            context and consequent=cons[i];
      end if;
    end loop;
    -- update Omega
    select sum(rank) into new_weight from FXReason where antecedent=ant and
        initial_context=context and consequent is not null;
    update FXReason set rank = 1.0-new_weight where antecedent=ant and initial_context=
        context and consequent is null;
    return true;
  end;
  $$ language plpgsql;
```

Listing A.8: revise_reason function.

A.4.6 consolidate_reason

`consolidate_reason` implements a version of pignistic transformation (with minor simplification) to transform the risk model $\mathbb{K}$ to a consolidated model $\mathbb{K}^c$. It populates the `FXConsolidatedReason` with normalised leads.

```
create or replace function consolidate_reason(context char, cutoff decimal) returns
    boolean as $$
declare
  lead_rec record;
  tmp_var varchar;
  tmp_count integer;
  tmp_omega decimal;
  cs decimal default 0.0;
begin
  if cutoff > 0.5 or cutoff < 0.0 then
    raise notice 'Invalide cutoff point %. Must be between [0.0-0.5]',cutoff;
    return false;
  end if;
  delete from fxconsolidatedreason;
  for lead_rec in select t.* from fxreason t join fxvariable x on t.initial_context=
      context and t.antecedent=x.symbol and t.consequent is not null loop
    -- calcualate causal strength
    if lead_rec.rank > lead_rec.neg_rank then
      cs = lead_rec.rank - lead_rec.neg_rank;
    else
      cs = 0.0;
    end if;
    if cs >= cutoff then -- add to the consolidate reason table first
      insert into FXConsolidatedReason(initial_context,antecedent,consequent,rank)
          values(context, lead_rec.antecedent, lead_rec.consequent, cs);
    end if;
  end loop;
    -- use betting frame as same as the known reasons in the consolidated database.
    -- user may have option to select different betting frame in the future
        implementation
  for tmp_var,tmp_count,tmp_omega in select t.antecedent,count(t.*),(1.0-sum(t.rank))
      from fxconsolidatedreason t where t.initial_context=context group by t.
      antecedent loop
    -- redistribution of m(Omega) to all existing reasons assuming consistent world
        and all
    -- conflicting information are due to our ignorance,i.e. treat m(phi)=0.
    tmp_omega = tmp_omega / tmp_count;
    update fxconsolidatedreason set rank=rank+tmp_omega where initial_context=context
        and antecedent=tmp_var;
  end loop;
  return true;
end;
$$ language plpgsql;
```

Listing A.9: consolidate_reason function.

Appendix B

Published Conference Papers

Wang, X. & Williams, M.-A., (2011), 'Risk, Uncertainty and Possible Worlds', *Privacy, security, risk and trust (PASSAT), 2011 IEEE Third International Conference on Social Computing*, pp. 1278-1283.

Al-Sharawneh, J.,Williams, M.-A., Wang, X. & Goldbaum, D, (2011), 'Mitigating Risk in Web-Based Social Network Service Selection: Follow the Leader', *The Sixth International Conference on Internet and Web Applications and Services*, pp. 156-164.

Wang, X. & Williams, M.-A., (2010), 'A Practical Risk Management Framework for Intelligent Information Systems', *Proceedings of The Fourteenth Pacific Asia Conference on Information Systems (PACIS)*, pp. 1866-1873.

Wang, X. & Williams, M.-A., (2010), 'A Graphical Model for Risk Analysis and Management', *Knowledge Science, Engineering and Management*, Springer Berlin / Heidelberg, vol. 6291 of *Lecture Notes in Computer Science*, pp. 256-269.

Bibliography

Abbott, R. (2009), *Managing Foreign Currency Exposures*, Tech. rep., Crane Group Limited.

Alchourron, C. E., Gardenfors, P. & Makinson, D. (1985), 'On the Logic of Theory Change: Partial Meet Contraction and Revision Functions', *The Journal of Symbolic Logic*, vol. 50, no. 2, pp. 510–530.

Andrews, J. & Dunnett, S. (2000), 'Event-tree analysis using binary decision diagrams', *Reliability, IEEE Transactions on*, vol. 49, no. 2, pp. 230–238.

Antoniou, G. & Sperschneider, V. (1993), 'Computing Extensions of Nonmonotonic Logics', *Proceedings of the Fourth Scandinavian Conference on Artificial Intelligence Electrum*, IOS Press B.V., pp. 20–30.

Antoniou, G. & Sperschneider, V. (1994), 'Operational concepts of nonmonotonic logics part 1: Default logic', *Artificial Intelligence Review*, vol. 8, pp. 3–16.

Arrow, K. J. (1950), 'A Difficulty in the Concept of Social Welfare', *The Journal of Political Economy*, vol. 58, pp. 328–346.

Aven, T. (2008), *Risk Analysis*, Wiley.

Bedford, T. & Cooke, R. (2001), *Probabilistic Risk Analysis: Foundations and Methods*, Cambridge University Press.

Belnap, J., Nuel D. (1970), 'Conditional Assertion and Restricted Quantification', *Noûs*, vol. 4, pp. 1–12.

Benferhat, S., Dubois, D., Prade, H. & Williams, M.-A. (2002), 'Practical Approach to Revising Prioritized Knowledge Bases', *Studia Logica*, vol. 70, pp. 105–130.

Benferhat, S., Dubois, D., Prade, H. & Williams, M.-A. (2010), 'A Framework for Iterated Belief Revision Using Possibilistic Counterparts to Jeffrey's Rule', *Fundamenta Informaticae*, vol. 99, pp. 147–168.

Blackburn, P., de Rijke, M. & Venema, Y. (2002), *Modal Logic*, Cambridge University Press.

Blais, R., Henry, M., Lilley, S., Pan, J., Grimes, M. & Haimes, Y. (2009), 'Risk-based methodology for assessing and managing the severity of a terrorist attack', *Systems and Information Engineering Design Symposium, 2009. SIEDS '09.*, pp. 171–176.

Bonnefon, J.-F., Neves, R. D. S., Dubois, D. & Prade, H. (2008), 'Predicting causality ascriptions from background knowledge: model and experimental validation', *International Journal of Approximate Reasoning*, vol. 48, no. 3, pp. 752–765.

Bouckaert, R. (1993), 'Probabilistic network construction using the minimum description length principle', Clarke, M., Kruse, R. & Moral, S. (eds.), *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, Springer Berlin / Heidelberg, vol. 747 of *Lecture Notes in Computer Science*, pp. 41–48, 10.1007/BFb0028180.

Boudreau, T., Davis, M., Delery, L., Korbich, J., Lambert, S., Vogel, E., Tawney, B. & Bennett, R. (2005), 'Electronic medical records: a multidimensional analysis', *Systems and Information Engineering Design Symposium, 2005 IEEE*, pp. 362–369.

Buntine, W. (1991), 'Theory refinement on Bayesian networks', *Proceedings of the seventh conference (1991) on Uncertainty in artificial intelligence*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 52–60.

Buntine, W. (1996), 'A Guide to the Literature on Learning Probabilistic Networks from Data', *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, pp. 195–210.

Charette, R. N. (1989), *Software engineering risk analysis and management*, McGraw-Hill.

Colyvan, M. (2008), 'Is probability the only coherent approach to uncertainty?', *Risk Analysis*, vol. 28, no. 3, pp. 645–652.

Cool, T. (2001), *Proper definitions for Risk and Uncertainty*, General Economics and Teaching 9902002, EconWPA.

Darwiche, A. (2009), *Modeling and reasoning with Bayesian networks*, Cambridge University Press.

Davidson, D. (1967), 'Causal Relations', *The Journal of Philosophy*, vol. LXIV, pp. 691–703.

DeGroot, M. & Schervish, M. (2002), *Probability and statistics*, Pearson education, Addison-Wesley.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977), 'Maximum Likelihood from Incomplete Data via the EM Algorithm', *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38.

Denecker, M., Marek, V. W. & Truszczyński, M. (2003), 'Uniform semantic treatment of default and autoepistemic logics', *Ar*, vol. 143, pp. 79–122.

Dubois, D. (1986), 'Belief structures, possibility theory and decomposable confidence measures on finite sets', *Computers Artificial Intelligence*, vol. 5, no. 5, pp. 403–416.

Dubois, D. (2010), 'Representation, Propagation, and Decision Issues in Risk Analysis Under Incomplete Probabilistic Information', *Risk Analysis*, vol. 30, no. 3, pp. 361–368.

Dubois, D., Berre, D. L., Prade, H. & Sabbadin, R. (1999), 'Using Possibilistic Logic for Modeling Qualitative Decision: ATMS-based Algorithms', *Fundamenta Informaticae*, vol. 37, pp. 1–30.

Dubois, D. & Guyonnet, D. (2011), 'Risk-informed decision-making in the presence of epistemic uncertainty', *International Journal of General Systems*, vol. 40, no. 2, pp. 145–167.

Dubois, D., Moral, S. & Prade, H. (1998), 'Belief Change Rules in Ordinal and Numerical Uncertainty Theories', Dubois, D. & Prade, H. (eds.), *Belief Change*, Springer Netherlands, vol. 3 of *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, pp. 311–392.

Dubois, D. & Prade, H. (1987), 'The principle of minimum specificity as a basis for evidential reasoning', Bouchon, B. & Yager, R. (eds.), *Uncertainty in Knowledge-Based Systems*, Springer Berlin / Heidelberg, vol. 286 of *Lecture Notes in Computer Science*, pp. 75–84.

Dubois, D. & Prade, H. (1988), *Possibility theory : an approach to computerized processing of uncertainty*, Plenum Press.

Dubois, D. & Prade, H. (1991), 'Epistemic entrenchment and possibilistic logic', *Artificial Intelligence*, vol. 50, no. 2, pp. 223–239.

Dubois, D. & Prade, H. (1997), 'A synthetic view of belief revision with uncertain inputs in the framework of possibility theory', *International Journal of Approximate Reasoning*, vol. 17, no. 2/3, pp. 295–324.

Dubois, D. & Prade, H. (1998), 'Possibility Theory: Qualitative and Quantitative Aspects', Gabbay, D. & Smets, P. (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic, vol. 1, pp. 169–226.

Dubois, D., Prade, H. & Sabbadin, R. (2001), 'Decision-theoretic foundations of qualitative possibility theory', *European Journal of Operational Research*, vol. 128, no. 3, pp. 459–478.

Etherington, D. W. (1987), 'Formalizing nonmonotonic reasoning systems', *Artificial Intelligence*, vol. 31, no. 1, pp. 41–85.

Fargier, H. & Sabbadin, R. (2005), 'Qualitative decision under uncertainty: back to expected utility', *Artificial Intelligence*, vol. 164, no. 1-2, pp. 245–280.

Fishburn, P. C. (1970), *Utility Theory for Decision Making*, John Wiley & Sons, Inc.

Gärdenfors, P. (1988), *Knowledge in Flux: Modelling the Dynamics of Epistemic States*, MIT Press.

Gärdenfors, P. (1992), 'Belief Revision: An Introduction', Gärdenfors, P. (ed.), *Belief Revision*, Cambridge University Press, chap. 1, pp. 1–28.

Gärdenfors, P. & Makinson, D. (1988), 'Revisions of knowledge systems using epistemic entrenchment', *TARK '88: Proceedings of the 2nd conference on Theoretical aspects of reasoning about knowledge*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 83–95.

Garvey, P. R. (2008), *Analytical Methods for Risk Management A Systems Engineering Perspective*, Chapman and Hall/CRC.

Gedik, B., Andrade, H., Wu, K.-L., Yu, P. S. & Doo, M. (2008), 'SPADE: the system s declarative stream processing engine', *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, SIGMOD '08, ACM, New York, NY, USA, pp. 1123–1134.

Gordon, M. J. C. & Melham, T. F. (eds.) (1993), *Introduction to HOL: a theorem proving environment for higher order logic*, Cambridge University Press, New York, NY, USA.

Grinstead, C. M. & Snell, J. L. (1997), *Introduction to Probability*, American Mathematical Society, chap. 8, pp. 305–324.

Grove, A. (1988), 'Two modellings for theory change', *Journal of Philosophical Logic*, vol. 17, no. 2, pp. 157–170.

Guyonnet, D., Come, B., Perrochet, P. & Parriaux, A. (1999), 'Comparing two methods for addressing uncertainty in risk assessments', *Journal of environmental engineering*, vol. 125, no. 7, pp. 660–666.

Haimes, Y. Y. (1981), 'Hierarchical Holographic Modeling', *IEEE Transactions on Systems, Man and Cybernetics*, vol. 11, no. 9, pp. 606–617.

Hansson, S. O. (1999), *A textbook of belief dynamics: theory change and database updating*, Kluwer Academic.

Helton, J. C., Johnson, J. D. & Oberkampf, W. L. (2004), 'An exploration of alternative approaches to the representation of uncertainty in model predictions', *Reliability Engineering and System Safety*, vol. 85, no. 1, pp. 39–71.

Hendershott, T. J. & Riordan, R. (2009), 'Algorithmic Trading and Information', *SSRN eLibrary*, vol. 09.

ISO 31000:2009 (2009a), 'Risk Management – Principles and Guidelines', International Standard Organisation.

ISO Guide 73 (2009b), 'Risk management – Vocabulary', International Organisation for Standardisation.

Kaplan, S. & Garrick, B. J. (1981), 'On The Quantitative Definition of Risk', *Risk Analysis*, vol. 1, no. 1, pp. 11–27.

Kelly, D. & Smith, C. (2011), *Bayesian inference for probabilistic risk assessment a practitioner's guidebook*, Springer.

Kitano, H. (1998), 'Research Program of RoboCup', *Applied Artificial Intelligence*, vol. 12, no. 2-3.

Klawonn, F. & Smets, P. (1992), 'The dynamic of belief in the transferable belief model and specialization-generalization matrices', *Proceedings of the eighth conference on Uncertainty in Artificial Intelligence*, UAI '92, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 130–137.

Kraus, S., Lehmann, D. & Magidor, M. (1990), 'Nonmonotonic reasoning, preferential models and cumulative logics', *Artificial Intelligence*, vol. 44, no. 1-2, pp. 167–207.

Lam, W. (1998), 'Bayesian Network Refinement Via Machine Learning Approach', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 240–251.

Lauritzen, S. L. (1995), 'The EM algorithm for graphical association models with missing data', *Computational Statistics and Data Analysis*, vol. 19, no. 2, pp. 191–201.

Lauritzen, S. L. & Spiegelhalter, D. J. (1988), 'Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems', *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 50, no. 2, pp. 157–224.

Lee, W. S., Grosh, D. L., Tillman, F. A. & Lie, C. H. (1985), 'Fault Tree Analysis, Methods, and Applications - A Review', *Reliability, IEEE Transactions on*, vol. R-34, no. 3, pp. 194–203.

Levi, I. (1988), 'Iteration of conditionals and the Ramsey test', *Synthese*, vol. 76, pp. 49–81.

Lewis, D. (1973), 'Counterfactuals and comparative possibility', *Journal of Philosophical Logic*, vol. 2, pp. 418–446.

Lifschitz, V. (1985), 'Computing circumscription', *Proceedings of the 9th international joint conference on Artificial intelligence - Volume 1*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 121–127.

Lifschitz, V., Morgenstern, L. & Plaisted, D. (2008), 'Knowledge Representation and Classical Logic', van Harmelen, F., Lifschitz, V. & Porter, B. (eds.), *Handbook of Knowledge Representation*, Elsevier, chap. 1, pp. 3–88.

Lindström, S. & Rabinowicz, W. (1992), 'Belief revision, epistemic conditionals and the Ramsey test', *Synthese*, vol. 91, pp. 195–237.

Lindström, S. & Rabinowicz, W. (1998), 'Conditionals and the Ramsey Test', Dubois, D. & Prade, H. (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic Publisher, vol. 3, pp. 147–188.

Liu, S. & Jiang, F. (2012), 'Based on HHM of the coal mine safety risk assessment methods', *Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE), 2012 International Conference on*, pp. 592–594.

Lukaszewicz, W. (1988), 'Considerations on default logic: an alternative approach', *Computational Intelligence*, vol. 4, no. 1, pp. 1–16.

Lyons, R. K. (2001), *The Microstructure Approach to Exchange Rates*, The MIT Press, London.

Makhoul, A., Saadi, R. & Pham, C. (2010), 'Risk Management in Intrusion Detection Applications with Wireless Video Sensor Networks', *Proceedings of Wireless Communications and Networking Conference*.

Markoff, J. (2011), 'Google Lobbies Nevada to Allow Self-Driving Cars', *The New York Times*, 10 May, Viewed on 30 May 2011, `http://www.nytimes.com/2011/05/11/science/11drive.html?_r=2&emc=eta1`.

McCarthy, J. (1980), 'Circumscription - A form of Non-monotonic Reasoning', *Artificial Intelligence*, vol. 13, no. 1-2, pp. 27–39.

McCarthy, J. & Hayes, P. J. (1969), 'Some Philosophical Problems from the Standpoint of Artificial Intelligence', *Machine Intelligence*, vol. 4, pp. 463–502.

Mediregs (2011), 'MediRegs ComplyTrack: Risk Assessment Manager', .
URL: *http://www.mediregs.com/healthcare-risk-management-software*

Moore, R. C. (1985), 'Semantical considerations on nonmonotonic logic', *Artificial Intelligence*, vol. 25, no. 1, pp. 75–94.

Murphy, K. P. (2002), 'Dynamic Bayesian Networks: Representation, Inference and Learning', Ph.D. thesis, Computer Science, University of California, Berkeley.

Nayak, A. C. (1994), 'Foundational Belief Change', *Journal of Philosophical Logic*, vol. 23, no. 5, pp. 495–533.

Nebel, B. (1998), 'How hard is it to revise a belief base?', Dubois, D. & Prade, H. (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic, vol. 3, chap. 3, pp. 77–145.

Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann Publishers Inc.

Pearl, J. (2000), *Causality: Models, Reasoning, and Inference*, Cambridge University Press.

Peppas, P. (2007), 'Belief Revision', F. van Harmelen, B. P., V. Lifschitz (ed.), *Handbook of Knowledge Representation*, Elsevier, chap. 8, pp. 371–359.

Poole, D., Mackworth, A. & Goebel, R. (1998), *Computational Intelligence*, Oxford University Press, New York.

Ramachandran, S. (1998), 'Theory Refinement of Bayesian Networks with Hidden Variables', Ph.D. thesis, The University of Texas at Austin.

Ramsey, F. P. (1931a), *General Propositions and Causality*, Kegan Paul, Trench & Trubner.

Ramsey, F. P. (1931b), 'Truth and Probability', Braithwaite, R. B. (ed.), *The Foundations of Mathematics and other Logical Essays*, Harcourt, Brace and Company, New York, pp. 156–198.

Reiter, R. (1980), 'A logic for default reasoning', *Artificial Intelligence*, vol. 13, no. 1-2, pp. 81–132.

Richards, E. P. & Rathbun, K. C. (1983), *Medical Risk Management*, Aspen Systems Corporation.

Richardson, M. & Domingos, P. (2006), 'Markov logic networks', *Machine Learning*, vol. 62, no. 1-2, pp. 107–136.

RiskMetrics (1996), *Riskmetrics: technical document*, Tech. rep., JP Morgan.

Rissanen, J. (1978), 'Modeling by shortest data description', *Automatica*, vol. 14, no. 5, pp. 465–471.

RoboCup SPL Rules (2011), 'RoboCup Standard Platform League (Nao) Rule Book', RoboCup Technical Committee.

Rott, H. (1991), 'Two methods of constructing contractions and revisions of knowledge systems', *Journal of Philosophical Logic*, vol. 20, pp. 149–173.

Rott, H. (1993), 'Belief Contraction in the Context of the General Theory of Rational Choice', *The Journal of Symbolic Logic*, vol. 58, no. 4, pp. 1426–1450.

Rubin, D. et al. (1988), 'Using the SIR algorithm to simulate posterior distributions', *Bayesian statistics*, vol. 3, pp. 395–402.

Russell, S., Binder, J., Koller, D. & Kanazawa, K. (1995), 'Local learning in probabilistic networks with hidden variables', *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 1146–1152.

Savage, L. J. (1954), *The Foundations of Statistics*, Wiley, New York.

Schaub, T. (1992), 'On constrained default theories', *Proceedings of the 10th European conference on Artificial intelligence*, ECAI '92, John Wiley & Sons, Inc., New York, NY, USA, pp. 304–308.

Schlipf, J. (1986), 'How uncomputable is general circumscription?', *In Proceedings of the Symposium on Logic in Computer Science,*, IEEE Computer Society Press, Cambridge, USA, pp. 92–95.

Seekircher, A., Laue, T. & Röfer, T. (2011), 'Entropy-based Active Vision for a Humanoid Soccer Robot', del Solar, J. R., Chown, E. & Ploeger, P. G. (eds.), *RoboCup 2010: Robot Soccer World Cup XIV*, Springer; Heidelberg, vol. 6556 of *Lecture Notes in Artificial Intelligence*, pp. 1–12.

Shachter, R. & Peot, M. (1989), 'Simulation Approaches to General Probabilistic Inference on Belief Networks', *Proceedings of the Fifth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-89)*, Elsevier Science, New York, NY, pp. 311–318.

Shafer, G. (1976), *A mathematical theory of evidence*, Princeton Univ. Press. Princeton, NJ.

Shafer, G. (1990), 'Belief functions', Shafer, G. & Pearl, J. (eds.), *Readings in Uncertain Reasoning*, Morgan Kaufmann Publishers, chap. 7, pp. 473–481.

Shenoy, P. P. & Shafer, G. (1990), 'Axioms for probability and belief-function propagation', *Uncertainty in Artificial Intelligence*, Morgan Kaufmann, pp. 169–198.

Singpurwalla, N. D. (2006), *Reliability and risk : a Bayesian perspective*, John Wiley & Sons, Inc, New York.

Smets, P. (1988), 'Belief functions versus probability functions', *Uncertainty and Intelligent Systems*, vol. 313, pp. 17–24.

Smets, P. (1990), 'Constructing the pignistic probability function in a context of uncertainty', Henrion, M., Shachter, R. D., Kanal, L. & Lemmer, J. F. (eds.), *Uncertainty in Artificial Intelligence*, North Holland, Amsterdam, vol. 5, pp. 29–40.

Smets, P. (1993), 'Belief functions: The disjunctive rule of combination and the generalized Bayesian theorem', *International Journal of Approximate Reasoning*, vol. 9, no. 1, pp. 1–35.

Smets, P. (1998), 'The Transferable Belief Model for Quantified Belief Representation', Gabbay, D. M. & Smets, P. (eds.), *Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Kluwer Academic Publisher, vol. 1, pp. 267–302.

Smets, P. (2005), 'Decision Making in the TBM: the Necessity of the Pignistic Transformation', *International Journal of Approximate Reasoning*, vol. 38, no. 3, pp. 133–147.

Smets, P. (2007), 'Analyzing the Combination of Conflicting Belief Functions', *Information Fusions*, vol. 8, no. 4, pp. 387–412.

Smets, P. & Kennes, R. (1994), 'The transferable belief model', *Artificial Intelligence*, vol. 66, no. 2, pp. 191–234.

Spirtes, P. & Glymour, C. (1991), 'An Algorithm for Fast Recovery of Sparse Causal Graphs', *Social Science Computer Review*, vol. 9, no. 1, pp. 62–72.

Spohn, W. (1983), 'Deterministic and probabilistic reasons and causes', *Erkenntnis*, vol. 19, pp. 371–396.

Spohn, W. (1988), 'Ordinal Conditional Functions: A Dynamic Theory of Epistemic States', Harper, W. & Brian, S. (eds.), *Causation in Decision, Belief Change, and Statistics*, Kluwer Academic Publisher, vol. 2, pp. 105–134.

Stalnaker, R. C. (1976), 'Possible Worlds', *Noûs*, vol. 10, no. 1, pp. 65–75.

Stephans, R. A. (2005), *Fault Tree Analysis*, John Wiley & Sons, Inc., chap. 15, pp. 169–188.

Takahashi, Y., Edazawa, K., Noma, K. & Asada, M. (2006), 'Simultaneous Learning to Acquire Competitive Behaviors in Multi-agent System Based on Modular Learning System', Bredenfeld, A., Jacoff, A., Noda, I. & Takahashi, Y. (eds.), *RoboCup 2005: Robot Soccer World Cup IX*, Springer Berlin / Heidelberg, vol. 4020 of *Lecture Notes in Computer Science*, pp. 243–254.

Tapiero, C. (2004), *Risk and Financial Management: Mathematical and Computational Methods*, John Wiley & Sons, Inc.

Twardy, C., Wright, E., Laskey, K., Levitt, T. & Leister, K. (2009), 'Rapid Initiative Assessment for Counter IED Investment', *Seventh Annual Workshop on Bayes Applications in conjunction with UAI09.*

Väänänen, J. (2001), 'Second-Order Logic and Foundations of Mathematics', *The Bulletin of Symbolic Logic*, vol. 7, no. 4, pp. 504–520.

Waring, A. & Glendon, A. I. (1998), *Managing Risk*, International Thomson Business Publishing Inc.

Watson, H. (1961), *Launch control safety study*, Tech. rep., Bell Telephone Laboratories, Murray Hill, NJ USA.

Williams, M.-A. (1994a), 'On the logic of theory base change', MacNish, C., Pearce, D. & Pereira, L. (eds.), *Logics in Artificial Intelligence*, Springer Berlin / Heidelberg, vol. 838 of *Lecture Notes in Computer Science*, pp. 86–105.

Williams, M.-A. (1994b), 'Transmutations of Knowledge Systems', J. Doyle, E. S. & Torasso, P. (eds.), *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning*, Morgan Kaufmann Publishers, pp. 619–629.

Xu, H. & Smets, P. (1994), 'Evidential Reasoning with Conditional Belief Functions', et al., D. H. (ed.), *Proceedings of Uncertainty in Artificial Intelligence*, Morgan Kaufmann, pp. 598–606.

Yager, R. R. (1987), 'On the dempster-shafer framework and new combination rules', *Information Sciences*, vol. 41, no. 2, pp. 93–137.

Yaghlane, B. B. & Mellouli, K. (2008), 'Inference in directed evidential networks based on the transferable belief model', *International Journal of Approximate Reasoning*, vol. 48, no. 2, pp. 399–418.

Zadeh, L. (1965), 'Fuzzy sets', *Information and Control*, vol. 8, no. 3, pp. 338–353.

Zadeh, L. (1978), 'Fuzzy sets as a basis for a theory of possibility', *Fuzzy Sets and Systems*, vol. 1, no. 1, pp. 3–28.

Zadeh, L. A. (1979), *On the validity of Dempster's rule of combination of evidence*, Tech. Rep. UCB/ERL M79/24, University of California, Berkely.

Zhang, N. L. & Poole, D. (1994), 'A simple approach to Bayesian network computations', *Proceedings of the 10th Canadian conference on artificial intelligence*, Morgan Kaufmann, San Francisco, CA, pp. 171–178.

Zuk, O., Margel, S. & Domany, E. (2006), 'On the Number of Samples Needed to Learn the Correct Structure of a Bayesian Network', *Proceedings of the Proceedings of the Twenty-Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-06)*, AUAI Press, Arlington, Virginia, pp. 560–567.