UNIVERSITY OF SOUTHAMPTON

Due diligence through risk analysis and belief propagation over provenance.

by

Belfrit Victor Batlajery

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

28 June,2020

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Belfrit Victor Batlajery

In many domains, the concept of due diligence is defined as taking reasonable care to protect something from unforeseen problems. The concept of due diligence is expected to be demonstrated to avoid undesired events or to anticipate (un)expected events. Moreover, demonstrating due diligence can mean complying with regulations and guidance. Thus, organisations can avoid a penalty in case something bad happens on the basis that they have taken reasonable precautions according to the applicable regulations and guidance.

The importance of due diligence and its implications in various domains motivates this research to develop a general scientific and systematic approach to demonstrate due diligence. Our claim is that scientific approaches should be utilised to develop an approach as they are usually proven to be valid and based on strong evidence. Set against this, since due diligence is contextual, it is challenging to develop a general approach that applies across multiple domains.

In developing our general approach to demonstrate due diligence, we systematically combine several scientific approaches into a framework called *prFrame*. These approaches are *Provenance*, *Risk*, and the *Probabilistic Graphical Model* (PGM), and it is these that comprise the three central pillars in constructing *prFrame*. We situate our research into the process of demonstrating due diligence in the general business product supply chain, where multiple product operators and authorities are expected to demonstrate due diligence. Indeed, our discussion with them suggests that they are continuously looking for a better approach to demonstrate due diligence.

The first pillar in *prFrame* is provenance. We consider provenance to be the most important aspect in *prFrame* since it underlies the other two pillars. Provenance is understood as a piece of historical information to explain how something is derived, who responsible for any changes to it, and what service is used to make those changes. The second pillar is risk, which can be described as a chance of undesired consequences. Finally, PGM constitutes the last pillar in *prFrame* on account of its role as the principle means of risk propagation in the provenance-based product supply chain.

To evaluate *prFrame*, we perform a set of exercises, as follows: 1) develop an ontology to conceptualise the domain of interest, 2) model the supply chain with the ontology 3) assess and infer the risk along the product supply chain, 4) establish experiments to evaluate the result of inference. A specific ontology is developed as an extension of a general ontology to model provenance, PROV-O (Ontology), so as to capture the risk along the product supply chain. The product supply chain itself is constructed based on the provenance of the product. The notion of risk that is captured comprises set risk models and risk factors, which are the main properties to assess risk quantitatively. The inference of risk is done by performing *Belief Propagation* as an inference technique based on the provenance of a product. Finally, a set of experiments is conducted to

evaluate the intuitive of the propagation result and its accuracy in the linear and non-linear product supply chain.

Our contributions in this research are therefore, 1) Development of an ontology to model the provenance of a product and map other legacy ontologies, 2) Create a risk model to overlie *Provenance Graph*s (PGs), 3) Establish a provenance-based description as a basis for Monte-Carlo simulations, 4) Develop a systematic provenance-based factorisation to allow a *Belief Propagation* technique, 5) Conduct a systematic evaluation for accuracy of states by *Belief Propagation* technique.

Authorship

I, Belfrit Victor Batlajery, declare that the thesis entitled *Due diligence through risk analysis and belief propagation over provenance*, and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

Signed:

Date: 28 June,2020

Acknowledgements

This dissertation could not have been completed without the contributions of many people over the course of the last four years of my Ph.D. journey. First and foremost, it is my pleasure to express my deep gratitude to my supervisors, Dr. Mark Weal, Dr. Adriane Chapman and Prof. Luc Moreau. Their dedication to constantly guide, encourage and lead me to my success in finishing this dissertation has been invaluable. Their advice and constructive feedback have always served to improve my academic skills until this point and will definitely benefit me further in the future.

I am very grateful to my family and friends in Indonesia for their consistent support and encouragement during my Ph.D. journey. To my beloved parents, Agustinus and Sri Eny Batlajery, who never stop praying to GOD Almighty for my strength and good health. My two younger brothers, Kurnia and Aditya Batlajery, are always my supportive siblings from Indonesia. My thanks to five children in Indonesia, Penta, Elo, Suci, Chika and Yoannes for their laughter and smiles that have eased my struggles. Further, to Sativa Koeswojo and friends in Yogyakarta for always connect me with them.

I would like to express my appreciation to my colleagues at the School of Electronics and Computer Science (ECS) at the University of Southampton for their abundant help and support for the topic of my research. Especially to Olabambo Olawasuji, Nhat Truong and Md. Mosaddek Khan. I also thank everyone in the Provenance Group for sharing ideas about their research.

I have also received support from my Indonesian and non-Indonesian friends in Southampton. To the Indonesian Student Society (Indosoc) Daryus Chandra, Nopa Dwi Mauliddiany, Rio Guntur, Agung Utomo, Masayu Dada Zuraida, Ihsan, Aya, Ipol for being my housemates for the last four years. All of you have been very helpful and supportive. To Puji Prabowo, Adinda Gladya, Jenna and Aisha, whose story has been an inspiration for me. Also to Donny Nurmayady, Heo Seong Bong and Takron Opassuwan.

In addition, I thank my Christian communities in Southampton, the Highfield Church and the KrisKat community, for always strengthening my faith. Especially, Joel Andu and Jessica Helena for their prayers that have brought peace in my mind. Finally, I thank profusely the Indonesian Endowment Fund for Education at the Ministry of Finance, the Republic of Indonesia, for funding my research at the University of Southampton. Their financial support is greatly appreciated and has made this achievement possible.

Glossary

Accuracy A degree of agreement between a measurement, or attribute, and some comparable measurement. In our research, it refers to the closeness of the inference arising from Belief Propagation to the (truest) value arising from Monte-Carlo simulation.

ANOVA test A parametric statistical technique to test whether different samples have been drawn from the same population. It is often used to test research hypotheses and is equivalent to the Kruskal-Wallis test, which is its parametric counterpart.

Belief Subjective attitudes towards a certain idea or concept. It is interchangeable with the term confidence in the sense that if we have a higher degree of belief, we are more confident that a certain event will occur as expected.

Belief Propagation A Message-Passing technique to perform an inference task under uncertainty.

Bipartite graph A graph whose vertices can be divided into two independent sets, U and V such that every edge (u, v) either connects a vertex from U to V or a vertex from V to U

Causal reasoning One of the inference patterns with probabilistic dependence that a particular event, if present, is sufficient to cause or explain another event. Also known as predictive inference or explaining away.

Conditional Probability Distribution (CPD) The probability distribution of an event occurring under the assumption that another event(s) has occurred with certainty. Its values are often represented as a Conditional Probability Table (CPT).

Consequence An effect (cost of fault) of something that occurs earlier.

Contaminated food Food with colonies of bacteria in it. In this research, this food has a state of bacteria present.

Critical Control Point (CCP) A point in the product supply chain where loss of control can lead to unacceptable safety risk.

d-separation A property of a graph to regulate how a node (variable) influences the other nodes in the graph.

Decision making The action or process of making decisions based on scientific facts or results. In the context of risk management, decision making considers the risk assessment results and other factors relevant for the health protection of consumers and the promotion of fair trade practices.

Deterministic study A study, which does not include any element of randomness in their characterisation of the process under investigation.

Directed graph A graphical representation for inference with conditional probability.

Documentation A process of preserving and retaining information either digitally or not.

Due Diligence The performance of any necessary action (e.g., assessment, monitoring, etc.) by an organisation at any time that demonstrates to regulators/auditors their understanding of their responsibilities in order to protect their consumers and avoid a penalty.

End-product testing Final testing before a product reaches a consumer.

Evidential reasoning One of the inference patterns with probabilistic dependence that assumes that observing an event should give a good indication of its cause. Also known as diagnostic inference.

Factor A function that describes the relationship between all variable nodes that connect to that factor node in a Factor Graph.

Factorisation A process of decomposing a more complex operation into its constituent less complex operations. In this research, our factorisation is based on a Generalized Distributive Law (GDL).

Factor Graph (FG) A bipartite graph consisting of a factor node for each factor function and variable node for each random variable that expresses the global function into a product of its local functions.

Food provenance A record that describes a food product and its ingredients, the processes involved in food transformation, and the organisations that are responsible for those processes from the source to consumption.

Food safety An effort (handling, storing and preparing food) throughout the entire food supply chain to minimise the risk of contamination and to protect consumers' health.

Food treatment An action of handling food by a food operator.

Frequency Table A table with frequency values of the changing states before and after a process.

Generalised Distributive Law (GDL) A generalisation of distributive arithmetic operations (i.e., multiplication and addition) to reduce the number of operations. Its simple expression can be described as $a * (b + c) = (a * b) + (a * c)$, in which it can be said that $(a * b)$ and $(a * c)$ are the factors resulting from a factorisation process of $a * (b + c)$). See **Factorisation** and **Factor**.

generated entity An prov:Entity that is generated or produced by prov:Activity.

Graph A collection of vertices (a.k.a. node or variable) joined by edges. It is interchangeable with the word network.

Graph topology The structure of the graph. It can be linear or non-linear.

HACCP plan Written documents based on the principle of HACCP.

Hazard A source of risk. In food safety, this may be a biological, chemical or physical agent in, or condition of, food with the potential to cause an adverse health effect.

Hazard Analysis Critical Control Point (HACCP) A systematic risk-based approach to identify, evaluate and control hazards, ensuring that processes are according to regulations and guidance.

Inference An assumption which one believes to be true based on existing facts or observation. Our inference technique is done by using the Belief Propagation technique, which is based on Bayes' Theorem.

Inferred variable node An unobserved node, whose values or states are in our investigation when performing an inference task.

Inter-causal reasoning An inference pattern with probabilistic dependence of mutual causes toward a common effect.

Joint Probability Distribution (JPD) A probability value of multiple events occurring at the same time. Its values can be represented in a Joint Probability Table (JPT).

Kruskal-Wallis A non-parametric statistical technique for testing whether different samples have been drawn from the same population. It is equivalent to the one-way between-groups test (ANOVA).

Linage of a product Historical information about a product.

Linear supply chain A supply chain without a branching structure.

Microbial sampling report A document about microbial content in food.

Modelling food Representation of the food ecosystem and the phenomena around it.

Modular Process Risk Model (MPRM) A process-driven framework to model statistically the transmission of undesired bacteria from one process to another process based on how food is handled in the food supply chain.

Monte-Carlo (MC) simulation An iterative computational technique to generate an approximation of certain outcomes by using randomly different inputs.

Non-linear supply chain A supply chain with a branching structure.

Observed variable node A variable node whoses value or state is known with certainty in a factor graph.

Ontology Conceptualisation referring to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon.

Posterior distribution A probability distribution after having seen the data. The opposite of posterior distribution is a prior distribution that incorporates our subjective beliefs without seeing the data.

Predictive input value (piv) A predictive numeric value that is used by by a risk model based to generate a predicted output value (pov).

Predictive output value (pov) A predictive numeric value generated by a risk model based on its associated risk factors and its numeric input.

prFrame A scientific and systematic general framework to assess risk, taking into account the aspects of provenance and the probabilistic graphical model.

Probabilistic Graphical Model (PGM) A scientific domain where probability and graphical theory are merged.

Probabilistic propagation A propagation procedure to calculate the posterior probability distribution based on observed information.

Probability distribution A function to assign how likely it is that the different possible values of the random variable will be obtained.

Process-centered provenance A modelling of provenance where a set of prov:Activity is the main focus.

Product Supply Chain An integrated process in the production and distribution of interacting with each other from source to consumer.

Provenance A record that describes the people, institutions, entities and activities involved in producing, influencing or delivering a piece of data or a thing in the world.

Provenance-based supply chain A supply chain that is constructed from the provenance of a product. An example in our research is a provenance-based product supply chain, where the product supply chain is constructed on the basis of its lineage.

Provenance Graph (PG) A directed graph from provenance records.

PROV A standardised language for exchanging provenance developed by W3C.

Quantitative Microbial Risk Assessment (QMRA) A quantitative approach to measure the risk of contamination by bacteria hazard.

Reasoning A cognitive process directed towards forming conclusions, judgements or inferences from facts or premises.

Regulation A set of rules, normally imposed by the government, that seeks to modify or determine the behaviour of firms or organisations.

Risk A function of the probability of an adverse health effect and the severity of that effect, consequential to a hazard in food.

Risk assessment A scientifically-based process consisting of the following steps: (i) hazard identification, (ii) hazard characterisation, (iii) exposure assessment, and (iv) risk characterisation.

Risk-based testing A highly effective testing technique that can be used to find and fix the most important problems as quickly as possible.

Risk factor Any aspect that triggers the development of risk.

Risk model A mathematical and/or graphical representation of the activities, which comprises a collection of elements, based on the specific circumstances which may determine the partial or total dysfunction of operations.

Sum-Product algorithm A generic message passing algorithm over a factor graph for statistical inference. It is equivalent to the Belief Propagation technique over a Bayesian Network that expresses the same factorisation with the factor graph.

Traceability The ability to trace links between source artefact and target artefact. In the food domain, it is the ability to trace and follow a food, feed, food-producing animal or substance intended to be, incorporated into a food or feed, through all stages of production, processing, and distribution.

Uncertainty A lack of perfect knowledge or limited observation of the parameter value, which may be reduced by further measurements.

Uncontaminated food Food without colonies of bacteria in it. In this research, this food has a state of bacteria absent or bacteria-free.

Unobserved variable node A variable node in a factor graph whose value or state is unknown.

used entity An prov:Entity that is used by prov:Activity in order to produce or generate another prov:Entity.

Variability A true heterogeneity of the population that is a consequence of the measurement (e.g., quantitative measurement).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The term due diligence is used in many fields such as finance, business, food and law referring to an approach to protecting something of value by taking any necessary action. This action is often seen as precautionary, conducted to identify the hazard and minimise risk. For example, conducting research before the acquisition of a company, or providing evidence that all necessary actions have been done to prevent crime from happening. In our work, we define due diligence as an organisation's performance of any necessary action at any time to show their understanding of their responsibility and to demonstrate it to regulators/auditors in order to protect their consumers and avoid a penalty. Essentially, not any action can be deemed as demonstrating due diligence, which makes it difficult to define. In this work, however, due diligence involves the continuous monitoring and risk assessment of a processes in the ecosystem or domain of the product supply chain. In a product supply chain, where a product undergoes several processes, exercising or demonstrating due diligence means that critical stages, which might lead to a negative outcome and could be anticipated, must be identified, measured to minimise risk, and controlled to ensure they happen effectively [1][2]. By demonstrating due diligence, the quality and safety of a product can be maintained; and this can serve to protect the consumers of the product and the product's operators in the product supply chain. In contrast, failing to demonstrate due diligence can lead to the emergence of undesired events, which ultimately can cause the failure of the system.

As the concept of due diligence is difficult to define, some sets of regulations have been created and continuously developed to control operators involved in the product supply chain [1]. ISO 9000, for example, is a series of quality management standards released by the International Organisation for Standardisation (ISO) to improve the internal operation of an organisation [3]. This regulation helps operators to maintain their product quality to meet consumer requirements, such as how the product should be packaged or transported. Another regulation is The Tobacco Products (Traceability and Security

Features) Regulations 2019[1] to regulate how tobacco must be treated and providing sanctions for breaches. Complying with these regulations involves documenting what processes the product has been gone through. Therefore, keeping the necessary records of past events (e.g., transactional evidence, purchase orders, etc.) can be deemed as demonstrating due diligence.

Besides controlling the quality and safety of the product, the regulations are also meant to minimise risks. Risk can be defined as a chance of danger, damage, loss, injury or any other undesired consequences [4]. In this sense, measuring risk can be seen as demonstrating due diligence because it can identify a potential hazard that then allows the necessary preventive actions to be taken. Without a risk assessment, a problem may potentially be addressed using an inappropriate approach, such as trial-and-error practice and do not necessarily target the problem. Moreover, it is difficult to establish a solid assessment to conduct preventive actions.

By definition, risk encompasses two important aspects, namely uncertainty and consequence [5][6]. The uncertainty of risk is often represented as a probability of occurrence; for example, the risk of being late to school if we use a car between 8 AM - 9 AM is 17%. Here, being late at school is a consequence we need to bear. Intuitively, the risk is dynamically changed depending on the available information when we assess it. For example, through our risk assessment, our risk of being late to school will increase to 27% when we hear about an accident on our route to school. With the same nuance, risk in one process can generate further risk, with an amplified effect in the ecosystem of the product supply chain [7]. This is often called the ripple effect of risk. Since assessing risk often depends on the information available at that time, propagating information across a supply chain is a preferable way to get the best results; since better information can improve the quality of decision making in the due diligence supporting system.

In this research, we adapt *Belief Propagation* as an inference technique through which we propagate the belief in order to infer risks across the product supply chain. In performing an inference, *Belief Propagation* propagates what we believe based on the available information as a message from one process to another; hence, the propagation algorithm is also known as a *Message Passing* algorithm. Since inferring risk is concerned with probability, it follows two basic probability rules: the summation and product rules. This technique has been extensively discussed in the *Probabilistic Graphical Model* (PGM) domain, in which probability and graph theory are merged.

In the domain of PGM, the *Belief Propagation* technique is often executed to perform an inference task over a graphical representation of a specific problem for which there is prior knowledge. We intend to improve the relevancy of inference with the notion of provenance, since provenance can record what has actually happened before (input),

[1] https://www.gov.uk/government/consultations/the-tobacco-products-traceability-and-security-features-regulations-2019

during, and after (output) a process in the ecosystem. The recorded facts captured in the provenance records can be used in two ways, to build the prior knowledge as a basis for probabilistic propagation, and to inject evidence to update the probability after by *Belief Propagation*. We are therefore in a quest to find a way to combine the *Belief Propagation* and more relevant information in our work.

We adopt the definition used by the World Wide Web Consortium (W3C) about provenance as *a record that describes the people, institutions, entities, and activities involved in producing, influencing, or delivering a piece of data or a thing in the world* [8]. The provenance of a product is captured/recorded in the form of provenance records and can explain what happened to the product during the processes in its supply chain (from the creation to the termination of the product). In other words, provenance makes something more visible for the product operators and regulators in the product supply chain. These capabilities allow due diligence to be demonstrated with confidence since every action is captured in the provenance records. Moreover, it potentially leads to improved decision making.

Although several approaches have been developed to capture the provenance of something, most are specifically domain-focused; i.e. less general [9][10][11]. Consequently, those approaches are difficult to implement in other domains and a more general approach to capture provenance is needed. Our approach uses PROV as a standardised language to capture provenance on the basis that it has three main advantages. First, it is a standardised and interoperable language. Second, it is more abstract (general) and expandable to the more specific domains. Third, it can be represented as a graphical model, which allows some analytical techniques to operate on it.

Our approach to combine the *Belief Propagation* and provenance is by capturing the provenance of a product with a PROV Model, which ultimately models the product supply chain in a graph representation called a *Provenance Graph* (PG). This graph captures all necessary information for risk assessment since it holds the information of the processes a product has gone through. With that information attached to the PG, risk can be calculated and propagated by executing *Belief Propagation* as an inference technique.

The use of *Belief Propagation* is often applied in a *Factor Graph* (FG) as a graphical representation. A *factor graph* is a bipartite graph[2] that expresses the global function into a product of local functions [12]. The product supply chain based PG is therefore converted into an FG to allow a *Sum-Product* algorithm performs an inference efficiently. The efficiency of calculation is achieved by rearranging the basic probability rules (*sum* and *product*) [13]. This concept (*Belief Propagation* over an FG) underlies our approach to propagate beliefs so as to infer risk across a supply chain.

[2] A bipartite graph is a graph whose vertices can be divided into two independent sets, U and V such that every edge (u, v) either connects a vertex from U to V or a vertex from V to U

All the processes involved in modelling a product with the PROV Model, and calculating and propagating the risk over the graph are done through our framework *prFrame*. *prFrame* is a framework to estimate a product's risk, which allows for observations to be taken into account, as well as estimates to be inferred for the unobserved part of the supply chain described by provenance records. In detail, *prFrame* comprises several processes such as modelling and integration of provenance and risk, provenance-based simulation, conversion from PG to FG, and risk propagation with the *Belief Propagation* technique.

1.1 Motivation

As described before, all operators in the product supply chain are forced by regulations to inspect, check or monitor (precautionary actions) their supply chain as a way to demonstrate due diligence. Here, we are interested in the distribution of a product in its supply chain, which represents a general overview of several processes that the product has gone through. Each process has its characteristics and parameters. This means that some processes naturally entail more risk than others, and a failure in one process can increase risk in the following processes in a ripple effect. Hence, it is important to understand what processes a product has gone through in order to be able to assess whether the product is at high risk or not.

In this research, we present a systematic approach to assess risk as a means to exercise due diligence in the product supply chain. Our approach relies on the concept of provenance to help operators and regulators understand a product supply chain. We argue that capturing the provenance of a product can help to identify the potential hazards during its processes and make sure that a product has undergone appropriate processes, since the recorded provenance can provide more relevant information for risk assessment. This information is used in a risk calculation and the results are propagated across the provenance-based supply chain. In addition, the recorded provenance of a product can lead to improved accountability and trust in decision making.

Our final aim is to provide a mechanism that allows the regulatory authorities to understand how a product has been processed, and assess the risk associated with that process. Against this background, our motivation is to develop a proper concept of due diligence and analytical methods to support that concept. These analytical methods are ultimately to be implemented in a system that supports the concept of due diligence. With the aim of creating such a system, we develop the general framework called *prFrame*. *prFrame* aims to provide a systematic approach to capture provenance thereby supporting reasonable decision making. In general, capturing provenance aims to monitor the processes in the ecosystem so that reasonable decisions can be achieved by applying an inference technique with *Belief Propagation*.

1.2 Thesis Statement and Research Questions

In order to develop a systematic mechanism to support due diligence that allows us to understand risk contextually, we define our thesis statement as:

Provenance-based risk model and belief propagation technique demonstrating due diligence as a systematic approach.

Our systematic approach comprises a set of methodical procedures that encompass several aspects. Those aspects are the integration of provenance and risk, construction of a product supply chain based on its provenance and risk, and performance of a provenance-based inference technique. Each aspect is thoroughly investigated and evaluated so that it can become a supportive pillar of our final systematic framework, called *prFrame*. With this statement, we develop our research questions as follows:

Research Question 1. *How can we model the provenance of a product to support its regulation and its risk assessment?* We attempt to answer this research question by using a formal and descriptive specification of a conceptualisation, called an ontology. In this context, we produce our ontology to cover the concept of the provenance of a product, including its regulation and risk.

Research Question 2. *How can we overlay a product supply chain, based on its provenance, with the existing risk models?* This research question is a complement to the first research question. Here, the standardised provenance model, PROV, captures the necessary information regarding the lineage of product, its related regulations, and its potential risk along the product supply chain. Ultimately, the outcome of this research question is a provenance-based supply chain.

Research Question 3. *How can Belief Propagation be augmented with provenance?* Our approach to addressing this research question is to perform a powerful inference technique, called *Belief Propagation* over the provenance-based graph. The provenance of a product consists of a set of processes from the beginning to the end, visualised as a PG. Factorisation of a probability distribution is therefore needed to convert the PG into an FG which is how *Belief Propagation* is generally performed.

Research Question 4. *Is the approach relevant to the common product supply chain?* Our final research question addresses the relevance of our framework as a systematic mechanism to assess risk. We perform a set of experiments and evaluations in a specific use case based on real-world supply chains. Finally, we discuss the result to see whether our framework is intuitively relevant to the real-world risk assessment or not.

1.3 Contribution

The final outcome of this thesis is the systematic mechanism of risk assessment through *Belief Propagation* over the provenance-based supply chain. To achieve that, we combine the concept of Provenance, with Risk and PGM. While the concept of provenance underlies the lineage of a product supply chain, the notion of risk and PGM underlie our approach to assess the risk and propagate it across the PG by performing the *Belief Propagation* technique. All of those concepts are encapsulated into our framework, *prFrame*, and the contributions it provides are as follows:

(i) Development of an ontology to model the provenance of a product and map other legacy ontologies.

(ii) Integration of risk and provenance in the *Provenance Graph* to allow performance of an inference technique.

(iii) Provenance-based description as a basis of a Monte-Carlo simulation.

(iv) A systematic provenance-based factorisation to allow the performance of a *Belief Propagation* technique.

(v) A systematic measurement for accuracy of states by *Belief Propagation* technique.

Among these contributions some have been peer-reviewed in the following publications:
1.) B. V. Batlajery, M. Weal, A. Chapman, and L. Moreau, "prfood: Ontology principles for provenance and risk in the food domain," in 2018 IEEE 12th International Conference on Semantic Computing (ICSC). IEEE, 2018, pp. 17–24.
2.) B. V. Batlajery, M. Weal, A. Chapman, and L. Moreau, "Belief propagation through provenance graphs," in International Provenance and Annotation Workshop. Springer, 2018, pp. 145–157.

1.4 Thesis structure

Following this chapter, Chapter 2 presents our **Background and related work**. We start by introducing the notion of due diligence and some examples of exercising it, in particular in the area of the product supply chain. We then introduce the concept of provenance and how this concept is able to model the lineage of a product and the risk across its supply chain. Next, we present our investigation of risk and how to calculate and measure it with a risk model and identified risk factors. Finally, we show our underlying approach to propagate the risk by performing *Belief Propagation* over the provenance-based supply chain.

Chapter 3 presents the **Research Methodology** of this research. In this chapter, we explain the general aspects that we need to cover in order to develop and evaluate our general framework, *prFrame*. In addition, we also describe the way we demonstrate and evaluate the framework.

In Chapter 4, we present the ***prFrame* pipeline** that supports us in developing a systematic framework to assess risk over the provenance-based supply chain. We divide this chapter into several sub-chapters, with each contributing to our final framework, *prFrame*. *prFrame* begins by overlaying risk and the provenance-based supply chain. The Monte-Carlo simulation is the next step before the conversion from the PG to the FG takes place. Our pipeline ends by performing *Belief Propagation* as an inference technique on the converted FG.

Our case study to demonstrate *prFrame* is presented in Chapter 5, **Food case study:*prFood***. Here, we choose food as a domain of interest for demonstrating *prFrame*. Based on the nature of the domain, we introduce our ontology, *prFood*, as an extension ontology of PROV-O (Ontology) to model and integrate the provenance of food with the risk models associated in the food supply chain. We also introduce a quantitative approach to calculating the risk of the Modular Process Risk Model (MPRM). Finally, we show how the risk of contamination is propagated by using *Belief Propagation* technique.

The case study of food contamination in Chapter 5 is actioned in Chapter 6 through our systematic **Experimental evaluation of *prFrame* as a case study**. First, we establish our method to measure the accuracy of inference by *Belief Propagation*, as well as the experimental setup needed to establish some parameters from the risk models and their associated risk factors. Then, several experiments are presented that consider the linear and non-linear food supply chains. Each experiment is designed to represent the actual food supply chains and a set of hypothesis is also presented. In the end, the results for all experiments are presented and ready to be discussed.

The results in Chapter 6 are discussed and evaluated in Chapter 7, **Discussion and evaluation**. This Chapter discusses the phenomena found in each of our experiments based on the resulting accuracies. We also show that those results follow our intuition about the risk of contamination, which indicates that *prFrame* can potentially be a systematic approach for assessing risk and supporting the concept of due diligence.

The **Conclusions and Future works** is our next chapter, Chapter 8. We summarise this thesis by reflecting on our thesis statement in Chapter 1. Moreover, we state our answer to each of the research questions. Finally, we close the thesis by recommending some future works. Those works are presented based on our thorough investigation of *prFrame* to enable to become a better framework when those works are implemented.

Chapter 2

Background and Related Work

2.1 Due diligence in a supply chain

While this term is not formally defined in law, the term due diligence is usually understood to include the process of identifying all critical stages or processes, then identifying adequate control measures to prevent the risk and putting in place appropriate management control procedures to ensure the control measures are followed through effectively [1][2]. According to the Merriam-Webster Dictionary, this term has been used since the fifteenth century [1]. Nowadays, it is used to define a set of actions necessary to protect something and is used in many domains, such as in finance, law, business, sport, human rights and health. These necessary actions are sometimes seen as precautionary steps to prevent undesirable incidents or events.

These precautionary actions are deemed important and need to be exercised, especially when the production and distribution of business products spans several places or they undergo several processes. Here, the integrated manufacturing and distributing processes to transform raw materials into a final product from the source to consumers can be defined as a product supply chain [14][15]. In principle, any operator involved in a product supply chain must handle the product appropriately so as to maintain its quality and safety until it reaches its consumers. Demonstrating due diligence can protect operators in the product supply chain from penalties and losses, both direct and indirect (e.g. loss of reputation).

To assist operators in demonstrating that they are exercising due diligence, a set of regulations are created. We identify two purposes of implementing the regulations in the product supply chain. First, to encourage all operators in the product supply chain to operate safely and reduce any potential hazard [1][16] in order to ensure the quality and safety of the product. Second, to enable the traceability and track-ability of the

[1] https://www.merriam-webster.com/dictionary/due%20diligence

product across its supply chain [17][18] as a way to protect both the operators and consumers in the event of an undesirable occurrence. If operators can achieve these two purposes it can help them to attain an overview of their product supply chain and monitor any potential risks within it.

Another way of demonstrating due diligence is product testing to identify the potential hazards. The process of testing for hazards attempts to verify compliance with standard product safety requirements and to ensure the safety of the product before it reaches consumers. This type of testing is often performed at the end of the production process to test the quality of the final product; hence, the name end-product testing. End-product testing is time, resource and cost intensive, however, and does not provide the necessary controls to cope with hazards because the processes to produce the final product will only be investigated if the results of the testing are negative. In other words, this approach is less preventive as it relies solely on the final product. Many operators and legislators, therefore, have turned to risk-based testing, such as HACCP (Hazard Analysis Critical Control Point) [19].

HACCP is a systematic approach to establishing good production, sanitation and manufacturing practices that ensure a safe product [20]. Although it is heavily used in the food industry, HACCP can be implemented in various domains. It has seven basic principles, which are listed as follows: 1.) List all potential hazards, 2.) Determine Critical Control Points (CCP), 3.) Establish critical limits for each CCP, 4.) Establish a monitoring system for each CCP, 5.) Establish corrective actions, 6.) Establish verification procedures, 7.) Establish documentation. These principles are implemented into the system to help operators identify and eliminate potential hazards; ultimately, minimising risk along with the product's production and distribution. Figure 2.1 shows an example of how the seven principles of HACCP can be implemented.

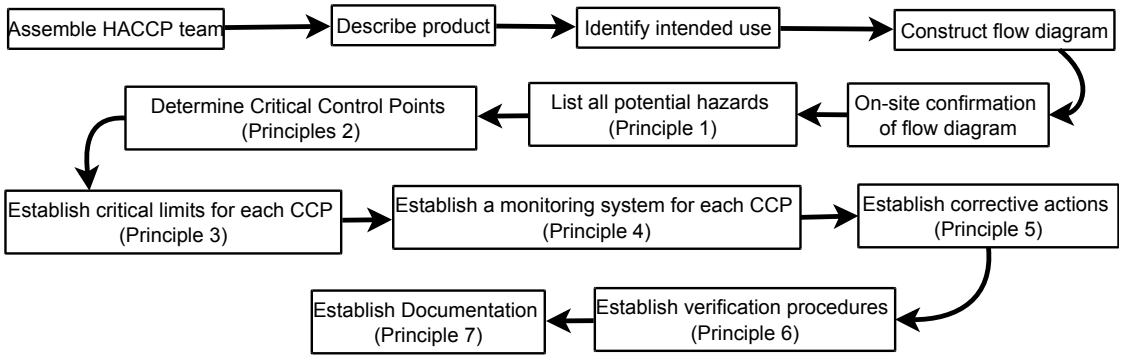


FIGURE 2.1: The logic sequence for application of HACCP.

Those steps help the product operators reflect on how they handle their product, including hazard identification, assessment and control. HACCP is performed by focusing on CCP, or the points in the product supply chain where a loss of control can lead to unacceptable safety risk. Each identified CCP is measured by using specified indicators to control the risk from the production to the final product. Although HACCP

facilitates the improvement of product safety, the management of an organisation has the discretion to determine what the final quality of the product should be. Although the government can set goals for reducing the risks that are associated with a product, therefore, it depends on each operator to decide on the risk they are willing to take in terms of government penalties [2].

As those organisations are inspecting, checking or monitoring (precautionary actions) their supply chain, they also need to retain the necessary information about the actions they have performed. This documentation can be used as evidence for demonstrating due diligence [2][19]. Providing evidence that shows how operators treat their product is crucial in helping them avoid a penalty in case of a system failure or other undesirable events. Ultimately, these documented actions and processes throughout the product lifetime depict the history of a product and are known as a product provenance [21][22]. In the next section, we introduce the notion of provenance as our approach to capturing the history of the product in order to construct the product supply chain.

2.2 Provenance

In the creation of a thing (e.g., book, wine, meat, etc.), the processes involved in production and distribution are the important factors to understand its origins. People often struggle to find a product's ultimate source, however, because of changes or modifications during the process of creation until its final form. Since many processes and people are involved through the journey of creation, it becomes difficult to identify who is responsible for what and which process accounts for which product. For example, a text that was written by one person may be changed by someone else through editing (adding or subtracting some words). Since the text is continuously changing, it becomes more difficult to keep track of the changes. In that sense, the concept of provenance was introduced to be able to explain what has changed, who is responsible for those changes, what services are used to process the changes, and how a final result is derived [23][24][25].

Meanwhile, the World Wide Web Consortium (W3C) defines provenance as *a record that describes the people, institutions, entities, and activities involved in producing, influencing, or delivering a piece of data or a thing in the world* [8]. It contains the description of the data and the processes involved during the data lifetime, such as how something is derived, who is responsible for a certain action, what the consequences of doing such an activity are, what the risks emerging during data production are, etc. With this capability, provenance is often seen as the historical evidence of an incident [21][22], and can be a piece of crucial information to determine whether the information should be trusted or not [24][26], and whether the result after a computation process is relevant or not. This feature of provenance helps a system or application towards system accountability,

which can benefit the operators in the product supply chain. Finally, the provenance records support reasoning and inference based on the recorded historical information, which can help inform a conclusion or find the cause of incident.

In terms of product traceability, a good provenance should capture information in such a way as to support interoperability and communication among the many operators involved in the product supply chain. The provenance records should be able to explain the origin of a product by providing a description of what has happened to a product from its creation to its current state. In this sense, a product derivation is the main focus to model the product lineage in the provenance records. Thus, examining the lineage of the product will help us construct the product supply chain based on a set of processes that change the properties of a product. In other words, the notion of a product's lineage is crucial to model the provenance-based product supply chain. In the next section, we introduce our approach to capture provenance with a standardised PROV.

2.2.1 Provenance Data Model and Ontology

Several authors have researched the task of capturing the provenance of something. Bhagwat et al. and Agrawal et al. propose methods to capture the provenance in a database [9][10]. The approach by Bhagwat et al. provides an annotation management system to understand the lineage of data in relational databases. Meanwhile, Agrawal et al. develop a database management system that captures data and the information of data linage, called Trio. In addition, a work by Sar and Cao also presents an approach to capture provenance in the file system [11]. Their approach is to capture provenance by tracking the execution of a file, its command lines and input. Finally, the Flexible Image Transport (FITS) format used in astronomy [27] and the Spatial Data Transfer Standard (SDTS) [28] are other attempts to capture provenance by including the provenance information in each file. Consequently, these approaches are not general enough to be applied in other domains.

Since those approaches are specifically domain-focused, it is difficult to use them in other domains and a more abstract approach to capture provenance is needed. In this research, we adopt a standardised language for exchanging provenance developed by W3C, called PROV. This provenance standardised language is intended to facilitate a machine-processable data model for provenance [29]. PROV provides three main benefits for capturing the provenance of something. First, it is a standardised and interoperable language. Second, it is more abstract (general) and expandable to the specific domain. Third, it can be represented as a graph, called a *Provenance Graph* (PG), which allows some analytical techniques to operate over it.

In order to capture the provenance of something, PROV relies upon three basic concepts: Entity (prov:Entity), Activity (prov:Activity) and Agent (prov:Agent). Figure 2.2 depicts the core concept of PROV and Table 2.1 describes these. In addition, there are several other concepts to express the relationship between the core concepts. Some of these relationships can be seen in Figure 2.2 and are described by W3C in Table 2.2.

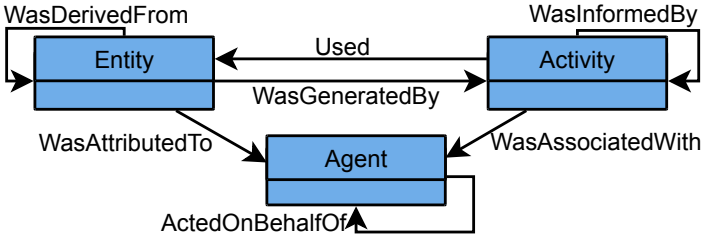


FIGURE 2.2: The core concepts of PROV.[1]

Concept	Definition
Entity	A physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.
Activity	Something that occurs over a period of time and acts upon or with entities; it may include consuming, processing, transforming, modifying, relocating, using or generating entities.
Agent	Something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity.

TABLE 2.1: The definition of entity, activity and agent in PROV.[1]

Concept	Notation	Definition
Derivation	wasDerivedFrom	Transformation of an entity into another, or expiry of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity.
Generation	wasGeneratedBy	The completion of production of a new entity by an activity. This entity did not exist before generation and becomes available for use after this generation.
Usage	used	The beginning of utilising an entity by an activity. The activity using some entities generated by the other activities.
Communication	wasInformedBy	The exchange of some unspecified entity by two activities, one activity using some entity generated by the other.
Delegation	actOnBehalfOf	The assignment of authority and responsibility to an agent (by itself or by another agent) to carry out a specific activity as a delegate or representative, while the agent it acts on behalf of retains some responsibility for the outcome of the delegated work.
Attribution	wasAttributedTo	The ascribing of an entity to an agent.
Association	wasAssociatedWith	An assignment of responsibility to an agent for an activity, indicating that the agent had a role in the activity. It further allows for a plan to be specified. This is the plan intended by the agent to achieve some goals in the context of this activity.

TABLE 2.2: The relations in PROV[1].

PROV consists of several building blocks that are defined to describe provenance. Those building blocks are in the form of document specifications and complement each other as shown in Figure 2.3. A complete PROV specification is presented in `https://www.w3.org/TR/prov-overview/`.

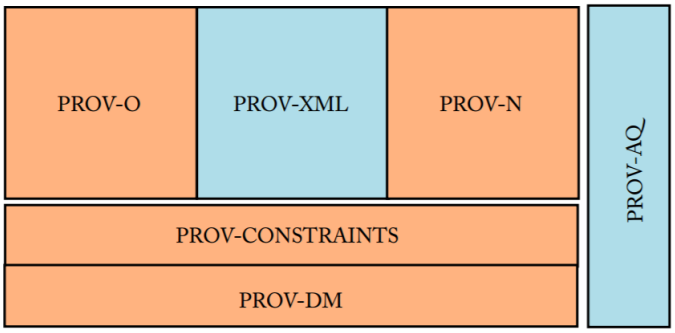


FIGURE 2.3: A simplified PROV specification [30]; orange denotes the recommendations, blue denotes the notes.[3]

PROV-DM (PROV Data Model)[4] is a provenance conceptual data model (standard interchange format) that allows provenance information to be interchanged between systems. This data model can be serialised into several different formats, including PROV-O, PROV-XML and PROV-N. PROV-O (PROV Ontology) is a Web Ontology Language (OWL2) document that provides classes, properties and restrictions in a provenance document generated by a provenance-aware application (i.e. an application with a capability to monitor and capture the provenance information.)[5]. While PROV-XML,[6] meanwhile, is a common language format for data interchange among systems. PROV-N[7] is a serialisation of the provenance data model meant for human consumption. Provenance is said to be valid when it follows the logical order that allows reasoning and analysis [31]. In order to validate a provenance document, PROV-CONSTRAINTS[8] is introduced. Finally, a provenance document needs to be stored somewhere and should be located, retrieved and queried. Thus, PROV-AQ[9] provides a mechanism for accessing and querying provenance documents. These documents are bundled in a set of specifications allowing provenance to be modelled, serialised, exchanged, accessed, merged, translated and reasoned over [32].

To capture and structure the detailed information of the past incident in the product life journey, PROV-DM uses PROV-O. PROV-O is a lightweight ontology that specifies the classes, properties and restrictions in respect to the information that is modelled with PROV-DM. For example, an animal is a class and a fish is an individual of that class. The object properties are used to describe the relations between classes or individuals, and datatype properties are used to record the data values that belong to classes or individuals. For example, modelling a five-year-old fish can be done by specifying a fish as an individual with age as its object properties and five years old as its datatype

[3] A completed specification is available in `https://www.w3.org/TR/prov-overview/`. Accessed: 28 June,2020.
[4] `http://www.w3.org/TR/2013/REC-prov-dm-20130430/`. Accessed: 28 June,2020
[5] `http://www.w3.org/TR/2013/REC-prov-o-20130430/`. Accessed: 28 June,2020
[6] `http://www.w3.org/TR/2013/NOTE-prov-xml-20130430/`. Accessed: 28 June,2020
[7] `http://www.w3.org/TR/2013/REC-prov-n-20130430/`. Accessed: 28 June,2020
[8] `http://www.w3.org/TR/2013/REC-prov-constraints-20130430/`. Accessed: 28 June,2020
[9] `http://www.w3.org/TR/2013/NOTE-prov-aq-20130430/`. Accessed: 28 June,2020

properties. As an ontology, PROV-O provides basic knowledge about provenance that can represent, exchange and integrate into different provenance-aware applications.

As described above, an ontology is one of the building blocks in the PROV defined in PROV-O. Although PROV is suitable for modelling the retrospective history of an entity [33], it is domain agnostic. Consequently, many ontologies extend PROV-O for their specific purposes. In their work, for example, Ali and Moreau develop an ontology, cProv, that allows traceability in cloud-based services at the service/platform level. Another work by Garijo also extends the use of PROV-O for documenting workflow plans in terms of steps and variables, known as P-PLAN [34]. In social computation, Packer et al. develop a provenance ontology with the aim of measuring a community-based service reputation [35] and Markovic et al. develop SC-PROV as a provenance vocabulary for social computation [33].

2.2.2 Provenance Graphs

PROV as a standardised provenance model was chosen in this research for its capability to represent a provenance of something as a graph, known as a *Provenance Graph* (PG). A PG is derived by translating a serialised format of provenance (PROV-O, PROV-XML and PROV-N) into a graphical representation. As explained, PROV has three core models (prov:Entity, prov:Activity, and prov:Agent), and they are represented as the graphical items shown in Figure 2.4.

FIGURE 2.4: The three core concepts in the provenance graph.

Figure 2.4 shows how the Entity, Activity and Agent in Figure 2.2 are depicted in a PG. The relations between those three concepts are depicted as an arrow *from* and/or *to* each of the core concepts, although they still follow the same directions and explanation as shown in Figure 2.2 and Table 2.2. To illustrate the visualisation of a provenance record into a PG, Listing 2.1 shows the snippet of the PROV-N document and Figure 2.5 shows its PG. The complete PROV-N of this supply chain is available in the Appendix.

```
...
activity(mixing_b,-,-,[prov:type = "prFood:mixing" %% xsd:string])

entity(boiled_spaghetti,[prov:type = "prFood:boiled_spaghetti" %% xsd:string,
prFood:stageDetails = "boiled_spaghetti" %% xsd:string])
entity(cooked_sauce_meat,[prov:type = "prFood:cooked_sauce_meat" %% xsd:string,
prFood:stageDetails = "cooked_sauce_meat" %% xsd:string])
entity(served_spaghetti,[prov:type = "prFood:served_spaghetti" %% xsd:string,
prFood:stageDetails = "served_spaghetti" %% xsd:string])

wasGeneratedBy(boiled_spaghetti,boiling,-)
wasGeneratedBy(cooked_sauce_meat,mixing_a,-)
wasGeneratedBy(served_spaghetti,mixing_b,-)

used(mixing_b,cooked_sauce_meat,-)
used(mixing_b,boiled_spaghetti,-)

wasDerivedFrom(served_spaghetti, cooked_sauce_meat)
wasDerivedFrom(served_spaghetti, boiled_spaghetti)
...
```

Listing 2.1: A snippet of a provenance record in a spaghetti supply chain.

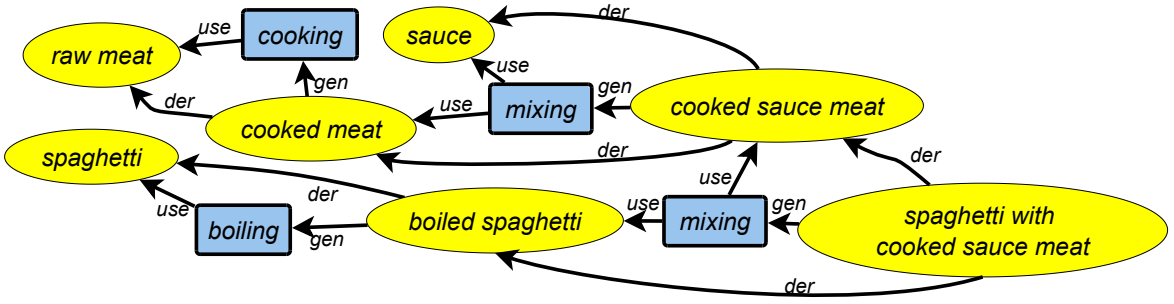


Figure 2.5: A provenance graph of spaghetti with cooked sauce meat.

Figure 2.5 shows the simple *step-by-step* of serving a plate of spaghetti with cooked meat sauce. Due to limited space, we abbreviate the relations between prov:Entity and prov:Activity into *use*, *gen*, and *der* as prov:used, prov:wasGeneratedBy, and prov:wasDerivedFrom consecutively. The supply chain was started from a slice of raw meat that was cooked (through the process of *cooking*) and mixed (through the process of *mixing*) with sauce into cooked sauce meat. Meanwhile, the spaghetti was boiled and mixed with the cooked sauce meat. The result of the last *mixing* process was a generated final spaghetti with cooked sauce meat. This modelling ultimately can be seen as a supply chain of the spaghetti. The lineage of a product can therefore be modelled with PROV-DM, which intuitively represents the product supply chain.

Essentially, all the information presented in a PG is based on product's provenance documents. The values needed for further analysis based on the PG must therefore be captured/recorded in the provenance documents. These values are annotated through the properties or attributes of the three core concepts of PROV. For example, if the

cooking process in Figure 2.5 has a range of temperatures between 70 to 90 degrees Celsius, then we would model the *cooking* process as the prov:Activity with attribute temperature as in Figure 2.6.

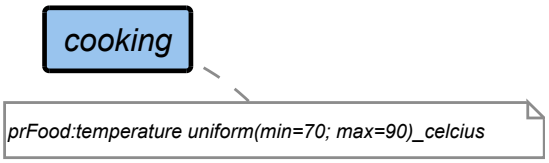


FIGURE 2.6: The temperature in the cooking process in prov:Activity.

Updating the value of the temperature should be done by modifying the provenance document and regenerating its PG. The same principle is also applied when adding another core concept. The new core concepts and their relations will be merged together with the existing core concepts. In addition, extracting historical information from the PG must be done by querying the provenance records. PROV-AQ provides the mechanism to query the provenance records.

When a recorded provenance of a product can be presented as a *provenance graph*, the derivation of something can be seen clearly, as shown in Figure 2.5. Since the connections of derivation are clearly depicted in the graph, one can easily investigate any anomalies in the product supply chain. When the recorded provenance gets richer, however, the PG becomes more complicated and it is then difficult to investigate an anomaly manually with the naked eye. This problem in the PG is addressed by developing an automatic analytical method that can be applied to the PG.

2.2.3 Provenance in various domains

Provenance has been developed and used in several domains and can be implemented in several ways. One of these has been explained in Section 2.2.1, where PROV-O is extended to capture more detail information on the more specific domain of interest. This approach has brought some provenance-aware applications, which we briefly mention in this section.

One example of a provenance-aware system is CollabMap,[10] where provenance is utilised to develop a trust mechanism that filters out incorrect information [36][37]. The aim of CollabMap is to control and measure the quality of data generated by unknown participants (crowd-produced data). In this application, a graph representation of provenance was analysed in order to recognise patterns and use these to assess the quality of crowd-sourced data. Similarly, HAC-ER[11] also uses crowdsourcing data generation and extracts situational awareness information to ensure that aid is better allocated when a

[10] `http://www.collabmap.org/`. Accessed: 25 July 2017.
[11] Watch its video here: `https://vimeo.com/119525848`. Accessed: 28 June,2020.

catastrophe happens [38]. HAC-ER tries to track and verify public reports in order to produce current actions whenever new information comes or old information becomes invalid. Another application that utilises provenance is SmartShare.[12] This application measures a subject's reputation in an online community by computing feedback about it (e.g., rating, comment, etc.) [35]. Since provenance can represent the past processes of something, its use can promote accountability of the system.

Provenance has also been used in e-social science to improve the Evidence Based Policy Assessment (EBPA) by providing the context of the data [39]. In this sense, the context could be who is involved in capturing the data, the assumptions underlying the data collection, the reasoning in concluding the result, how the data is spread for further use, etc. Overall, all processes involved in EBPA can be captured with the concept of provenance. Similarly within the food context, the details of food, including its supply chain, can also be captured in a standardised provenance format [40][41][42]. Batlajery et al. combine the notion of provenance and probabilistic propagation to assess risk. Their work holistically assesses the risk of contamination across the food supply chain from the food production process to the distribution to the end consumers. Markovic et al., in their work, model food provenance by extending provenance ontologies to provide an alert system if regulations are abused [40]. Their work also extended other existing ontologies to model HACCP-based food preparation within the kitchen in order to comply with food safety monitoring systems. This safety monitoring system, essentially, enhances the use of the Internet of Things (IoT) in the food domain. In general, the use of provenance in many domains aims to increase transparency [43][44][45].

Besides increasing transparency, provenance also enables the exploration of the motivation behind certain activities and provides a reasoning for decision making in the social computing arena [45]. Finally, Baillie et al. also utilise provenance to assess the quality of the dataset [46]. They provide QUAL-DM (Data Model) with QUAL-O, which is compatible with PROV-O, in order to represent a quality matrix for data quality assessment. The quality of data also has been researched in linked data and open data ecosystems [47][48][49]. In particular, provenance has been proven to support the validation of open data and to improve public accountability and services.

2.3 Risk

In this section, we review the basic concepts about risk. Risk can be defined as a chance of danger, damage, loss, injury or any other undesired consequences [4]. A similar definition has also been mentioned by several authors [5][6][50][51]. Risk is triggered by a source of risk, also known as a hazard, or anything that causes harm (e.g., bacteria or

[12] `http://groups.inf.ed.ac.uk/smart-society/`. Accessed: 28 June,2020.

chemical substances) [6]. In defining risk, it is agreed that it involves two elements, the consequences (impacts) and associated uncertainty (probability of occurrence).

Consequence can be seen as an effect (cost of fault) of something that occurs earlier. For example, delay in an online banking transaction may have been caused by an application failure. Cost can affect how the stakeholders deal with risk in practice because they usually prioritize the most critical harm in the system first when the they have a budget constraint [52]. Uncertainty in risk, meanwhile, is derived from how risk is often interpreted in the form of a probability distribution [5][6][53]. This probability distribution illustrates the chance of a fault being present, and can be measured or assessed. In our example about online banking transactions, we may think that there is a 50% chance of an application failure and the remaining 50% from other factors (e.g., network issues, invalid certificates, insufficient funds, etc.). Both of those aspects can be mathematically formalized with Equation 2.1

$$RE(f) \; = \; P(f) \; x \; C(f) \tag{2.1}$$

Equation 2.1 presents the formula to calculate a Risk Exposure as a function f. It is a multiplication of Probability that an unexpected event will occur and the Consequence or Cost that emerges if the event actually occur. Based on the above definition and equation, it is understandable to analyze potential risk and reduce the amount of risk in order to avoid any undesirable event [52].

In general, there is a three-stage process to be followed when analysing the risk, namely Risk Assessment, Risk Management and Risk Communication [54][55]. Risk assessment can be understood as a scientific evaluation of the risk, where any potential hazard is identified and measured. The assessment can be quantitative (numerical expression) or qualitative (qualitative expression). It comprises tools and techniques (approaches) to organize in formations in order to better understand the interaction between several factors or variables that contributes to the potential risk. Qualitative risk assessment results in a descriptive estimation and is mostly derived from a range of interviews, questionnaires, observations, focus groups, etc. In contrast, quantitative risk assessment is based on predefined formulas and mathematical expressions to produce numeric estimations [56]. After risk assessment, risk management is performed to weigh alternatives in order to either accept or reduce the assessed risk. Later, the best alternatives need to be implemented. Last but not least, risk communication is used in bridging risk assessors, risk managers, and other interested parties to exchange information in regard to risk [57]. The relation of those three-stage process is depicted in Figure 2.7.

In this research, we focus on risk assessment only, in particular, a quantitative risk assessment. A quantitative method works with numerical data and can be classified as a deterministic or stochastic study. Deterministic studies do not include any element of

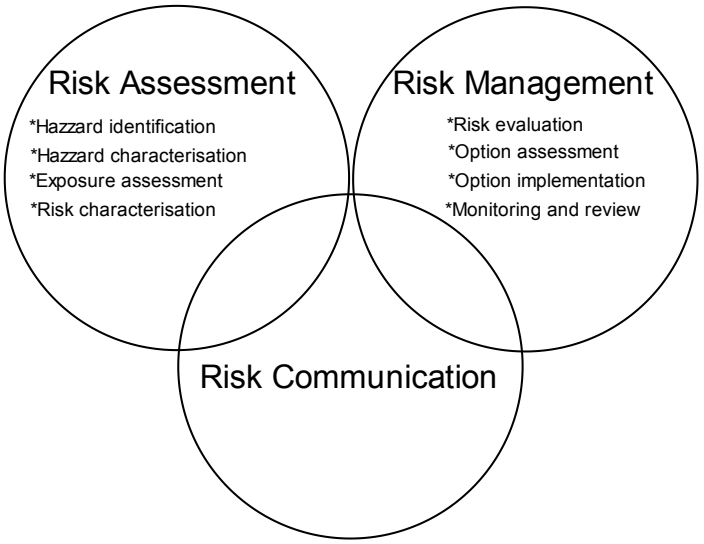


FIGURE 2.7: Three-stage process of analysing risk.

randomness in their characterisation of the process. They use a single point estimation (usually mean or worst-case-scenario) for calculation and present results as a single number (usually including confidence intervals). This approach does not give much insight into the drivers of the risk, however [58]. Moreover, a calculation with the worst-case-scenario will produce an extreme output without regard to the minimum value, which may also produce an extreme output. Using the mean as a point estimation, however, will produce the average risk, but ignore the extreme values (minimum and maximum), which may be infrequent but have severe consequences [59]. On the other hand, stochastic studies include randomness in the form of probability distributions to give a better representation of the natural processes that are inherent to real practice [60].

Using the quantitative risk assessment over qualitative risk assessment has some advantages. First, quantitative risk assessment is executed by simulating multiple scenarios or procedures. This will provide in-depth understanding of risk with various factors that contribute to it. Second, the cause and effect of risk can be easily traced during the simulation, which help in identifying potential causes of a certain event. Third, qualitative risk assessment can be considered as an integrated approach between multiple disciplines; hence, it provides a common understanding in communication between the stakeholders of different disciplines. Finally, it helps identifying only the dominants accidents scenarios to avoid wasting resources [61].

In both approaches to risk assessments, qualitative or quantitative, there are two important aspect: a risk model and a risk factor. A risk model can be understood as a tool to assess the risk that takes into account the probability and consequence of an undesired event [62]. The assessment is done by looking at the experience in the past and incorporating current circumstances and variables as well as a set of assumptions. There are 2

concepts when performing a quantitative risk analysis, validity and reliability. Validity can be seen as how success the risk analysis describes the concept it attempts to describe and Reliability can be seen as the consistency in producing the result [61].

As our focus is quantitative, the risk model is usually in the form of mathematical formulae that require some inputs to produce a numerical output. These inputs are known as risk factors, which can be defined as anything that contributes to the changing of risk and further be propagated through mathematical models. In other words, anything that triggers the development of risk can be seen as a risk factor. In principle, the mathematical models that are used to propagate risk are the risk model, which is a simplified version of reality. As a consequence, this simplification may hold some drawbacks as followed:

- As a simplification of the real-world events, the integrity and reliability of the results by a risk model is often weak. This can be dangerous to be implemented without knowing the model's limitations.
- A model will always as a limitation even with the similar type of product. This can potentially lead to a wrong conclusion or decision.
- The data that used in the model can be inconsistent and can lead to the wrong conclusion, even if the model is correct.

To our knowledge, several risk models have been developed to inform about the risk held in a process. In general, those are developed with the aim of understanding the development of risk in order to help a decision maker take an appropriate decision. The Value-At-Risk (VAR) financial risk model provides a mechanism to compute the loss [63] and Rougas et al. extend this into the CyberVAR (Cyber Value-At-Risk) risk model to model cyber threats [64]. In the supply chain domain, the Food and Agricultural Organization and the World Health Organization (FAO/WHO) together present a risk assessment to deal with contamination with Salmonella in chicken and chicken eggs [65]. Meanwhile, Faisal et al. explore the risk in supply chains [51] and Alhomidi and Reed propose a model of the risk in a network security context [56].

2.4 Probabilistic Graphical Model (PGM)

In the actual world, we often see the occurrence of an event with uncertainty. This uncertainty can be seen as the effects of multiple aspects through our limited observation [66][67]. For example, we are not sure yet whether or not it will be a rainy day on Wednesday. On Monday, when the weather forecast informs us that it will be raining on Wednesday, our uncertainty decreases as we are more certain now about Wednesday's

weather. When we observe that the sky is blue on Tuesday, however, we become uncertain again about the weather on Wednesday. Thus, exploring an interesting event under uncertainty takes into account different possibilities, which provides the chance that all possible outcomes may occur.

In many domains (e.g., food, finance, health, etc.), studying or investigating an event under uncertainty is often conducted by modelling that event graphically. The main reason for this is that such modelling is fast and cheap, yet able to visualise the interaction between aspects that possibly contribute to the occurrence of an event. This interaction is also known as a dependency between nodes in the graph. Those concepts are a part of our discussion about the *Probabilistic Graphical Model* (PGM). With PGM, an actual event is translated (visualised) as a graph with edges and nodes.

Graphs have been widely defined by several authors as a collection of nodes/vertices that can be connected to each other by means of edges [66][68][69]. The nodes in the graph represent the random variables, while the edges intuitively and naturally represent the relations between the nodes they connect to, and those relations hold uncertainty in the form of a probability distribution [70][71][72][73]. In the following sections, we start our discussion with the notion of probability and move on to the concept of graphical representation with a probability distribution.

2.4.1 Probability Theory

In general, probability can be understood as an approach to measure the uncertainty of the occurrence of an event. It refers to the degree of confidence that an event will occur [74]. For example, the probability $\mathrm{P}(X)$ of an event X quantifies the degree of confidence that X will occur. With $\mathrm{P}(X) = 1$, we are certain that one of the outcomes in X occurs and $\mathrm{P}(X) = 0$ indicates that all outcomes in X are impossible. Other probability values between 0 and 1 represent options that lie between these two extreme values. Since human observation is always insufficient, however, and since unexplained events sometimes occur, it is not preferred to represent the probability of an occurrence with 0 (impossibility) or 1 (always occurs).

In the probability of $\mathrm{P}(X)$, X is said to be a random variable that, due to random chance, will have variation in its outcomes or values $\{x_1, x_2, x_3, \ldots, x_k\}$ [75]. For example, the colour of the cars that pass a gate is a random variable named "colour", and the values it takes can be {*red, green, black, etc.*}. Since the variable is random, its values are expected to be different as more experiments are performed and more samples are taken. These values are usually represented in the form of a probability distribution.

2.4.1.1 Probability Distribution

A probability distribution is a function to assign how likely the different possible values of the random variable are to be obtained [76]. There are two different sets of values a random variable can take, discrete and continuous. While the discrete values take a finite set of values, such as a set of numbers {*1, 2, 3, 4, 5, 6*}, the continuous values take any value from a continuum, such as any real number in the interval [0,1].

For a discrete random variable X, the form of its probability distribution function is equal to each of its possible values. For instance, in a six-sided die, each side would have a probability of 1/6. In this context, we refer to a probability distribution function as a *Probability Mass Function* (PMF). For a function P(X) to be a valid PMF, P(X) must be non-negative for each possible value in X. Moreover, a random variable must take some values in the set of possible values with a probability summing to one; so, we require that P(X) must sum to one [77].

In the case of continuous random variables, a random variable can take any value from a continuum, such as the set of all real numbers of an interval. For a continuous random variable X, we cannot form its probability distribution function by assigning a probability that X is exactly equal to each value. The probability distribution function for the continuous random variable is therefore called a *Probability Density Function* (PDF), which assigns the probability that X is near each value [78][79].

Depending on the random variable (discrete or continuous), there are several well-known probability distributions to describe the dataset. Some distributions that describe discrete random variables are binomial, discrete uniform, geometric, negative binomial and hypergeometric. In contrast, normal, uniform, triangular, logistic, exponential and log-normal distributions are used to describe continuous random variables. Some of the probability distributions with their parameters used in this research are presented in the Appendix.

2.4.1.2 Sum and product rules

Some formal functions and rules should be obeyed in probability theory. Generally, probability can be expressed in two fundamental rules, namely the *sum rule* and the *product rule*, as shown in Equation 2.2. Most probabilistic inference and manipulation problems are solved by these two equations and become the basic calculation for the *Sum-Product* algorithm in probabilistic propagation.

$$(a)\ \textbf{sum rule} \quad P(X) = \sum_{Y} P(X,Y) \qquad (b)\ \textbf{product rule} \quad P(X,Y) = P(Y|X)P(X) \tag{2.2}$$

In Equation 2.2, $P(X)$ is referred to as a marginal distribution applied to the distribution of random variable X and is simply verbalised as the probability distribution of X. It is obtained by summing all possible values of other random variables (sum rule), according to the law of total probability. In other words, it is a distribution of a random variable regardless of the value of other random variables.

In many cases, the questions often involve the values of several random variables or a joint probability distribution (JPD), written as $P(X, Y)$. This distribution describes all random variables in its set (e.g., random variables X and Y). For example, we might be interested in the event that a student has high intelligence and gets "A" for his/her grade ($Intelligence = high$ and $Grade = A$). To discuss such an event, we need to consider the joint distribution across these two random variables ($Intelligence$ and $Grade$). The joint distribution of two random variables has to be consistent with the marginal distribution, in that $P(X)=\sum_{Y} P(X, Y)$, since, by definition, the marginal is obtained by summing the joint distribution across all variables except X.

Similarly, a conditional probability distribution (CPD) can be verbalised as the probability of Y given X or $P(Y|X)$ that specifies the belief in Y under the assumption that X is known (observed) with certainty [75]. For example, $P(Intelligence|Grade = A)$ is used to denote the conditional distribution across the events described by $Intelligence$ given the knowledge that the student's grade is "A". Note that the conditional distribution over a random variable, given an observation of the value of another one is not the same as the marginal distribution.

When the CPD deals with discrete-valued random variables, we can resort to its values in a tabular representation of CPDs. The conditional probability of P$(Y|X)$ can be encoded as a table that contains an entry for each joint assignment to X and Y. This table represents every possible discrete CPD and is known as a Conditional Probability Table (CPT) [66]. We will discuss more about CPT in Section 2.4.2 since this table is an essential property of a *Bayesian Network* (BN).

Inferring from a model is the same as finding the CPD over some variables. Predicting values for a new data point is basically trying to find the conditional probability of the unknown variable, given the observed values of other variables [66][76]. In other words, after learning that an event X is true, we need to know how this situation changes the probability of another event occurring that is conditional on event X. The answer lies in the notion of conditional probability. In the next section, we present the notion of a directed graph as a graphical representation for inference with conditional probability.

2.4.2 Graphical Theory

The role of a graph in probability is important since it provides a vivid representation of the sets of variables that are relevant to each other in any given state of knowledge

[73]. In graphical representation, that relevancy is represented as an edge to represent the relationship between the nodes. In this research, we focus more on the concept of a directed graph since its edges represent a direct influence between nodes, called a *Bayesian Network* (BN).

In the BN, the nodes represent random variables and the edges represent dependencies [75]. Since the directed edges represent the direct influence between nodes, there are several structures of connections that affect how one node influences the others. Note that the direction of the edge does not restrict the flow of influence across the BN. Figure 2.8 shows three basic dependency structures of BN.

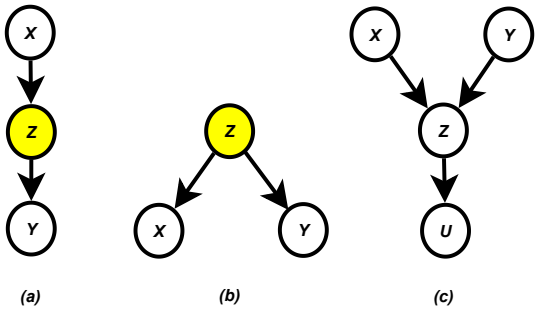


FIGURE 2.8: Directed graphs with arrows to indicate the direction of influence.

Figure 2.8 shows that the direction of the arrow holds some influence from the source (parent) to the destination (child) in three types of structural connections. These are serial, diverging and converging connections [80]. From the graph in Figure 2.8(a), we can see that X and Y are conditionally independent given Z. In other words, X and Y are independent if Z is observed. The same rule also applies in a diverging connection, shown in Figure 2.8(b), where X and Y are independent when Z is observed. In contrast, the opposite rule is applied in converging connection, Figure 2.8(c), as X and Y are conditionally dependent given Z (X and Y are dependent if Z is observed). These types of structures are known as *d-separation*, which is a property of a graph which regulates how a node influences the other nodes in the BN.

From those basic connections, we derive three types of reasoning [81]. First, is causal reasoning, which occurs when the observation of the parent changes the probability of its child. The underlying idea is that the introduction of the observed parent will strengthen our belief in its children. Moreover, the introduction of the parent not only changes the probability of its children but can go further downstream in the graph. For example, observing X in Figure 2.8(a), will increase our belief in Z and Y. The opposite of causal reasoning is evidential reasoning. This occurs in the situation where, when the child is observed, our belief in its parents is also changed. This reasoning brings the idea that knowing the outcomes (children), will change our belief regarding their causes (parents). Using the same graph as in Figure 2.8(a), we can see that observing Y will change our belief in X and Z. The last type of reasoning is inter-causal reasoning. This type of reasoning occurs when two different causes have a common effect. An example

of inter-causal reasoning is the converging connection, whereby knowing Z in Figure 2.8(c), changes our belief in X and Y.

BN relies on the CPDs, in the form of CPTs, to perform an inference task. Based on the structures in Figure 2.8, Figure 2.9 shows the constructed CPTs for each graph in Figure 2.8. As an illustration, we use two states, i.e., *Yes* or *No*, as the states in each node for all graphs.

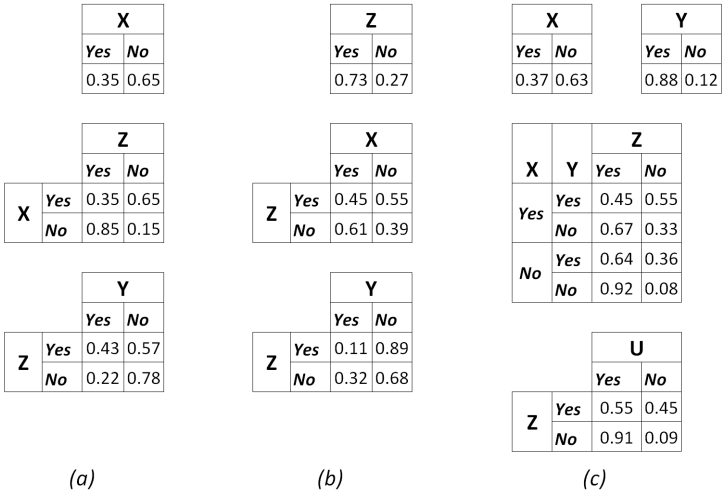


FIGURE 2.9: The CPTs of each node in the Bayesian Networks in Figure 2.8

Figure 2.9 shows the CPTs in all nodes from the BNs in Figure 2.8. The size of the CPT in each node depends on the number of states (Sta) and the number of nodes (Nod), with the formula Sta^{Nod}. For example, the nodes without parents have the size of the CPT of the node's state only (2^1). Thus, the more the states and/or neighbours the node has, the bigger the size of the CPT that node has. For example, node Z in Figure 2.8(c) shows the size of 2^3 as the CPT has three nodes (X, Y and Z) and two states (*Yes* and *No*).

In Figure 2.9, the numbers in each cell are the arbitrary numbers representing the conditional probability distribution between nodes in the BNs in Figure 2.8. Note that, every row in each table sums to 1, representing the probability of a node with a certain state, given the state(s) of its parent(s). In general, this describes how much influence the node with a particular state has on its neighbours. The construction of the CPT is a factorisation of the joint distribution between all nodes, based on the conditional independence in the BN with the function in the Equation 2.3. Equation 2.3 is a product of the probability functions in each node, conditioning on its parent(s). Here, $x_{pa(i)}$ is a set of parents of node x_i. Thus, $p(x_i|x_{pa(i)})$ implies the parent-child conditional distribution.

$$p(x_1, ..., x_x) = \prod_{i=1} p\left(x_i \mid x_{\mathrm{pa}(i)}\right) \tag{2.3}$$

The use of BN as a causal model has been integrated with provenance to assess the trust about data (i.e. to compute the belief in derived data) [82]. Thus, they propose a model that utilises provenance records that convey how the information is propagated and how accurate the source is. The model uses BN-based provenance, which implies a causality through an efficient factorisation with CPT. In principle, the BN corresponds to a compact factorisation of the joint probability distribution and this notion can be presented as a factor in another graph representation, called a *Factor Graph* (FG).

2.4.3 Factor Graph

A *Factor Graph* (FG) is a bipartite graph consisting of two types of nodes (a variable node and a factor node), where each node only has neighbours of the opposite type [66][79][83]. A bipartite graph is a graph whose nodes can be divided into two independent sets, U and V, such that every edge (u, v) either connects a node from U to V or a node from V to U. In an FG, a variable node is depicted as a circle and represents every random variable in the distribution. Moreover, a factor node is depicted as a square and represents each factor $f(x)$ in the joint distribution.

A FG expresses the global function into a product of local functions [12]. This local function is known as a factor in a factor node, which describes the relation between all variable nodes that connect to that factor node. Consider Equation 2.4, the global function is g, and f_a, f_b, f_c, and f_d are said to be the factors which the connected variable nodes depend on. This factorisation can be expressed as the FG shown in Figure 2.10.

$$g(x1, x2, x3) = f_a(x1, x2) f_b(x1, x2) f_c(x2, x3) f_d(x3) \tag{2.4}$$

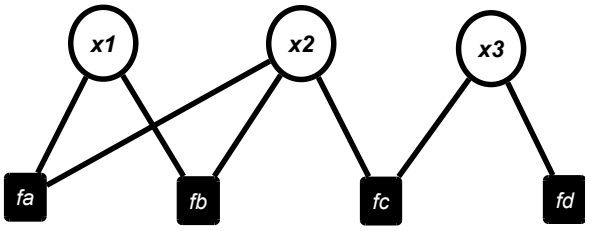


FIGURE 2.10: An example of a factor graph with factorisation in Equation 2.4.

The main advantage of an FG over other forms of graphical representation is that it allows us to be more explicit about the details of the factorisation [66][83]. Note that from Figure 2.10, there are two factors $f_a(x_1, x_2)$ and $f_b(x_1, x_2)$ that are defined over the same set of variables. In an undirected graph, the product of two such factors would simply be lumped together into the same clique[13]. Similarly, $f_c(x_2, x_3)$ and $f_d(x_3)$ could be combined into a single potential over x_2 and x_3. The FG, however, keeps such

[13] A clique is a maximal subset of the vertices in an undirected network such that every member of the set is connected by an edge to every other [71].

factors explicit, so it is able to convey more detailed information about the underlying factorisation.

Because a factor is essentially a function, we can inherit an FG from BN. When converting a BN to an FG, we simply create variable nodes in the FG corresponding to the nodes of the BN and then create factor nodes corresponding to the conditional distributions or CPTs. Finally, we add the appropriate links that connect the variable nodes and the factor nodes. Note that there can be multiple structures of FGs, all of which correspond to the same BN [79], as illustrated in Figure 2.11.

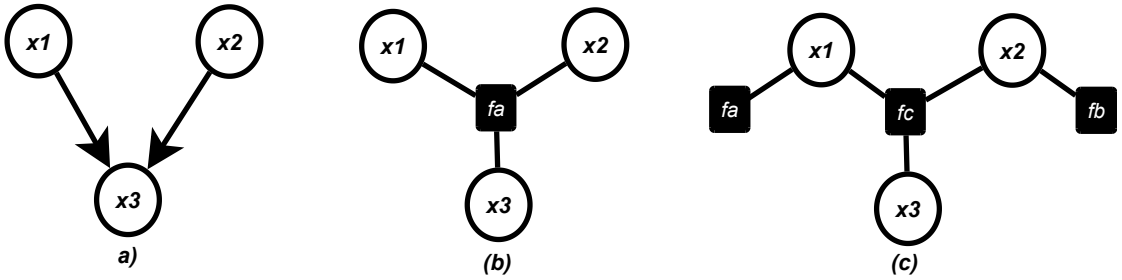


FIGURE 2.11: Conversion from a directed graph to a factor graph.

Figure 2.11(a) is a BN with the factorisation $p(x1)p(x2)p(x3|x1, x2)$. Figure 2.11(b) represents the same distribution as Figure 2.11(a), whose factor can be expressed as $f(x1, x2, x3) = p(x1)p(x2)p(x3|x1, x2)$. Figure 2.11(c) also represents the same distribution with Figure 2.11(a), but with different factorisation $fa(x_1) = p(x_1)$, $fb(x_2) = p(x_2)$ and $fc(x_1, x_2, x_3) = p(x_3|x_1, x_2)$. Since multiple different FGs can represent the same directed or undirected graph, this allows the FGs to be more specific about the connection [79]. To illustrate the conversion, Figure 2.12 shows the converted FGs from the BNs in Figure 2.8.

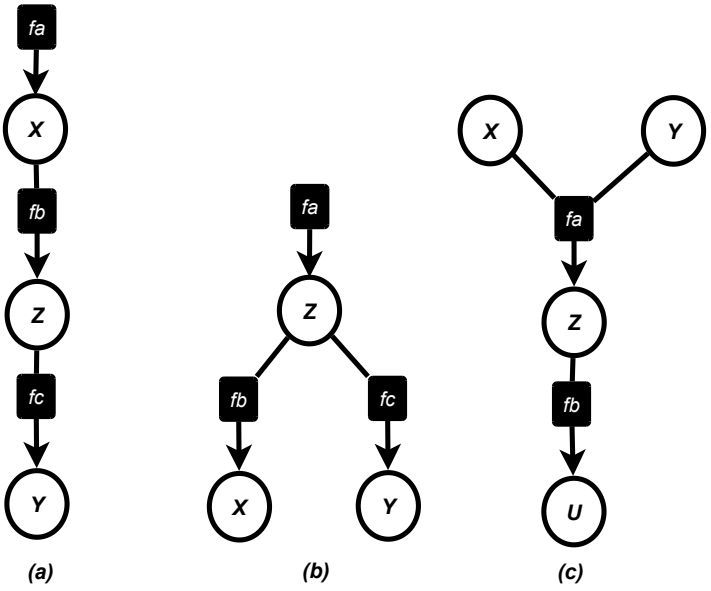


FIGURE 2.12: Directed graph with arrows to indicate the direction of influence.

In Figure 2.12(a), each factor node can be mapped to the CPDs in Figure 2.9(a). They are $f(a) = P(X)$, $f(b) = P(Z|X)$ and $f(c) = P(Y|Z)$, similar to Figure 2.9(b) and 2.9(c). Note that, the edges from factor nodes to variable nodes are directed edges to imply the conditional independence between the parents and the children derived from the BN. In other words, the directed arrow from a factor node to a variable node indicates the influence from a factor to a variable node.

With the BN, one can manipulate the model by using various techniques to perform inference [66]. In this context, Kschischang et al. initially proposed the use of the FG for the *Belief Propagation* technique [83]. In the next section, we present the notion of *Belief Propagation* as an inference technique to propagate the probability distribution between nodes.

2.4.4 Belief Propagation

An inference in graphical models often involves observing some nodes and computing the posterior distribution[14]. When we enter evidence and use it to update our belief about the probability, it is often referred to as propagation and its computation can be done with the *Belief Propagation* technique that was first coined by Pearl in his work in 1982 [81]. It was originally derived for exact inference in a tree-structured graph but it has now been proved to work with graphs containing cycles (loops) [66]. This works by using the conditional independence relationships in a graph to perform efficient inference.

The principle of *Belief Propagation* is to exchange information (e.g., messages) between nodes. This allows the nodes to communicate their local state by sending messages over the edges [66][79][83]. By local, we mean that a given node updates the outgoing messages on the basis of incoming ones from the previous iterations. This is the main characteristic of *Message Passing* algorithms, which usually take an FG representation as input and update it recursively through local computations to calculate marginal distribution over the variable nodes in the FG. Some variances of the *Message Passing* algorithm are described in the Generalised Distributive Law (GDL), including *Sum-Product* and *Max-Sum* algorithms, to perform an inference task effectively [71].

In general, the messages are passed around and get updated until a stable belief state is reached (convergence). Every time the messages are passed and get updated, it counts as one iteration. In theory, the number of iterations should be less than or equal to the diameter of the graph [83]. Once the convergence is reached, the calculation to determine the marginal probabilities of all the variables is started. Depending on the type of graph, however, some may not reach convergence due to circular reasoning. This typically happens when the graph contains a cycle in it. Although convergence is

[14]Posterior distribution is a probability distribution after having seen the data. The opposite of posterior distribution is a prior distribution that incorporates our subjective beliefs without seeing the data.

not guaranteed, *Belief Propagation* has been found also to have outstanding empirical success in loopy graphs [66].

According to *Standard Message Passing* (SMP) protocol,[15] leaf nodes (i.e., nodes with only one neighbour) initiate the process by sending all their parents an identity message (synchronously). The main reason is that all leaves have their locally initiated values already; therefore, they can send the message immediately. Unlike the SMP protocol, however, nodes may also be initialised randomly, although they can then update their outgoing messages at any time and in any sequence (asynchronously). In this case, the messages are still guaranteed to converge, and the marginal values can be calculated as the solution. The asynchronous version can exchange more messages than the synchronous version before reaching the convergence state, however, meaning that it experiences more communication costs than the synchronous case.

One of the variants of the *Belief Propagation* algorithm is called the *Sum-Product* algorithm, which is an exact inference algorithm in tree-structured graphs [79]. It relies on an iterative *Message Passing* algorithm derived from the Bayesian procedure that computes the marginal distribution at every node simultaneously to explore the conditional independence. Specifically, each variable leaf node, x_i, sends a message $Q_{i\to j}(x_i) = 1$, and each function leaf node f_j sends a message $R_{j\to i}(x_i) = F_j(x_j)$. Following the SMP protocol, each time a node receives a message from an edge, it computes outgoing messages based on Equation 2.5 and Equation 2.6, then sends the messages to all remaining edges. When a variable node has received all messages from its neighbours, it can start calculating the exact marginal value according to Equation 2.7. Here, M_i is the set of functions connected to x_i, and N_j represents the set of variable indices, indicating which variable nodes are connected to function node F_j.

message from variable node to factor node

$$Q_{i\to j}(x_i) = \prod_{k\in M_i\setminus j} R_{k\to i}(x_i) \tag{2.5}$$

message from factor node to variable node

$$R_{j\to i}(x_i) = \sum_{x_j/i}[F_j(x_j) \sum_{k\in N_j\setminus i} Q_{k\to j}(x_i)] \tag{2.6}$$

Marginalise

$$Z_i(x_i) = \sum_{j\in M_i} R_{j\to i}(x_i) \tag{2.7}$$

To understand how the calculation of the *Sum-Product* algorithm works, consider Figure 2.13 as an example of an FG. This graph has four variable nodes and three factor nodes, which capture all possible combinations the factor node may have from its connected variable nodes. Table 2.3 shows all iterations in the *Sum-Product* algorithm.

[15] Within a graphical representation of a DCOP (Distributed Constraint Optimisation Problems), a message can be sent from a node v on an edge e if, and only if, all messages received at v are on edges other than e.

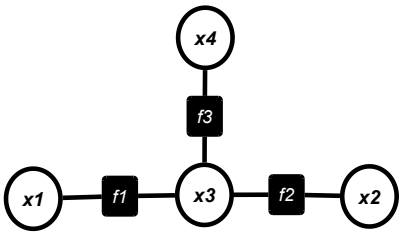


FIGURE 2.13: An example of a factor graph.

Iteration	**Message**			**Message type**
1	$Q_{x1\to f1}$	$Q_{x2\to f2}$	$Q_{x4\to f3}$	Variable node → Factor node
2	$R_{f1\to x3}$	$R_{f2\to x3}$	$R_{f3\to x3}$	Factor node → Variable node
3	$Q_{x3\to f3}$	$Q_{x3\to f1}$	$Q_{x3\to f2}$	Variable node → Factor node
4	$R_{f3\to x4}$	$R_{f1\to x1}$	$R_{f2\to x2}$	Factor node → Variable node
5	$x_1 = R_{f1\to x1}$; $x_2 = R_{f2\to x2}$; $x_3 = R_{f1\to x3} * R_{f2\to x3} * R_{f3\to x3}$; $x_4 = R_{f3\to x4}$			At Completion (When each variable has received messages from all of its neighbours)

TABLE 2.3: The iterations in the *Sum-Product* algorithm.

As depicted in Table 2.3, the initial messages (iteration 1) are sent from the leaves according to the sum-product protocol. Since all the leaves are variable nodes, the value of each message is 1 according to $Q_{i\to j}(x_i) = 1$. In the second iteration, all the messages are the messages from the factor node to the variable node; thus, Equation 2.6 is used. After the second iteration, the variable x_3 has received all messages from f_1, f_2, f_3. Thus, it can send a message back to all of the factor nodes in the third iteration. At this iteration, variable x_3 is able to calculate its final value because it has received all its incoming messages ($R_{f1\to x3}$, $R_{f2\to x3}$, and $R_{f3\to x3}$). The final message is calculated by multiplying all the incoming messages on the basis of the sum-product's protocol ($x_3 = R_{f1\to x3} * R_{f2\to x3} * R_{f3\to x3}$). The last iteration occurs when all the factor nodes send their message to all leaves (x_1, x_2, x_4). Finally, x_1, x_2, x_4 should be able to calculate their marginals, since each of them has received all their incoming messages ($x_1 = R_{f1\to x1}$; $x_2 = R_{f2\to x2}$; $x_4 = R_{f3\to x4}$).

2.5 Summary

In this chapter, we have introduced the notion of *Provenance*, *Risk* and *Belief Propagation* as the basis from which to build analytical methods to support due diligence. Since due diligence is about preventive actions to protect something, these three concepts can provide a comprehensive insight to help the operators and authorities in a product supply chain in the task of undertaking and demonstrating due diligence. With the provenance of a product and risk assessment, it is possible to identify the likelihood and location of potential risk in a product supply chain. While identifying the likelihood of contamination relies on the risk assessment model, identifying the location of a product

with high risk can be reliant on the concept of provenance and traceability. In addition, *Belief Propagation* is heavily used to propagate the risk across the lineage of a product supply chain, which is derived from the provenance of a product.

Provenance is a good concept to support traceability in the product supply chain. With provenance, the processes in the production, transformation and distribution of a product can be captured, and associated risk factors can be compared with what is recommended by regulations. Although the notion of provenance has been used to support the traceability of the product, it may not be in a digital form and not in a provenance standardised format that can be interchanged among systems easily. To capture the provenance of a product, we therefore adopt the PROV model because of its ability to model a product's ecosystem and since it can be transformed into a graph that can be a basis for performing inference tasks.

To identify and calculate risk across the product supply chain, several risk models have been developed in both qualitative and quantitative approaches. In this research, our focus is on quantitative risk assessment, where a risk model takes a set of risk factors to calculate the risk in the product supply chain. This quantitative approach considers a stochastic method to represent the randomness within the nature of the problem. Thus, most of the risk models define their associated risk factors in the form of probability distribution in their calculation. Consequently, Monte-Carlo simulation is often used in quantitative risk assessment as a way of coping with the uncertainty and variability in risk.

We have also discussed several concepts that allow us to perform inference. We started with the concept of PGM, where both graphical and probability theories are introduced. In that section, we discussed the basic concept of probability and the underlying idea of inference in the form of a graphical model. In addition, we also learn about multiple types of reasoning that help us understand how the probability changes due to interactions between variables in the network.

To solve inference problems, it is often convenient to convert both directed and undirected graphs into a different graphical representation called a *Factor Graph* (FG). The main reason is that an FG allows a global function of several variables to be expressed as a product of factors over subsets of those variables. This allows some algorithms (i.e., the *Message Passing* algorithm) to compute the marginal and conditional probability more efficiently by re-arranging the *sum* and *product* rules. This is the main reason for choosing an FG as a graphical representation to perform inference. Once we have a graphical model of the problem in FG, we can do inference and reasoning with the *Belief Propagation* algorithm, which is based on *Message Passing* schema.

Finally, an inference algorithm is also discussed in this chapter. Specifically, the *Sum-Product* algorithm that applies the local *Message Passing* algorithm that allows the nodes to communicate their local state by sending messages over the edges and updating them

recursively. The *Sum-Product* algorithm is chosen since this algorithm is based on the *Message Passing* algorithm. The *Message Passing* algorithm works based on the GDL, which can be used to calculate the joint probability distribution more efficiently by re-arranging the two main operators in the probability distribution, namely summation and multiplication. With the *Sum-Product* algorithm, reasoning and inference on acyclic graphs are also possible since it calculates conditional distribution on an unobserved node, based on an observed node.

Overall, since the product supply chain can be defined as a set of interconnecting product processing steps, it should encode all the knowledge about how the product is transformed from its source to its final stage. In this case, the product supply chain can be modelled by capturing the provenance of the product which can later be converted into an FG. Once the FG of the food supply chain is constructed, one can manipulate the model with the *Belief Propagation* algorithm to perform inferences about the risk over the PG of the product. In other words, we can estimate the probability that represents a risk with or without precondition in each process in the product supply chain.

Chapter 3

Research Methodology

In this chapter, we detail the methodological procedure in our work to assess risk by means of provenance-based *Factor Graph* (FG) representation, using *Belief Propagation* as an inference technique. We begin our methodology by integrating the notion of provenance with risk. We compare the existing approach to assessing risk by using Monte-Carlo (MC) simulation with and without the notion of provenance and develop a specific ontology to capture the provenance in a specific domain. After that, we design a pipeline, which we call *prFrame*. The pipeline itself is explained in detail in Section 4. Next, we apply the *prFrame* pipeline to a specific case described in the literature. Finally, our methodology ends with systematic evaluation through a set of experiments.

3.1 An integration of provenance with a risk

As mentioned in Section 2.2.3, the concept of provenance has been researched in various domains. In those cases, the notion of provenance is integrated into the systems to gain more benefits. In this research, we integrate provenance and risk through several steps, namely: the provenance-based Monte-Carlo simulation and the domain-specific ontology development with an aim to assess risk along the provenance of a product. This work is our second contributions where risk and provenance in the *Provenance Graph* (PG) are integrated to allow the use of inference techniques.

With the specifically developed ontology as an extension of PROV-O, we can integrate the provenance of a product and its risk (risk models and their associated risk factors). This approach will record the provenance of something together with its risk along its supply chain. Hence, the calculation of risk will only follow the lineage of a product by using the information of what actually happens in the supply chain.

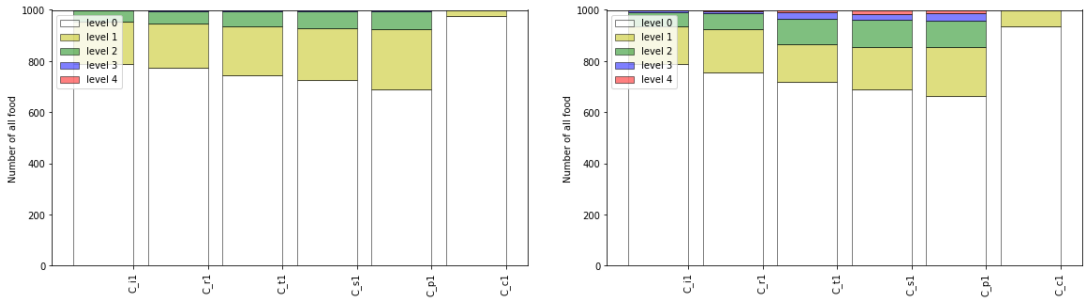
The integration adapts *process-center provenance*, where the processes in the ecosystem become the main focus. A process will take the inputs and calculate the outputs based

on the risk models and the risk factors. The risk model and its risk factors are therefore associated with a process and are annotated in prov:Activity, which represents the activity or process. In addition, the integration of provenance and risk using PROV can ultimately be represented as a graph. This has several advantages, such as acting a basis to perform an inference task by means of a graphical representation. More details of this integration can be seen in Section 4.1.

3.1.1 Provenance-based Monte-Carlo simulation

There have been several quantitative studies to measure risk using Monte-Carlo (MC) simulation. Such simulations are performed mainly because some of the risk factors are unknown and are presented in the form of a probability distribution. By using MC simulation, it is possible to take into account all the possible values of the risk factors, to predict the risk along the supply chain. Most of these approaches, however, use less recent and less contextual data for their risk calculation. This may cause an irrelevant measurement of risk calculation.

In our work, by contrast, we aim to integrate provenance as a means to an end to produce more relevant and contextual risk assessment. We address this problem by providing provenance-based descriptions as a basis to simulate the flow of products in the product supply chain with MC simulation. This is our third contribution, in which the MC simulation is performed by means of the PG. After the process of integration, the integrated provenance-risk graph is generated with the values of the risk models and their associated risk factors in the form of a probability distribution. These probability distributions (as well as the distribution of the numeric input values) are simulated through MC simulation to generated predicted output values (*pov*-more details about *pov* is explained in Section 4.1) . As an illustration, Figure 3.1 shows the result of MC simulation with and without provenance.



(a) Without Provenance

(b) With Provenance

FIGURE 3.1: Monte Carlo simulation with and without provenance.

Figure 3.1 shows the amount of food contamination (*y-axis*) in several food stages (*x axis*). Although the simulation is based on a report that was produced in 2002 [65] and is a based of our set of experiments in Chapter 6, Figure 3.1 demonstrates our argument that provenance can potentially improve the relevancy of decision making as the result of the simulation provenance-based MC simulation (with more recent information) is different with the result of non-provenance-based MC simulation. Figure 3.1a shows the results of simulation without provenance, which relies only on historical data from a report produced in 2002 [65]. This shows that there is very little food that is contaminated to levels 3 and 4 in the food stages. After more than ten years, this result may no longer be relevant, however. In contrast, Figure 3.1b shows the result of simulation with provenance, which captures the provenance on a timely basis, even after the report was published. It clearly shows more widespread level 3 and 4 contamination in some food stages. With this simple illustration, it is reasonable to assume that provenance will make the result more contextual and relevant than the simulation without provenance.

3.1.2 Ontology development

The integration between provenance and risk requires the conceptualisation of a specific domain. This process can be achieved by using Ontology and it is this that is our first contribution where we develop an ontology to model the provenance of a product and to map other legacy ontologies. We use ontology for three reasons. First, because it is standardised and can be interlinked easily with another ontology. Second, because PROV (a standardised provenance language) also uses its ontology, PROV-O, to model the provenance of something. Third, ontology enables us to perform some reasoning, which is essential to demonstrate due diligence. Since PROV-O is general, we need to develop a more specific ontology as an extension of PROV-O. The extended ontology is specified to the domain of the problem, namely, in our research, the *prFrame*, *prFood* and *prFoodMapping* ontologies we develop in Chapter 5.

These extended ontologies allow us to model a specific product, all operators who produce and distribute the product, and any documentation related to that product. The integration also models the product supply chain based on the provenance of the product, which we refer to as the provenance-based supply chain. This supply chain focuses on the set of processes with their associated risk. In addition, a provenance-based supply chain represents the ecosystem of the product making it easy for the operators to comply with their requirements.

3.2 Design a pipeline as a framework

prFrame is a framework to estimate a product's risk in its supply chain described by provenance, which allows for observations (by directly sampled contamination levels) to

be taken into account, as well as estimates to be inferred for unobserved parts of the supply chain. Overall, *prFrame* comprises two major techniques, namely, incorporating the risk models and their risk factors with the provenance of a product, and a conversion from a *Provenance Graph* (PG) into a *Factor Graph* (FG). Those techniques result in *prFrame* to reason, estimate and understand risk across the supply chain, even when we have only partial knowledge of it. In addition, it can be the basis for the operators and authorities to develop a rationale for control procedures.

In the first technique, PROV can be used to model a product provenance and to capture the risk model and its risk factors to allow a quantitative risk assessment tool to estimate risk. As mentioned, many risk models use MC simulations to take into account the variation of randomly distributed risk factors to be propagated through mathematical models. This approach relies on the directed nature of PGs, and propagates the predicted output value along the edges of these graphs, according to the evidenced formulae of the risk models. This approach does not support any actual or *up-to-date* knowledge, however, since it relies on distributions of the predicted output values and risk factors, derived from past studies (demonstrated in Section 3.1.1).

The second technique utilises *Belief Propagation* to calculate the marginal probability distribution for each unobserved variable node, conditional on these observed variable nodes over the FG. In the FG, an observed variable node is a variable node whose state is known with certainty, while an unobserved variable node is a variable node whose state is unknown. *Belief Propagation* performs the inference algorithm by rearranging *sum* and *product* rules. Thus, we factorise the probability distribution through the conversion from PG into FG for efficient inference by *Belief Propagation*. This is our fourth contribution, which is a systematic provenance-based factorisation to allow the application of the *Belief Propagation* technique.

3.3 Apply a real-world use case

We demonstrate that the effectiveness of the *prFrame* framework lies in the ability easily to incorporate new evidence that can allow a more accurate estimation of more contextual and relevant risk. We chose food to be our domain of interest for applying our use case. In particular, the risk of food contamination in the food supply chain. This scope drives the development of a food ontology to conceptualise the risk entailed in each process in the food supply chain. In addition, the risk models and the risk factors are those needed to calculate the risk of contamination.

In applying the *prFrame*, we chose a well-structured example with food data and discussion of the chicken supply chain so as to measure the risk of salmonella contamination in broiler chicken [84]. We chose this report as our use case because it describes the *process-by-process* in the chicken supply chain thoroughly . In addition, the report also

provides the complete risk models and risk factors in each process of the chicken supply chain. Note that these risk models and risk factors were specifically developed for salmonella in the broiler chicken. Thus, other risk models and risk factors may need to be considered in the case of other types of bacteria. Finally, all information from the report is modelled with PROV to produce a provenance-based supply chain and *prFrame* pipeline can be executed to assess the risk of contamination. We present this use case in Chapter 5.

3.4 Evaluation through a set of experiments

The aim of our evaluation is to better understand the propagation of risk in the linear and non-linear product supply chain. We perform a set of experiments to see if conversion from PG into a FG through factorisation of probability distribution produces a sensible and plausible risk assessment. This is an important aspect in *prFrame* as it also considers different types of branching that are common in the production and the distribution of a product. There are several aspects that we consider in our experiment.

(a) The first aspect is the accuracy of inference by *Belief Propagation* to determine the states in different set of topologies for linear and non-linear topologies. In both cases, we are concerned with accuracy as the length of topologies get longer. The length of topology is determined by the number of product's stages in its supply chain. Since the accuracy is related to uncertainty, we also experiment with the confidence in performing inference by means of *Belief Propagation*.

(b) We also try to find the best single point to observe, i.e. that which brings the highest accuracy in a graph. In this exercise, we take the average of $P(inf_{(1...n)}|obs)$, where *obs* is a single observed variable node and $inf_{(1...n)}$ is the rest of variable nodes in a FG we want to infer.

(c) In our experiment, we also try to identify the most influential product stage that can change the distribution significantly. We perform this exercise by observing the average accuracy in the stage that can change the distribution significantly.

(d) Another experiment we conduct is an experiment with different paths (or batches) in order try to understand how risk is developing in different structures of the FG.

Finally, all of the experiments are discussed and evaluated in order to understand how the information is propagated in different topologies of the FG to assess the risk. Essentially, we investigate the relevance of the result by *Belief Propagation* in as many possible structures of supply chain as we could have.

3.5 Summary

In this Chapter, we present the methodology we adopted for this research. This methodology entails investigating MC simulation as a common way to measure a risk, and augmenting its result contextually to improve its relevance by using provenance records; developing a specific ontology to allow the integration of provenance and risk in a specific domain; desiging then *prFrame* pipeline to integrate provenance and risk, and performing *Belief Propagation* in respect to the provenance-based supply chain. We then demonstrate the *prFrame* in a simple topology as a case study and finally, we evaluate and experiment with more complex topologies.

Chapter 4

prFrame pipeline

In this chapter, we detail our methodology to produce a systematic approach to assess risk over provenance-based graph representations as a means to the end of demonstrating due diligence. The pipeline comprises a *step-by-step* approach towards the construction of our framework, *prFrame*. Those steps are the integration of provenance and risk, simulation based on this integrated provenance and risk, the conversion of graph representations, and finally, the propagation of risk by means of an inference technique. In the following sections, we thoroughly discuss each of those steps.

The pipeline starts with the construction of a *Provenance Graph* (PG) based on the actual processes in the ecosystem with their risk factors. Note that the processes are represented as a prov:Activity and prov:Entity represents the input and the output of each process. Thus, the risk models and risk factors are annotated as an attribute in their associated prov:Activity. This annotation is done by using a set of ontologies. The main ontology in this context is PROV-O (PROV Ontology)[1] to standardise the provenance of an event and its associated risk, allowing us to perform automatic inferences and reasoning. Essentially, the use of a risk model will mathematically predict the risk in the form of a probability distribution through a Monte-Carlo (MC) simulation.

In the second step, *prFrame* identifies the risk model and its associated risk factors to perform the MC simulation so as to produce our prior belief. The MC simulation is performed to take into account all possible values of the risk factors to measure risk quantitatively in the form of a probability distribution, so that we can answer questions such as " *What is the chance that food gets contaminated after the cooking process alone?*" Also, the MC simulation constructs the Conditional Probability Distribution (CPD) and represents it as a Conditional Probability Table (CPT) which can be used to support the risk propagation through the *Belief Propagation* technique.

[1]Ontologies exist to capture the knowledge in respect to domains of interest, and PROV-O provides a general set of vocabularies and their relations to describe the provenance of something.

In step three, the constructed CPTs are merged back into the PG after the MC simulation before the PG is converted into the *Factor Graph* (FG) in step four. In order to perform the inference technique properly and efficiently, this conversion in step four requires factorisation based on the structure of the PG. *prFrame* accommodates the conversion of linear and non-linear structures. The conversion of the linear structure is coded as $-2O$ (*none-to-one*), $O2O$ (*one-to-one*), while a non-linear structure is coded as $-2M$ (*none-to-many*), $O2M$ (*one-to-many*), $M2O$ (*many-to-one*), and $M2M$ (*many-to-many*). In the more specific case, *prFrame* also categorises the non-linear structure into *dirE* ($M2O_{dirE}$ and $O2M_{dirE}$) and *indE* ($M2O_{indE}$ and $O2M_{indE}$). More details about these structures are explained in Section 4.3 through the process of factorisation.

Finally, *prFrame* utilises the *Sum-Product* algorithm as the *Belief Propagation* technique to propagate the risk automatically by means of the FG inherited from the PG in the presence of new evidence (Section 4.4). The details about the *Sum-Product* algorithm to find the marginal probability distribution have been discussed in Section 2.4 while the procedure to propagate a message with the *Belief Propagation* technique was discussed specifically in Section 2.4.4.

4.1 Integration of product provenance and risk model, and capture of risk factors

This section is our second contribution, which is the integration of risk and provenance in the PG to allow the use of inference techniques. As described in Section 2, provenance records can be used to describe a lineage of something. Although this "something" can be tangible or intangible, we are more concerned here about the derivation of a tangible object, such as a laptop, newspaper, shoes, bags, etc. Those objects can be seen as a business product, which has been through some set of processes in order to reach its final stage. The capability of provenance records to describe the lineage of the product allows us to model the product's derivation from one stage to another. The transition of the stages is a result of the specific processes in the product supply chain. Eventually, the provenance records can also be seen as a product supply chain because they can describe the lineage of the product.

As a product is changed from one stage to the next stages through its processes some risks are introduced as part of those processes. These risks are modelled by the experts to calculate or measure their impact. These models take their associated risk factors as the parameters for their calculations. Thus, capturing these models and their risk factors along the product supply chain will make risk more visible from the beginning to the end of the product's lifetime.

In our approach, the integration of provenance and risk uses PROV to produce the provenance records and can then be visualised as a PG, which represents the product

supply chain. This provenance-based supply chain holds the information about the risk to be calculated; and hence allows the propagation of risk over the PG. In general, PROV core concepts (i.e., prov:Entity, prov:Activity, and prov:Agent) map a product, its set of processes and the operators; and the relations between them describe the derivation of the product through their processes. In more detail, PROV Ontology (PROV-O) is extended by additional specific ontologies to capture the values of the related risk factors and use them as the inputs to the mathematical functions defined as the risk models. See Section 2.3 about the risk model and risk factor.

With the additional domain-specific ontology, the details of related risk models and risk factors can be annotated in the prov:Activity when constructing a provenance-based supply chain. Later during the MC simulation, the input from the *used* prov:Entity (edge prov:used) can be processed with the risk factors and risk model in the prov:Activity to generate the output in *generated* prov:Entity (edge prov:wasGeneratedBy). This approach allows us to break the processes in a product supply chain into individual independent modules and map each module with its associated risk within the PG, as shown in Figure 4.1.

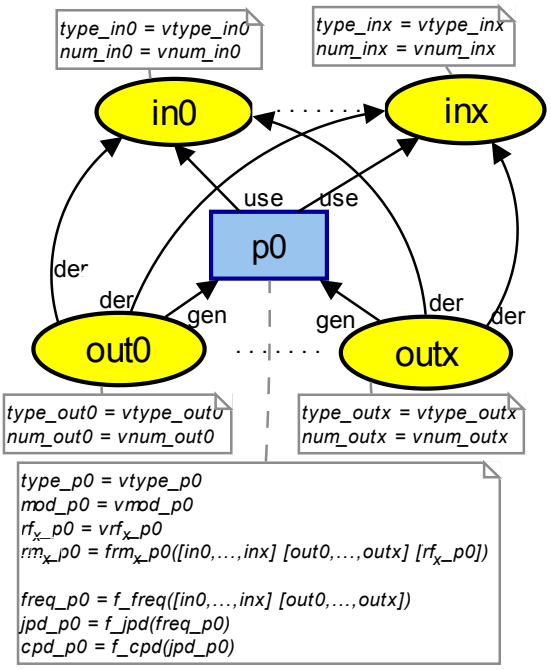


FIGURE 4.1: The general principle for overlaying risk in a provenance-based supply chain.

Figure 4.1 depicts our approach to the integration that describes how a process of a product (blue rectangle **p0**) uses the inputs of products (yellow oval {**in0**,...,**inx**}) to produce (or transform into) their outputs (yellow oval {**out0**,...,**outx**}). In detail, it shows that **p0**, as an instance of prov:Activity, has dependencies with input prov:Entity {**in0**,...,**inx**} through the prov:used (indicated by *use* edge) and output prov:Entity {**out0**,...,**outx**}

through prov:wasGeneratedBy (indicated by *gen* edge). With this dependency, this approach is able to describe which outputs use which inputs by using link prov:wasDerivedFrom (indicated by *der* edge).

Each prov:Activity (i.e., **p0**), contains associated risk factors as the parameters for a function in a risk model to calculate the output values. These risk factors are captured as the attributes in rf_x_p0, where $_x$ is an index to accommodate multiple risk factors. The values of the risk factors, vrf_x_p0, are often in the form of a probability distribution. We intentionally capture the risk factors by annotating them in a PG for two reasons. First, to be able to explain and reason the phenomena of the risk before and after a process; second, to construct a risk model that calculates the output values to mimic or model how the actual process produces its output.

Besides the risk factors, the risk models (i.e., rm_x_p0) also need to be captured in order to calculate the numeric output values (i.e., *num_out0,...,num_outx*). This mathematical formula represents a risk model that takes into account all the risk factors and all numeric input values (i.e., *num_in0,...,num_inx*). The mathematical formulae of the risk models are meant to generate the distribution of numeric output values through MC simulation (Section 4.2). A risk model is defined as a mathematical formula that mimics the actual process (i.e., *mod_0*). While some processes are easy to model, others are complicated and may require multiple risk models if they are to be modelled fully. In the event of such multiple risk models, a prov:Activity will have multiple rm_x_p0, where $_x$ indicates an index for each risk model.

Note that the numeric output values of one process may become the input values for the next process. In *prFrame*, they are known as the *predicted input value* (*piv*) that is captured in the prov:Entity input and the *predicted output value* (*pov*) that is captured in the prov:Entity output. We define a predicted output value in Definition 4.1. This means that all numeric input values or *piv*, the mathematical formulae of the risk models, and the values of the risk factors in each prov:Activity are important and are expected to be provided in the expected PG.

Definition 4.1. A predicted output value (*pov*) can be defined as a predictive numeric value generated by a risk model based on its associated risk factors and its numeric input.

In addition, *prFrame* itself has its ontology with 4 main entities, prFrame:ProductStage, prFrame:ProductProcessing, prFrame:risk, and prFrame:structure. *prFrame* ontology is designed to be less domain-specific than *prFood* as it only concerns the stages and processes in the product supply chain and the structure of the product supply chain. By putting the prFrame:ProductStage and prFrame:ProductProcessing as the subclasses of prov:Entity and prov:Activity, it will allow us to model a sequence of processes with their input and output stages as a general product supply chain. Finally, prFrame:structure

is attributed with class prFrame:ProductProcessing as a data property to help indicating the nature of processes and structuring the *Factor Graph* (FG) in the later steps. In addition, prFrame:risk will only be attributed to the prFrame:ProductProcessing class that posses risk we are interested in modeling. This is important as converting to the FG will only for prov:Activity that has prFrame:risk. Figure 4.2 shows the core class diagram of *prFrame* ontology.

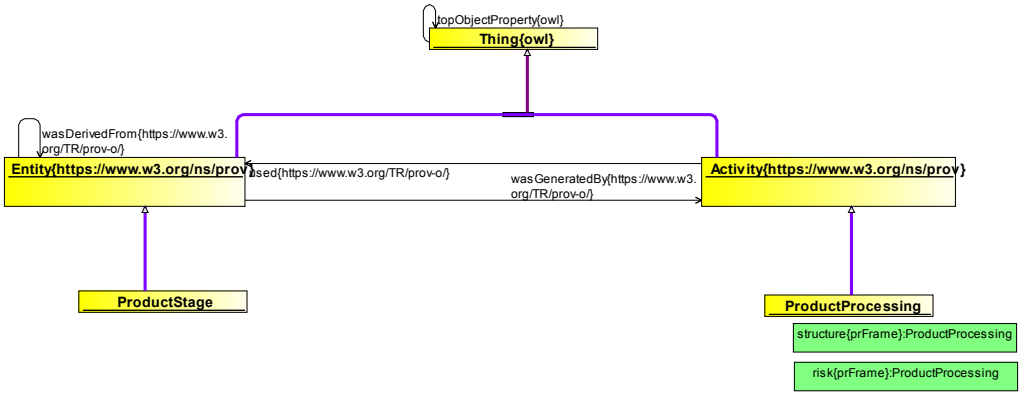


FIGURE 4.2: A class diagram of *prFrame* ontology.

In Figure 4.2, as prFrame:ProductStage, prFrame:ProductProcessing are the subclasses of prov:Entity and provenance respectively, they inherit the same characteristic of their own parents. This will allow any specific product (not limited to food as our use case) to use *prFrame* ontology for their own supply chain, similar with *prFood* in the food domain. In addition, the prFrame:structure is restricted to prFrame:ProductProcessing as a datatype property because prFrame:structure will determine the shape or topology of the product supply chain.

To capture or annotate the risk models and risk factors, *prFrame* uses a domain-specific ontology because the risk models and risk factors have different definitions and conceptualisations between domains. For example, prFood:HomeTempDist is a risk factor in the food domain that aims to capture the temperature so as then to be able to calculate the growth rate of bacteria; on the other hand, healthOnt:cholesterol is a risk factor in the health domain that aims to capture cholesterol level in order to calculate the health index of a person.

Similar to rf_x_p0, the numeric input values, *num_in0* to *num_inx*, are often captured as a probability distribution in order to represent a range of possible input values during the MC simulation. In the simulation, *f_freq([in*$0, ..., inx$*] [out*$0, ..., outx$*]*, *f_jpd(freq_p0)*, and *f_cpd(jpd_p0)* are called as functions to construct the Frequency Table, Joint Probability Table (JPT), and Conditional Probability Table (CPT) with the scope of input and output of the prov:Activity. The function *f_freq([in*$0, ..., inx$*] [out*$0, ..., outx$*] [*rf_x_p0*])* is meant to quantify the changing of the states between the inputs ($[in0, ..., inx]$) and the outputs ($[out0, ..., outx]$) after the simulation. In other words, it quantifies the changes

of *f_jpd(freq_p0)* and *f_cpd(jpd_p0)* before becoming the subsequent function to construct the joint and conditional distributions (and present them as the JPT and CPT) based on the result of quantified changing state by function *freq_p0*. The details of how those functions construct their tables are discussed in Section 4.2.

Eventually, this integration generates a PG with the notion of risk annotated within it as depicted in Figure 4.3. Figure 4.3 also represents an example of the expected PG. This allows risk to be propagated within the PG of a product, and this integration can help us understand what processes the product has gone through and thereby to assess the quality and safety of the product in the form of a basic graph for the MC simulation before its conversion into another graphical representation for risk propagation by means of the *Belief Propagation* technique. The process of instantiating the predicted output values or *pov* by performing an MC simulation is discussed in the next section.

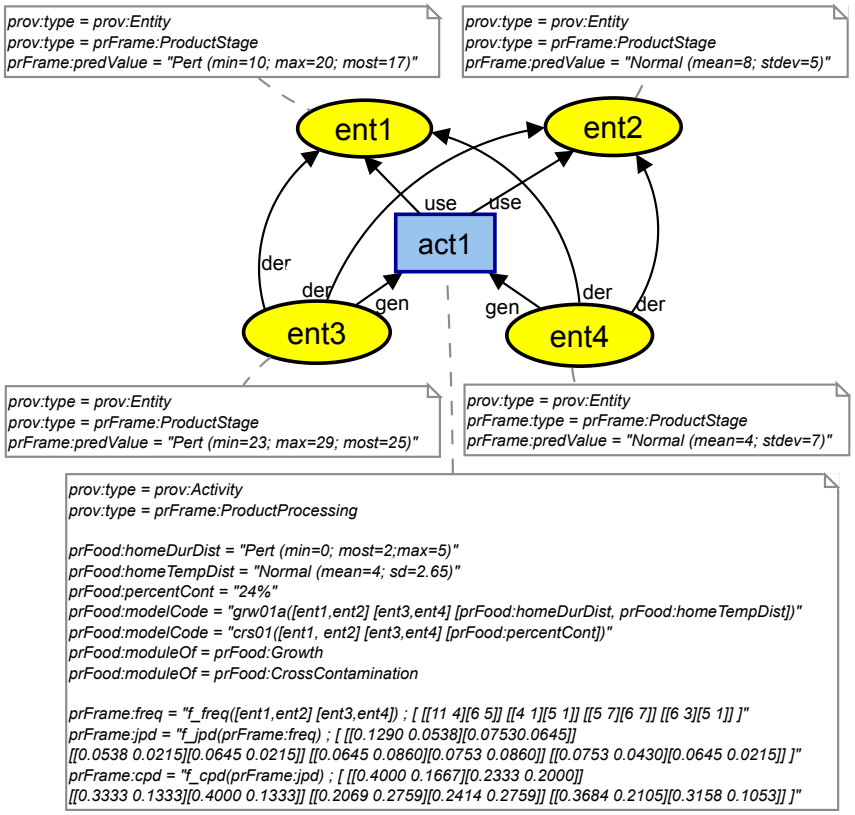


FIGURE 4.3: Example of the expected provenance graph with its populated values based on Figure 4.1.

4.2 A provenance-based Monte-Carlo simulation technique

In this section, we describe our third contribution, which is a provenance-based description as a basis of a Monte-Carlo simulation. As mentioned, the values of the risk factors

are often represented as a distribution. Thus, a Monte-Carlo (MC) simulation is performed in order to accommodate all possible values in the distribution of the input values and the risk factors. MC simulation is a computer-based technique allowing variation in randomly distributed inputs to be propagated and combined through mathematical models [85][86]. The simulation consists of the repetition of the same calculation (by using the formula captured as the risk model) multiple times (iterations). In each iteration, a value is selected from the distribution of each parameter, and the predicted output value is calculated. This ultimately generates a distribution of the predicted output values after each process in the PG.

The process of the MC simulation begins by identifying the processes and their risk models in the PG. Note that the PG is already annotated with the risk information as a result of the integration in the previous section. Each risk model and its associated risk factors are annotated in the prov:Activity to model how risk develops in the process. These risk factors are often derived from past studies or historical data that has been captured on provenance. Based on the initial values as the inputs, a risk model calculates a predicted output value with its mathematical function. This output value becomes an input for the next process until the end of the supply chain. As an iterative simulation, each process in the PG generates a set of predicted output values in a form of distributions. These distributions become the basis from which a Frequency Table, JPT and CPT can be constructed and annotated back into the initial PG. Listing 4.1 shows the pseudocode used to perform the MC simulation for the PG in Figure 4.1.

```
Step 1: Load a provenance graph of a product that represents its supply chain.
Step 2: Identify each activity type, its risk model and associated risk factors
        with function f_ActEnt(PG).
Step 3: With function f_ActEnt(PG), for each activity identified in Step 2,
        identify each entity with the relation prov:used.
Step 4: Apply the function f_State(ActEnt) to calculate the predicted output
        values for each entity with relation prov:wasGeneratedBy in respect to
        each activity.
Step 5: Apply the function f_freq([in0,...,inx] [out1,...,outx]), f_jpd(freq_p0),
        and f_cpd(jpd_p0) in each prov:Activity as a basis for risk propagation
        through Belief Propagation.
Step 6: Merge the result of the CPT back into the prov:Activity in the original
        provenance graph.
```

LISTING 4.1: A pseudocode for provenance-based MC simulation.

Listing 4.1 shows how the integrated provenance-risk graph allows the risk model to calculate risk across the product supply chain. Function *f_ActEnt(PG)* loads this integrated provenance-risk graph to identify the processes and their associated risk models and risk factors (Step 2). The function identifies all prov:Activity that also has a type prFrame:ProductProcessing since that prov:Activity represents the processes that are essential in the supply chain. Once the processes are identified, the same function tries to identify the prov:Entity (and its associated input values) as the inputs to a risk model in each process (Step 3). In the Algorithm 1, the PG is taken as an input to produce a

list of a set of identified activities and entities. Algorithm 1 identifies each object that has type prov:Activity and prFrame:ProductProcessing. This prov:Activity, as well as its attributes, is assigned to Act_x (Algorithm 1 line 7). Next, the algorithm identifies the *used entities* through the edge with type prov:used (Algorithm 1 line 9) and *generated entities* through the edge with type prov:wasGeneratedBy (Algorithm 1 line 11). The Act_x, $EntUse_x$, $EntGen_x$ are appended as a list of prov:Activity with its prov:Entity as *used* and *generated entities*. *ActEnt* therefore consists of a set of lists of prov:Activity with its *used* and *generated* prov:Entity. To illustrate Algorithm 1, Figure 4.3 will produce a set of list as *[act1, ent1, ent3]*, *[act1, ent1, ent4]*, *[act1, ent2, ent3]*, *[act1, ent2, ent4]* as a value of *ActEnt*. Each list consists of a triple with specific order [Act_x, $EntUse_x$, $EntGen_x$] and this should be preserved in order to identify which are the process, input, and output. However, as each list already represents a single dependency between $EntUse_x$ and $EntGen_x$ through Act_x, the order of each list in *ActEnt* is not important.

After executing Algorithm 1, function *f_State(Act,Ent)* performs MC simulation based on the formula of the risk models to generate all possible predicted output values based on their associated risk factors and input values (Step 4). When the simulation has been done, the distributions of the *predicted output value* (*pov*) are used to construct the Frequency Table, JPT and CPT via the functions *f_freq([in*$0, ..., inx$*] [out*$0, ..., outx$*])*, *f_jpd(freq_table)*, and *f_cpd(jpd_table)* subsequently (Step 5). Basically, the distributions of the predicted output values can be seen as having an impact on each process (represented by prov:Activity) in the product supply chain (represented by PG). This impact takes into consideration all possible values of the inputs and risk factors. Algorithm 2 is then executed to instantiate the *pov* in each prov:Entity identified in the PG. The algorithm takes a set of lists of prov:Activity with its used and generated prov:Entity. It starts by identifying each list in *ActEnt* that consists of prov:Activity, used prov:Entity, and generated prov:Entity. For each list, the *pov* is calculated based on the input (used prov:Entity), and the risk models and risk factors in prov:Activity (Algorithm 2 line 3). After calculating *pov*, the construction of the distribution tables is executed so as ultimately to produce the list of CPTs (Algorithm 2 lines 4, 5 and 6). Figure 4.4 illustrates the construction of a Frequency Table, JPT and CPT.

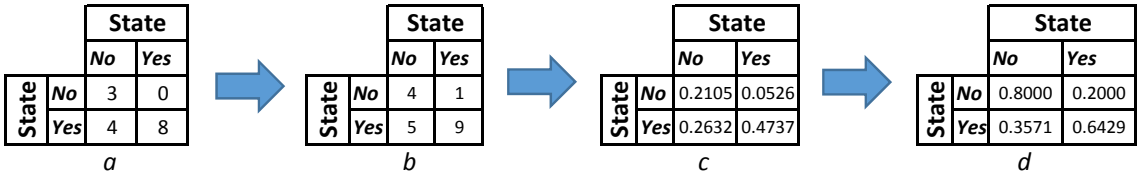


FIGURE 4.4: The process of constructing a CPT from a Frequency Table.

The Frequency Table in Figure 4.4a shows the frequency values between the states of a process before (as a row) and after (as a column). It shows that there are a total of 15 events, four of which have changed their states from *Yes* to *No*. The Frequency Table in Figure 4.4a shows that there may be a possibility of having a cell with 0 (zero)

Algorithm 1 function ActEnt(pG)

```
Input  : pG: Provenance Graph
Output: ActEnt: A list of a set of identified Activities and Entities
1  var Act_x: prov:Activity
2  var EntUse_x: prov:Entity Used
3  var EntGen_x: prov:Entity GeneratedBy
4  var o: Object
5  foreach o ∈ pG do
6      if type(o)==prov:Activity AND type(o)==prFrame:ProductProcessing then
7          Act_x ← type(o)
8          foreach type(o) == prov : usage ∈ Act_x do
9              EntUse_x ← type(o)
           end
10         foreach type(o) == prov : wasGeneratedBy ∈ Act_x do
11             EntGen_x ← type(o)
           end
12         ActEnt ← append([Act_x, EntUse_x, EntGen_x])
       end
   end
13 return ActEnt
```

Algorithm 2 function State($ActEnt$)

```
Input  : ActEnt: A list of a set of identified Activities and Entities
Output: CPT: A set of CPD tables in each prov:Activity
1  var l: List
2  foreach l ∈ ActEnt do
3      piv, pov ← calculatePov(ActEnt_l)
   end
4  freq_table ← f_freq([EntUse_x, piv][EntGen_x, pov])
5  jpd_table ← f_jpd(freq_table)
6  cpd_table ← f_cpd(jpd_table)
7  CPT ← append(cpd_table)
8  return CPT
```

value, such as the cell from state *No* to *Yes*. This would subsequently create a JPT and CPT with 0% probability, i.e. impossibility. With the assumption, however, that our observations are never able to capture an actual event with complete fidelity, we assign a small probability for all events by adding 1 in all cells in the Frequency Table as shown in Figure 4.4b. Adding 1 indicates that at least there an event occurs once in each cell of the JPT. This will only cause small changes in probability and the resulted CPT will have a little effect (does not differ much from actual observation); yet the notion of impossibility can be avoided. This notion is important since it captures the natural uncertainty of the occurrence of an event in many domains, representing our inevitable lack of knowledge or erroneous observation as there is always an *unknown factor* that contributes to the occurrence of an event. For example, where food is found to be contaminated even though all known factors suggest that it should be uncontaminated.

In Figure 4.4c, a JPT is constructed by dividing each cell by the total value of the Frequency Table in Figure 4.4b. Further, the CPT in Figure 4.4d is constructed by dividing each value of the JPT with its corresponding row that represents the states before the process. Basically, the CPT is a result of the factorisation of the joint distribution in the prov:Activity for risk propagation. The constructed CPT helps us answer more complex questions, such as "*What is the chance of a product having a Yes state after the process*

if that product has a No state before the process?" For example, in Figure 4.4d, the probability of getting state *Yes* after the process, given the state *No* before the process is 0.2000.

The constructed CPTs are merged back into the original PG after the MC simulation before the PG is converted into another type of graph for inference purpose. The CPTs are annotated in their associated prov:Activity so as to represent the conditional distributions between the input prov:Entity and output prov:Entity. In other words, the CPTs represent the dependency between the products before and after the processes along the product supply chain.

The factorisation of JPD, results in a CPT as a factor. This factor will be the basis for an inference technique to propagate the risk on the basis of the provenance-based supply chain. Here, we use *Belief Propagation* as an inference technique that performs an inference task efficiently based on the *Factor Graph* (FG). Thus, in the next section, we introduce our next step, namely to convert a PG into an FG as a basis for risk propagation with the *Belief Propagation* technique.

Before the construction of the distribution tables (Frequency Table, JPT and CPT), the number of states in each prov:Entity need to be defined. The fewer the states, the less complicated the construction of the distribution tables, as explained in Section 2.4.2. Moreover, the computation cost is more expensive when the number of states are larger because we have to consider all the combinatorial possibilities of all the inputs (node parents) and outputs (node children), as described in Section 2.4.2 when constructing the CPT. Figure 4.3 exemplifies the final populated PG based on Figure 4.1.

4.3 The conversion of a Provenance Graph to a Factor Graph

This section describes our fourth contribution, which is a systematic provenance-based factorisation to allow the application of the *Belief Propagation* technique. As mentioned, the notion of *Belief Propagation* demands a *Factor Graph* (FG) representation to allow it to perform an inference task efficiently. Thus, the PG generated by Listing 4.1 is converted into the FG. Note that, the PG has previously been annotated with the CPTs from the MC simulation. Basically, all CPTs are the result of the factorisation of the joint distribution in the product supply chain, and those CPTs become the underlying functions of the factors in the factor nodes after the conversion.

In an FG, a factor can be described as a function that takes arguments from the variable nodes and returns a value for every possible combination arising from those variable nodes. In defining a factor from the PG, we use the annotated CPT in a prov:Activity, which holds the notion of conditional probability for every prov:Entity that is linked to it.

The connection between a prov:Activity and a prov:Entity is done via both edges *Usage* (prov:used) and *Generation* (prov:wasGeneratedBy), in the presence of the edge *Derivation* prov:wasDerivedFrom that identifies the input(s) and the output(s) of a process. Algorithm 3 shows the pseudocode of the conversion.

Algorithm 3 function factorGraph(pG)

```
Input  : pG: Provenance Graph
Output: fG: Factor Graph
1  var n_x: Variable node ; var f_x: Factor node ; var o: Object ; var unDirEdge: Undirected edge ; var dirEdge:
     Directed edge ;
   Function convertEntity(o):
2  |  return n_x
   Function convertActivity(o):
3  |  return f_x
   Function convertEdge(o):
4  |  if type(o)==prov:used then
5  |  |  return unDirEdge_x
   |  end
6  |  if type(o)==prov:wasGeneratedBy then
7  |  |  return dirEdge_x
   |  end
8  foreach o ∈ pG do
9  |  if type(o)==prov:Entity AND type(o)==prFrame:ProductStage then
10 |  |  n_x ← convertEntity(o)
   |  end
11 |  if type(o)==prov:Activity AND type(o)==prFrame:ProductProcessing then
12 |  |  f_x ← convertActivity(o)
   |  end
13 |  if type(o)==prov:used OR type(o)==prov:wasGeneratedBy then
14 |  |  edge_x ← convertEdge(o)
   |  end
   end
15 return Factor Graph (fG)
```

Algorithm 3 shows how the conversion is executed. The process of conversion takes the PG with the annotated CPTs as an input. Next, the algorithm identifies all objects (i.e., prov:Entity, prov:Activity and all edges) in the PG (*line 8*). For each identified object, the algorithm converts prov:Entity that has a type prFrame:ProductStage into a variable node (*line 10*), prov:Activity that has a type prFrame:ProductProcessing into a factor node (*line 12*), and the edges (prov:used and prov:wasGeneratedBy) into edges in an FG (*line 14*). prFrame:ProductStage and prFrame:ProductProcessing are essentially the prov:Entity and prov:Activity that contribute to the supply chain. To give a notion of the direction of dependency, the edge prov:wasGeneratedBy is converted into a directed edge, but the edge prov:used is converted into a non-directed edge. While converting prov:Entity into a variable node is a *one-to-one* mapping, converting prov:Activity into a factor node requires the notion of factorisation of probability distribution among the variable nodes it connects to.

The notion of *d-separation* is essential in the conversion of prov:Activity since it determines the dependency between variable nodes in the FG. As discussed, the concept of *d-separation* configures how influence travels between nodes (i.e., variable nodes) in three patterns (see Section 2.4.2). With *d-separation*, exploiting the factorisation between variable nodes in the FG can be performed properly with *Belief Propagation* as

an inference technique. In the provenance context, the role of *d-separation* can be translated into several structures as *null-to-one* ($-2O$), *null-to-many* ($-2M$), *one-to-one* ($O2O$), *one-to-many* ($O2M$), *many-to-one* ($M2O$), and *many-to-many* ($M2M$). These are annotated in each prov:Activity as it represents a process in the supply chain.

The first two structures are $-2O$ and $-2M$. $-2O$ is a linear structure with only a single prov:Activity and a single prov:Entity generated by that prov:Activity, as shown in Figure 4.5a. On the other hand, $-2M$ is a non-linear structure with a single prov:Activity and multiple prov:Entity generated by that prov:Activity, as in Figure 4.5b. In both structures, there is no dependency between prov:Entity and the prov:Activity since those structures do not have a prov:used edge. In the context of *Bayesian Network* (BN), these structures represent the nodes without a parent. Converting these structures into the FG produces the same probability distributions as the marginal probability distribution or the probability of occurrence in a single event, as depicted in Figure 4.5c and Figure 4.5d. Note that in Figure 4.5d, the conversion of $-2M$ introduces multiple factor nodes because each individual prov:Entity does not have dependency on another prov:Entity (as there is no edge of prov:used). They can therefore be represented with an individual function in each factor node that determines the probability distribution of each variable node.

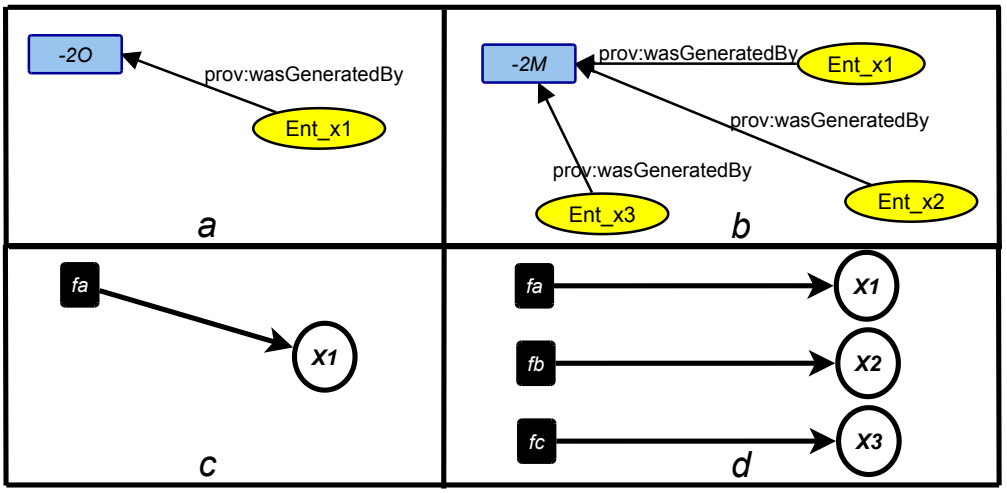


FIGURE 4.5: The $-2O$ and $-2M$ structures.

The third structure is $O2O$. This linear structure always involves the dependency between only two prov:Entity connected by one prov:Activity, as shown in Figure 4.6a. The structure is established with the edges prov:used to prov:Entity (*used entity*) and prov:wasGeneratedBy from prov:Entity (*generated entity*). These prov:Entity (*used* and *generated entities*) are linked by the edge prov:wasDerivedFrom to indicate that the *generated entity* is originated from the *used entity*. In this structure, the annotated CPT in prov:Activity determines the conditional distribution between the *used* and *generated entities* as $P(Ent\_x2|Ent\_x1)$, as in Figure 4.6a. Because of the dependency in this structure, the factorisation will generate a single factor node, fa, within the scope of its

variable nodes ($x1$ and $x2$), as shown in 4.6b. This structure becomes the basic structure for the more complicated structures.

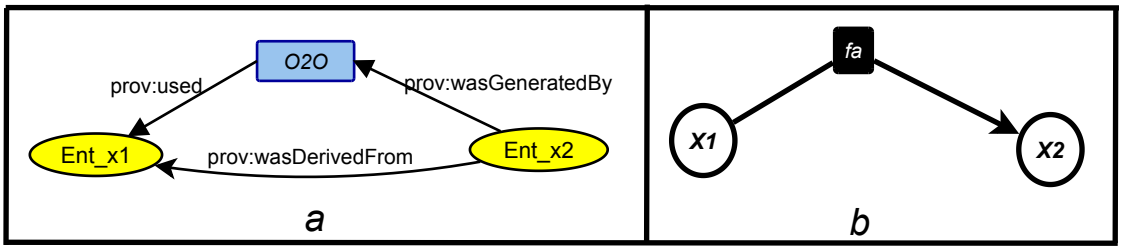


FIGURE 4.6: An $O2O$ structure.

The conversion is more complicated in the non-linear structures, $M2O$ and $O2M$. We define $M2O$ as a structure in the PG with a single prov:Activity that takes multiple prov:Entity (*used entities*) to generate a single prov:Entity (*generated entity*), as shown in Figure 4.7a. On the other hand, $O2M$ is defined as a structure in the PG with a single prov:Activity that takes a single prov:Entity (*used entity*) to generate multiple prov:Entity (*generated entities*), as shown in Figure 4.7b. In these structures, the *generated entity(ies)* are linked to the *used entity(ies)* through the edge prov:wasDerivedFrom. Due to the nature of a product supply chain we categorise the non-linear structure ($O2M$ and $M2O$) into two categories each, *dirE* (*dependent entity*) and *indE* (*independent entity*). The difference between these two categories is a dependency between the *used entity(ies)* and the *generated entity(ies)* and they are illustrate in Table 4.1 for easy reading.

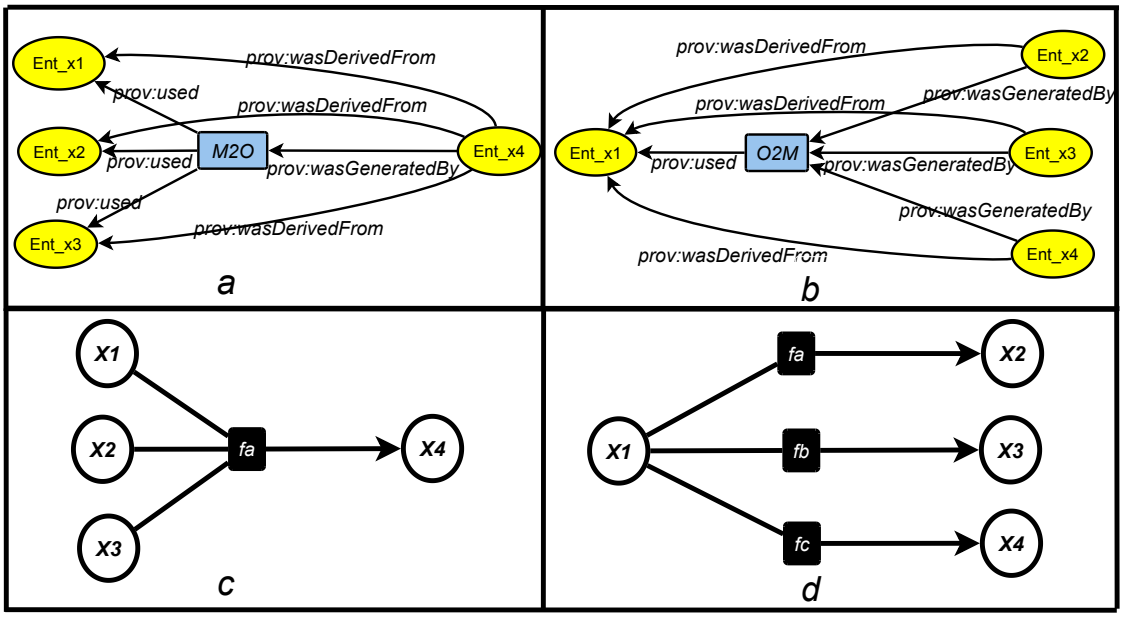


FIGURE 4.7: The $M2O$ and $O2M$ structures.

The *dirE* structure refers to a non-linear structure where a direct dependency exists between all the instantiated items of *used entity(ies)* and *generated entity(ies)*. Since all instantiated items in the *used entities* and *generated entity* hold a dependency, we can

Non-linear Structure	Description
$M2O_{dirE}$	A directed non-linear structure where an individual output is generated based on each individual inputs. **Illustration**: A process where a vegetable and a mushroom are blended with a sauce. Because the generated entity (a blended vegetable-mushroom sauce) contains all the parts from the used entities (a vegetable, mushroom and sauce), the process of blending is categorised as $M2O_{dirE}$.
$O2M_{dirE}$	A directed non-linear structure where all individual outputs are generated based on an individual input. **Illustration**: A process where a single tomato is sliced into three pieces. In this example, the process of slicing is categorised as $O2M_{dirE}$ because the original tomato (used entity) ends up as three tomatoes (generated entities).
$M2O_{indE}$	An indirect non-linear structure where an individual output is generated based only an individual input from multiple inputs. **Illustration**: Chickens are collected from multiple different locations into a single location. In this example, the chickens are not mixed or blended; thus, the condition of chicken before and after the process does not change. This process only affects the quantity of chicken in a single generated entity, which is the total of all chickens from the used entities.
$O2M_{indE}$	An indirect non-linear structure where all individual outputs are generated based only on an individual input. **Illustration**: The chickens from one location are transported into three different locations. In this example, a single chicken cannot possibly be in all three destination locations (compare with the previous example about slicing a tomato in $O2M_{dirE}$). Thus, this process of this transporting is categorised as $O2M_{indE}$ and affects only the quantity of chickens in the generated entities.

TABLE 4.1: Illustrations of the different non-linear structures in the food supply chain.

model this structure as $P(Ent\_x4|Ent\_x1, Ent\_x2, Ent\_x3)$, where Ent_x4 is a *generated entity*, and Ent_x1, Ent_x2 and Ent_x3 are the *used entities* (Figure 4.7a). In the product supply chain, this means that each individual product item in Ent_x1, Ent_x2, and Ent_x3 contributes to the existence of each individual product item in Ent_x4. The converted factor node of this structure has a factor (fa) that holds the factorisation of $P(x4|x1, x2, x3)$ for all variable nodes $x1$, $x2$, $x3$, and $x4$ in Figure 4.7c.

Another non-linear structure that has direct dependency is $O2M_{dirE}$, where all instantiated items in the *generated entities* take some part of each instantiated item in the *used entity* (Figure 4.7b). The factorisation of this structure can therefore be defined as $P(Ent\_x4, Ent\_x2, Ent\_x3|Ent\_x1)$, where Ent_x1 is a *used entity*, and Ent_x2, Ent_x3 and Ent_x4 are the *generated entities* (Figure 4.7d). Similar to $M2O_{dirE}$, this means that each individual product item in Ent_x1 contributes to the existence of each individual product item in Ent_x2, and Ent_x3 and Ent_x4. Although we can generate a single factor node with the function that holds conditional dependency of $P(Ent\_x4, Ent\_x2, Ent\_x3|Ent\_x1)$, we factorise this single factor node into the multiple factor nodes to preserve the conditional distribution. As a result, the quantity of the generated factor nodes is equal to the *generated entities*, as shown in Figure 4.7d.

As the opposite of the *dirE* structure, the *indE* structure is constructed when a non-linear structure (*O2M* and *M2O*) has no direct dependency between instantiated items in the *used entity(ies)* and *generated entity(ies)*. For the $M2O_{indE}$ structure, the factorised CPD is the same as in $M2O_{dirE}$, where $P(Ent\_x4|Ent\_x1, Ent\_x2, Ent\_x3)$ is the result of the multiplication of $P(Ent\_x4|Ent\_x1)$, $P(Ent\_x4|Ent\_x2)$ and $P(Ent\_x4|Ent\_x3)$ (Figure 4.7a). As a result of the multiplication, this structure also generates a single factor node, fa, with the scope of all variable nodes in the FG, as shown in Figure 4.7c. In the product supply chain, this structure means that each individual product item in Ent_x1, Ent_x2 and Ent_x3 only contributes to some of the individual items in Ent_x4.

The final structure of non-linear chains in this factorisation is $O2M_{indE}$, where no direct dependency exists between the instantiated item in the *generated entities* and the instantiated item in the *used entity*. This also means that each individual product item in Ent_x1 only contributes to some of the individual items in Ent_x2, and Ent_x3 and Ent_x4. In this structure, the factorisation can be defined as $P(Ent\_x2|Ent\_x1)$, $P(Ent\_x3|Ent\_x1)$, and $P(Ent\_x4|Ent\_x1)$, where Ent_x1 is the *used entity*, and Ent_x2, Ent_x3 and Ent_x4 are the *generated entities* (Figure 4.7b). Hence, the quantity of factor nodes after the conversion is also equal to the number of *generated entities*, as shown in Figure 4.7d, which is structurally the same as in $O2M_{dirE}$.

Overall, all of the structures are based on the concept of *d-separation*. The structure of *O2O* mimics causal/evidential reasoning, *O2M* mimics the common cause, and *M2O* mimics the *v-structure* or inter-causal reasoning (see Section 2.4.2). Although the structures of a converted FG for both *dirE* and *indE* are the same, the main difference lies in the context of the product supply chain. When a *dirE* structure is a basis of the MC simulation, the quantity of products before and after a process is equal. This is because a product before a process (*used entity(ies)*) is always a part of a product after a process (*generated entity(ies)*). In contrast, the quantity of products before and after a process is not equal in a *indE* structure because a product before a process (*used entity(ies)*) cannot be a part of all the product(s) after the process (*generated entity(ies)*).

We present Algorithm 4 to determine the type of structure in order to convert a PG into an FG. The algorithm takes the PG with the annotated CPTs as an input and is begun by identifying the object in the PG that has a type of prov:Activity and has a potential risk (*prFrame:Risk*) captured in it (*line 2*). For each identified prov:Activity, the algorithm identifies the prov:Entity that a prov:Activity used (*Ent_u*) and the prov:Entity that a prov:Activity generated (*Ent_wgb*) (*line 3*). Next, the algorithm checks the structure of the identified prov:Activity through *prFrame:structure* and generates the number of factor nodes (*lines 4 to 13*), before returning them (*line 14*).

Finally, we leave the *M2M* (*many-to-many*) structure as our future research. The structure of *M2M* is a process that uses many *used entities* to generate many *generated entities*. Basically, it can be divided into two subsequent processes that consist of *M2O*

Algorithm 4 Algorithm for factoring a factor node in a *factor graph*.

Input : PG: The Provenance Graph **Output:** fac: A factor

foreach $x \in PG$ **do**
  if *type(x)==prov:Activity AND hasAttributes(x)==prFrame:risk* **then**
    $\langle Ent\_u_x, Ent\_wgb_x \rangle \leftarrow \underline{identifyEntity}(x)$
    if *prFrame:structure(x) == −2O or prFrame:structure(x) == −2M* **then**
      $fac_x \leftarrow \underline{construct}(P(Ent\_wgb_x))$
    end
    if *prFrame:structure(x) == O2O* **then**
      $fac_x \leftarrow \underline{construct}(P(Ent\_wgb_x \mid Ent\_u_x))$
    end
    if *prFrame:structure(x) == M2O*$_{dirE}$ **then**
      $fac_x \leftarrow \underline{construct}(P(Ent\_wgb_x \mid \forall\ Ent\_u_x))$
    end
    if *prFrame:structure(x) == M2O*$_{indE}$ **then**
      $fac_x \leftarrow \underline{construct}(\prod_1^n P(Ent\_wgb_x \mid Ent\_u_x))$
    end
    if *prFrame:structure(x) == O2M*$_{dirE}$ *OR prFrame:structure(x) == O2M*$_{indE}$ **then**
      $fac_x \leftarrow \underline{construct}(\forall\ P(Ent\_wgb_x \mid Ent\_u_x))$
    end
  end
end
return fac

and $O2M$. Firstly, $M2O$ where the first process uses many *used entities* to generate one *generated entity*. Secondly, $O2M$ where the second process uses one *used entity* to generate many *generated entities*.

All the structures in *prFrame* are constructed to facilitate the calculation of risk not only in the production, but also in the distribution of a product. This aim is usually achieved with the linear structure only because many general product supply chains are often shown linearly from the production to the consumer without the details of branching. Hence, the details of the products distribution in different locations are neglected. The non-linear structure is introduced to detail for the production and distribution of a product in the product supply chain. For example, the linear product supply chain may only capture a single entity to represent a retailer in its supply chain, although the products are distributed to several retailers. With the non-linear supply chain modelled by *prFrame*, it is possible to represent the retailers in multiple locations. *prFrame* characterises the branching in the non-linear structure into *direct* (*dirE*) and *indirect* (*indE*). The aim is to facilitate the production and distribution of a product in more detail until the batching level. The *dirE* structure is mostly used to model the development of a product from one stage to another stage, since this more closely represents the production of a product. The *indE* structure is mostly used to model the process of the distribution of a product. Thus, the *indE* structure often deals with distribution of bulk or batches of product in the product supply chain.

4.4 Inference by means of Belief Propagation

As described, our inference task is achieved through *Belief Propagation*. To perform inference efficiently, this technique requires an FG. This FG is derived from the CPT-annotated PG through the conversion process set out earlier. In short, its technique propagates the messages between factor nodes and variable nodes across the FG (hence, the name *Message Passing* algorithm).

The message that is propagated by *Belief Propagation* is a conditional probability among variable nodes in the FG. This message can be considered as a piece of information about what a factor node 'thinks' in respect to the state of its neighbour variable nodes, based on its defined function. It does this by taking all the messages from its neighbouring variable nodes (except the one it intends to send its message to) and then computes those messages based on the encoding function that factor node has. This local computation in that factor node becomes the new message the factor node sends to its destination variable node. Next, the variable node inspects all the incoming messages from its neighbouring factor nodes and calculates what it believes, before sending its message again to its destination factor node. This process will continue over-and-over again until no message is updated. In principle, this process is the same as the process we demonstrate in Section 2.4.4.

As mentioned, our interest is to capture the provenance of a product in order to model and understand its risk across its supply chain. Thus, *Belief Propagation* is in this context used to propagate the probability, which essentially is a message in the context of the *Message Passing* algorithm, to infer the risk. Since risk holds a notion of probability, our approach adapts the *Sum-Product* algorithm to propagate the distribution of probabilities by rearranging two basic rules, the *sum rule* and the *product rule*, according to the laws of probability, through the *Sum-Product* algorithm (see Section 2.4.1.2).

Using the *Sum-Product* algorithm to generate inferences from an FG is similar to finding a probability distribution in the variable nodes by conditioning on other variable nodes. In this case, the *product rule* is used to find the probability of occurrence of two independent events, and the *sum rule* is used to find the joint probability of the individual event. By using these two rules, our approach allows the propagation of the updated observed variable nodes to infer the unobserved ones. In the FG, an observed variable node is a variable node whose state is known with certainty, and an unobserved variable node is a variable node whose state is unknown.

4.5 Summary

The aim of this chapter was to introduce three scientific and systematic techniques to estimate a product's risk in its supply chain, as described by provenance. Overall,

prFrame comprises three major techniques, each of which have been discussed in the previous sections. Those are incorporating the risk models and their risk factors with the provenance of a product, the MC simulation based on a PG, and a conversion from a PG into an FG (a bipartite graph containing nodes for variables and factors). In the first technique, PROV can be used to model a product provenance, which can be seen as a product supply chain. This modelling captures the risk model and its risk factors to allow a quantitative risk assessment tool to estimate risk. The second technique uses the MC simulation to take into account the variation of randomly distributed risk factors to be propagated through mathematical models. This approach relies on the directed nature of PGs, and propagates the predicted output value along the edges of these graphs, according to the evidenced formulae of the risk models. The last technique is a conversion from PG to FG to allow the *Belief Propagation* technique to take observations of the output value in the product supply chain and thence to calculate the marginal probability distribution for each unobserved node, conditional on these observed nodes. *Belief Propagation* requires a notion of an FG to perform an inference algorithm efficiently by rearranging sum and product rules. We have also demonstrated that it can be easily derived from a PG via the conversion technique.

These techniques result in our framework called *prFrame* to reason, estimate and understand risk across the product supply chain, even where we have only partial knowledge of it. To conclude this chapter, with these three techniques, *prFrame* is potentially a good framework to assess risk. By representing the product supply chain in provenance format, our aim is to develop an analytical method to understand how risk is calculated and propagated across the actual product supply chain. Thus, *prFrame* should also consider all the dependencies and key criteria (e.g., structure of the food chain, representation of food characteristic, microbial sampling, etc.) that are relevant to develop analytical methods in assessing risk across the network of the product supply chain. Afterwards, we aim to implement a reasoning technique to estimate risk with a view to providing a tooling system to support due diligence.

Chapter 5

Food case study: *prFood*

Food is defined by the Food and Agriculture Organization (FAO) and the World Health Organization (WHO) as *any substance, whether processed, semi-processed or raw which is intended for human consumption, including drinks, chewing gum and any substance which has been used in the manufacture, preparation or treatment of 'food' but excluding cosmetics, tobacco, and substances used only as drugs* [54]. Since food is an essential need of our body, in principle, it should be safe, compositionally correct, not contain harmful contaminants, contain only permitted additives, correctly described, bear all necessary markings, and be labelled truthfully and processed safely.

Before food reaches its consumers, it is often circulated in different places, a process often referred to as the food supply chain. Here, we adopt a definition of the food supply chain as a systematic processing of food, consisting of all stages (processing, packaging, storage, distribution and retail, transport, handling, food preparation and consumption) from the on-farm production to the consumption in homes, restaurants and/or institutions (e.g., school, university, etc.) [84]. By describing these food processes, a lineage of food can be captured, ultimately providing an overview of the food supply chain. Since food should be processed safely, food safety can be defined as the efforts (in relation to handling, storing and preparing food) throughout the entire food supply chain to minimise the risk of contamination and to protect consumer's health [84][87][88]. Those definitions fit with the aim of the Food Safety Act (FSA) 1990 to protect consumer's health [87].

One aspect of complying with the FSA is taking reasonable precautionary actions (e.g., washing hands, temperature checking, documentation, etc.) in food production, distribution and handling. Those actions are important to keep food uncontaminated by any hazard such as microbial pathogens or chemical substances and should be controlled across the food supply chain [1]. Since hazards can be introduced accidentally or intentionally at any point in the food supply chain, it is necessary to anticipate food contamination by measuring its risk. In this case study, we attempt to measure the risk of microbial contamination quantitatively through the general framework proposed by

Nauta, called the Modular Process Risk Model (MPRM) [89], which we discuss in detail in Section 5.1.

When a high level of bacteria is found in food, it is important to investigate how food operators have handled that food at all points in the supply chain, i.e. food traceability. Food traceability is defined by European Union Law as *the ability to trace and follow a food, feed, food-producing animal or substance intended to be, or expected to be incorporated into a food or feed, through all stages of production, processing, and distribution* [17]. This allows food operators to identify the origin of food because traceability will keep the information about what the raw material or ingredients of food are right through to the final food product. By identifying what, where, when, who and how food has been handled, food operators have an overview of potential fraud or contamination in respect to any food item. For that reason, traceability of food is suggested to provide safer food supplies, thus minimising risk to consumers [18]. Food tracebility, therefore, is an inherent part of due diligence.

Traceability should not only record what food is received and from which supplier, but also what food is dispatched and to which customer. It should be able to explain the connection between inputs and outputs of food in the food supply chain. This is known as the *one-up/one-down* concept and must be implemented by the food operators. There are two terms often mentioned in this concept, tracing and tracking. On the one hand, tracing (an inherently backward looking task) is the ability to identify the origin of food; hence, known as an upstream or *one-step-up* process. On the other hand, tracking (an inherently forward looking task) is the ability to follow the downstream of the path in the supply chain; hence, known as a downstream or *one-step-down* process [90]. With the *one-up/one-down* concept, food operators and food enforcement authorities have visibility and transparency of where and how food is treated along the food supply chain, ultimately demonstrating due diligence.

With a view to minimising the risk of food contamination and to support the traceability and track-ability of food, various food regulations have been created [16][88]. One of these regulations is the 1995 Food Safety (General Food Hygiene) Regulation, which requires all Food Business Operators[1] to implement a Hazard Analysis Critical Control Point (HACCP)-like system [91]. With HACCP-like systems, all food operators involved in the food supply chain can review their food processes and analyse them to anticipate an undesirable outcome. A HACCP-like system within a food operator requires the details of food (its specification and how it should be handled) and the flow of food to be described [20]. Once the system is implemented, the parameters defined in the HACCP plan[2] must be checked. The main objective in implementing HACCP is to detect in a timely fashion if there is an increased risk of food contamination. Since food

[1] A food business operator is a natural or legal person responsible for ensuring that the requirements of food law are met within the food business under their control [17].

[2] A HACCP plan is a written HACCP-based procedure.

is traded globally, however, it can be a challenge to attain an overview of the risk over the whole food supply chain that involves many food operators [92][93].

5.1 Quantitative Microbial Risk Assessment (QMRA)

As mentioned earlier, our approach quantitatively assesses the risk of contamination by undesired bacteria across the food supply chain. We adopt Quantitative Microbial Risk Assessment (QMRA) as a model that can be performed in two steps: hazard identification and characterisation, and exposure assessment [94]. In the first step, the characteristics of undesired bacteria need to be studied to gain a better understanding of how, and in what conditions, they grow. The next step is an exposure assessment, which is defined as an interdependent process where the links from one process to the other should be clarified [94]. In this step, the lineage of food is drawn and each process involved is described to see the transmission of bacteria.

In QMRA, frameworks for exposure assessment are proposed to estimate the microbial load in the food supply chain quantitatively [95][96][97]. These frameworks have several limitations, however, such as not being easy to use as a general framework, a overly data-based approach, and not incorporating uncertainty in the model. Based on those limitations, we adopt the framework proposed by Nauta called Modular Process Risk Model (MPRM) [89]. MPRM is a process-driven framework to estimate the risk of food contamination based on how food is handled. This framework is specifically utilised as a general science-based statistical tool to model the transmission of undesired bacteria from one process to another process in the food supply chain. Figure 5.1 shows its schematic representation adapted from [89].

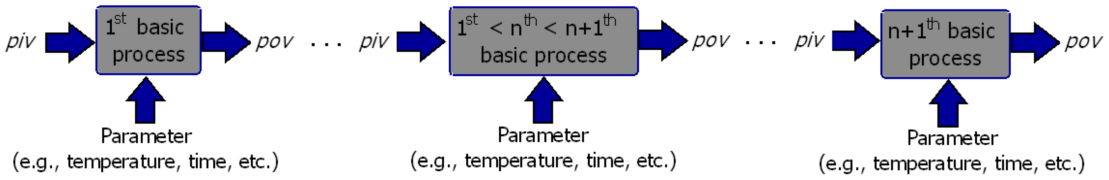


FIGURE 5.1: A schematic representation of MPRM.

In Figure 5.1, the food supply chain is shown be split into smaller processes or modules. For each module, the *input-output* relation for the bacterial load in food is calculated by a risk model that mimics the actual food process. The output (*pov*) of the previous process will be the input (*piv*) for the next process. In each process, the associated parameters (e.g., time, temperature, etc.) are set as the risk factors that determine the predicted output values (*pov*). The role of these risk factors is important to change the outcome of a process, which consequently affects the outputs of the next processes. Their form is often in a probability distribution to accommodate all possible values a

risk factor can have. The aim of this framework is to evaluate the health risk arising from undesired bacterial exposure and to identify paths leading to a higher number of bacteria. As a result, preventive actions can be performed by food safety authorities in those processes where a high level of undesired bacteria is present [86][98].

In MPRM, there are six basic processes that can affect the colonies of bacteria after a food process. These are Growth, Inactivation, Partitioning, Mixing, Removal and Cross Contamination [94]. Growth and inactivation are two basic bacterial processes, which are strongly dependent on the characteristics of the bacteria investigated and the surrounding environmental conditions. In each process, a variety of models that represent growth and inactivation can be applied; although it is suggested to use as simple a model as possible. The model selection depends on the statement of purpose, process knowledge, data availability and the alternative scenario considered [89]. If the transmission is too complex, or if essential parameters are unknown, a *black-box* model may be used.

Growth can be regarded as an increasing number of undesired bacteria over time due to the effect of risk factors in the food supply chain. The growth event depends on the risk factors, such as time, temperature, product shape and size, and strain of bacteria [78]. Growth does not make the fraction of the contamination unit bigger, however (see Table 5.1). A fraction of contamination unit is a physical area on which the colonies of bacteria lie in a unit of food (a.k.a. prevalence or the number of existing bacteria in a fraction of food). In contrast, inactivation is characterised by a decrease in the number of undesired bacteria in a food product [65]. As with growth, inactivation also depends on time and temperature. For example, cooking at low temperatures or for short cooking times can result in the survival of undesired bacteria. Even when food is cooked correctly, however, undesired bacteria can still survive and inactivation models are used to calculate the rate of this survival in different scenarios. As opposed to the growth module, inactivation can reduce the fraction of the contaminated unit (i.e., bacteria-free food means no fraction in food that is contaminated in Table 5.1).

Partitioning, mixing, removal and cross-contamination, meanwhile, are four handling processes that are assigned depending on how food is handled. Partitioning occurs when a major unit of food is slice up into several minor units, while mixing describes a process in which several minor food products are combined to form a major product. Mixing will result in a joined product with the overall population of micro-organisms being roughly the sum of the populations of the joined materials. Both mixing and partitioning do not change the total number of undesired bacteria. Note that Table 5.1 shows the number of undesired bacteria for the total amount of food ($N_{(tot)}$), not for a single food per se. An example of partitioning is cutting a slice of meat into two pieces. The colonies of bacteria are still the same before and after the cutting process (as long as no growth model is involved in the cutting process). Similarly, mixing two types of vegetables will not increase the colonies of bacteria before and after the mixing process.

In our non-linear structure in Table 4.1, partitioning is the same as $O2M_{dirE}$ and mixing is the same as $M2O_{dirE}$.

Since removal is a process where some units (or parts of units) are selected and removed from the production process, it reduces both the fraction and the total number of undesired bacteria in food. Finally, cross-contamination is described as the transferal of undesired bacteria from one object to another object. This module can increase the fraction of contamination because the undesired bacteria from other objects can inhabit different places within a single food. As a consequence, the total number of undesired bacteria can increase. The number of undesired bacteria transferred can vary depending on the interaction between the objects. Table 5.1 presents the aftereffects of each basic module taken from [89].

Basic Module	**Effect on P (the fraction of contaminated units)**	**effect on N_{tot} (the total number of cells over all units)**	**effect on unit size**
Growth	=	+	=
Inactivation	-	-	=
Mixing	+	=	+
Partitioning	-	=	-
Removal	-	-	=
Cross-contamination	+	=/+	=

TABLE 5.1: The basic processes of the MPRM and their qualitative effect on the prevalence (P), the number of the organism in all units (N_{tot}) and the unit size.

5.2 *prFood* Ontology

In order to understand food processes and analyse the risk around them, we need to model the food ecosystem and its processes across the food supply chain so as to be able to identify potential contamination problems. Our modelling approach entails the use of an ontology to model and capture knowledge about food and the risk of bacterial contamination, which comprises the second contribution in this research. In defining an ontology, we refer to a definition provided by Studer et al., namely: *Conceptualization refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. Explicit means that the type of concepts used and the constraints on their use are explicitly defined. Formal refers to the fact that the ontology should be machine-readable. Shared reflects the notion that an ontology captures consensual knowledge, that is, it is not private of some individual, but accepted by a group* [99]. With an ontology, a common understanding of the domain of interest can be achieved in a standardised format and, thereby, data exchanges become easier.

Modelling food and its supply chain can be done with PROV and its ontology, PROV-O, which can articulate the processes of food in the food supply chain over time. We believe that PROV is a suitable framework for food provenance because of its ability to

describe the entities, agents and activities that may have influenced each piece of data about food. It should provide an *ease-use* for food operators and authorities to comply with the food requirements through food provenance. Thus, we define food provenance as in Definition 5.1.

Definition 5.1. Food provenance can be defined as a record that describes a food product and its ingredients, the processes involved in food transformation, and the organisations that are responsible for those processes from the source to consumption.

We introduce our ontology, *prFood*, which models and captures food, including its specification, distribution and transformation. *prFood* supports risk assessment for checking/auditing, verification, and further investigation. Together with *prFrame* ontology, it is designed to capture the lineage of food and the information regarding food processes and the risk of bacterial contamination necessary to compute risk. In capturing this information, *prFood* will identity each process and associate it with a risk model and risk factors. Also, a set of food regulations, which regulates the appropriate handling of that food, is captured in a process where it is applied. That information is modelled as parameters to allow risk models to calculate the risk in those processes.

Overall, the task of modelling the history of food is one that attempts to capture the lineage of food and its associated risk. It gives some benefits, such as: 1) Providing visibility or transparency of the interconnected food supply chain, 2) Providing crucial insights, such as identifying critical food processes and 3) Assessing downstream impact or tracing faults upstream. In developing *prFood*, we extend PROV-O to enable risk calculation in the food domain. Overall, the above benefits can be deemed as demonstrating due diligence and are delivered by *prFood* through the *prFrame* framework. As an ontology, *prFood* supports due diligence to conceptualise and modelling a specific domain, providing a basic concept to perform monitoring and reasoning. This is a fundamental concept in taking decisions to demonstrate due diligence. Before we develop *prFood*, however, in the following sections we identify and discuss several requirements and design principles arising from food regulations.

5.2.1 Domain requirements

The first requirement we are concerned with is the traceability of food. To comply with Regulation (EU) 178/2002 on the principles of food and feed traceability, all food operators need to retain documentation regarding relevant transactions. These should include the details of the sold food products, the suppliers and the customers. This essentially establishes the concept of *one-up/one-down* according to the regulation. A *one-up/one-down* concept is a concept in tracing and tracking food to explain the connection between inputs and outputs of food a food operator treats. Food provenance

potentially allows us to do more than *one-up/one-down* because it can help each operator in the provenance to identify the source of food it receives, and the destination of food it sends.

The second requirement we have identified is the nutritional information of food. Regulation (EU) 1169/2011 about food information provided to consumers requires food operators to provide nutritional information to their consumers, having special regard for consumers with special dietary requirements. To comply with this regulation, *prFood* should be able to check the compositional ingredients of nutritional information in a food product. Besides the nutritional information, some other mandatory pieces of information should also be passed on to the consumers. These are the name of the food, the list of ingredients and their quantities, allergen or intolerance information, the quantity of food, date of minimum durability, storage conditions and/or conditions of use, the name and address of the food operators under whose name the food is being marketed, the country of origin and instructions for use.

The next requirement is to allow one to check the temperature in food processes according to Regulation (EC) 852/2004, Regulation (EC) 853/2004, and The Food Hygiene Regulations, 2006. The growth of bacteria is highly affected by temperature in the processing, manufacturing, handling and distribution of food, including retailers and caterers. Checking the temperature used in the food processes can therefore help food operators to comply with these regulations.

Our final requirements is continuous monitoring, by which we mean monitoring performed on a timely basis. A continuous monitoring procedure is preferable as it helps ensure that produced products have met the acceptance criteria. This can be achieved by implementing a HACCP-like system when handling food. According to the 1995 Food Safety (General Food Hygiene), food operators are required to check the instructions or the HACCP description in the food product as a way to comply with this requirement.

5.2.2 Ontology design principles

In developing an ontology, the food concepts are defined as classes and properties in *prFood*. To define these, *prFood* uses the prefix prov to denote URI http://www.w3.org/ns/prov# and the prefix food to denote URI https://provenance.ecs.soton.ac.uk/food#. Based on this principle, the PROV core concepts (i.e., prov:Entity, prov:Activity, and prov:Agent) are described as classes in *prFood*, while the relations in the PROV (e.g., prov:used, prov:wasGeneratedBy, etc.) are represented as properties. In addition, some classes are organised into a super- or sub-class hierarchy. As an example, food:Product is a superclass of food:SampleProduct, implying that anything that is a food:SampleProduct is also a food:Product. The same principle also applies to the property in ontology (e.g., food:specialisationOf).

On the basis of the food requirements in the previous section, we identify several principles in developing an ontology that models food and its lineage to allow computation of risk based on the identified risk model and risk factors. First, the core requirement of traceability in the food supply chain means that provenance serves as a backbone in designing *prFood*. Since *prFood* is an extension of PROV-O, the food concept (e.g., food, bacteria, sampling report, etc.) are defined as the subclasses of prov:Entity and the processes related to food (e.g., analysing, cooking, storing, etc.) as the subclasses of prov:Activity. Moreover, the food operators (e.g., laboratory, retailer, inspector, etc.) are defined as subclasses of prov:Agent.

The second principle is to be able to perform an automatic risk assessment. To achieve this, some properties are captured as the parameters for risk calculation. These parameters are, basically, the risk models and the risk factors. The distributions of the risk factors are treated as datatype properties, which allow the risk models to take them as the inputs for automatic risk calculation. The next principles are based on the shareability and independence. *prFood* provides a design that allows other ontologies to be integrated through the mapping procedure. As a result, although the legacy food ontologies are developed independently from *prFood*, this does not prevent us from using them to support risk calculation.

5.2.3 Application design principles

From the application point of view, *prFood* is developed to convey the notion of provenance in the food domain. In this section, we provide some mathematical representations of the description logic between vocabularies in *prFood*. Each activity within a food process (prFood:FoodProcessing) will be assigned through prFood:hasBasicModule with one (or more) of six basic modules (prFood:BasicModule) adopted from MPRM, the risk framework to estimate the risk of bacteria after a food process. The property of prFood:hasBasicModule, therefore, has a universal restriction between class prFood:FoodProcessing and prFood:BasicModule in *prFood*, as shown in 5.1.

$$\forall hasBasicModule.FoodProcessing. \tag{5.1}$$

Along the food supply chain, a food product is transformed from one stage into another stage by a food process. Since a food product can only have one stage at a particular time (e.g., a food product cannot be in both stored and prepared stages simultaneously), some sub-classes of food:Product hold union classes.

$$\begin{aligned} ProcessedProduct \sqcup TransportedProduct \sqcup RetailedProduct \sqcup \\ StoredProduct \sqcup PreparedProduct \sqcup CookedProduct \end{aligned} \tag{5.2}$$

In the prFood:BasicModule, two sub-classes that are connected through a union relation on the basis that they cannot happen at the same time are prFood:Growth, when the microbial load is increased, and prFood:Inactivation, when the microbial load is decreased. Those sub-classes, as explained, are the microbial conditions in MPRM; thus, they hold a union relation as follows:

$$Growth \sqcup Inactivation \tag{5.3}$$

The other basic modules are the environmental conditions that affect the growth or reduction of bacterial load. Growth of microbial can occur because of cross-contamination and mixing of products. Therefore, prFood:Growth, prFood:CrossContamination, prFood:Mix are connected to each other through an intersection relation as follows:

$$Growth \sqcap CrossContamination \sqcap Mix \tag{5.4}$$

In contrast, the reduction of microbial (prFood:Inactivation) is driven by the slicing or removal of a food product. Thus, prFood:Inactivation, prFood:Partition, prFood:Removal also hold an intersection relation with each other as follows:

$$Inactivation \sqcap Partition \sqcap Removal \tag{5.5}$$

As designed, all classes described in *prFood* are subclasses of either prov:Entity, prov:Activity, or prov:Agent. Meanwhile, the object properties (the properties between objects) in *prFood* are described in the verbal form of the past tense to express the past evidence associated with food. We also define data properties (the link properties between object and literal) to capture the data about food, risk models or risk factors. We also adapt several properties from PROV-O, such as prov:wasDerivedFrom, prov:wasGeneratedBy, and prov:used to capture explicitly how a certain came about or where a certain food came from. This makes provenance an important aspect of *prFood*. For instance, the transformation of food from stored food into cooked food can be modelled in a triple as follows:

prFood:CookedFood *prov:wasDerivedFrom* *prFood:StoredFood*

All classes of food processes are defined in *prFood* (prFood:FoodProcessing). These classes will have some data properties to detail how the processes are performed with a view to subsequent investigation should a contamination outbreak occur. For example, prFood:HaccpInformation is a data property that explicitly describes the HACCP plan in the food processes. In addition, class prFood:FoodProcessing is linked to class prFood:Parameter, which explicitly captures all parameters, such as the duration and temperature. The class prFood:MicrobialModel captures the mathematical formulation to

estimate the number of bacteria in food after each food process is performed. The calculation formula depends on the type of bacteria (prFood:Bacteria), the MPRM basic module (prFood:MprmModule), and the parameters or the risk factors (prFood:Parameter). These mathematical functions are the risk models, which were formulated by the experts to best represent how the colonies of bacteria develop according to the basic modules.

To be able to explain the specification of food, we provide class prFood:FoodSpecification, which is a superclass of prFood:AllergenInformation, prFood:NutritionalInformation, etc. Each of the subclasses is attributed to the data properties that capture the related information about the details of food. For example prFood:energy is a data property in class prFood:NutritionalInformation that captures the amount of energy in food.

Based on those requirements and principles, Figure 5.2, Figure 5.3, Figure 5.4, and Figure 5.5 show the class diagrams of *prFood*. Due to limited space, we separate a single diagram into those figures that capture the subclasses of prov:Entity in Figure 5.2 and Figure 5.3, subclasses of prov:Activity in Figure 5.4, and subclasses of prov:Agent in Figure 5.5. In general, Figure 5.2 shows that each food (prFood:Food) contains several information, such as nutritional, diet, and allergen information; and that information is captured as the subclasses of prFood:FoodSpecification. To capture the different stages of food during its lifetime, the prFood:FoodTransformed class is created and it has several food stages as its subclass (e.g. prFood:RetailedFood, etc.) as shown in Figure 5.3. This transformed food can be linked with a suitable subclass of prFood:FoodProcessing in Figure 5.4 to indicate the result of a certain food process. For instance, food prFood:RetailedFood is a result after the process of prFood:Retailing. In addition, each food process can be linked to any subclasses of prFood:MprmModule (Figure 5.2) to capture the potential risks in that food process. With this simple procedure, *prFood* ontology allows us to construct or model a sequence of the food processes and food stages, ultimately representing the food supply chain. Moreover, we can include the operators (e.g., food business, person, etc.) of food process by linking a suitable class in Figure 5.5.

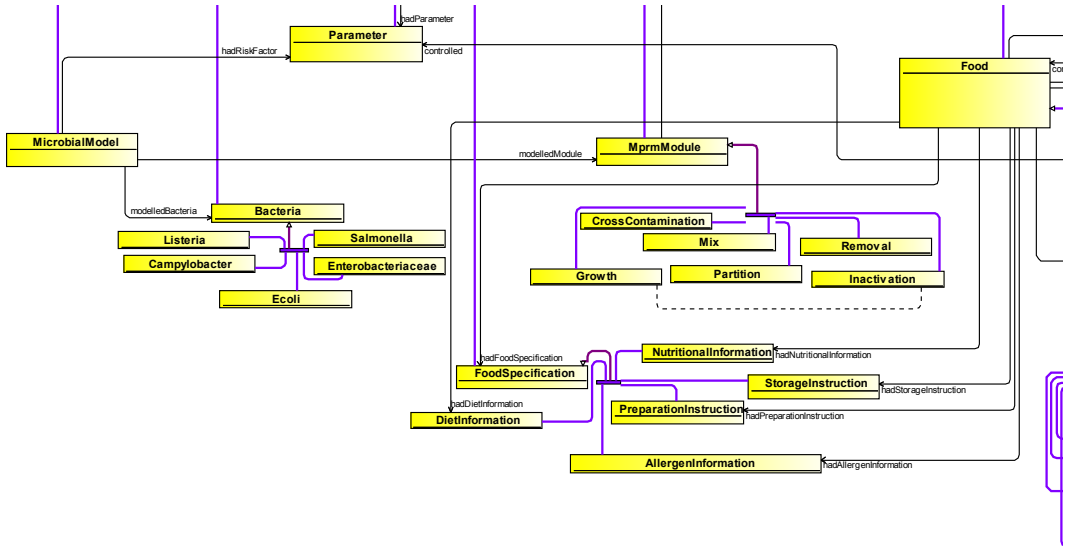


FIGURE 5.2: A class diagram of *prFood*(a): subclass of prov:Entity.

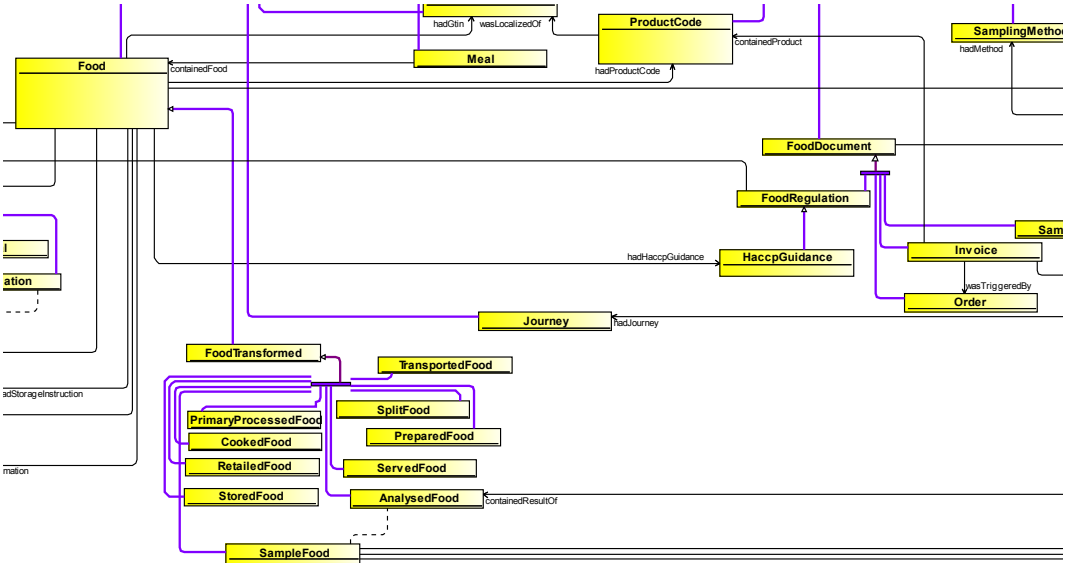


FIGURE 5.3: A class diagram of *prFood*(b): subclass of prov:Entity.

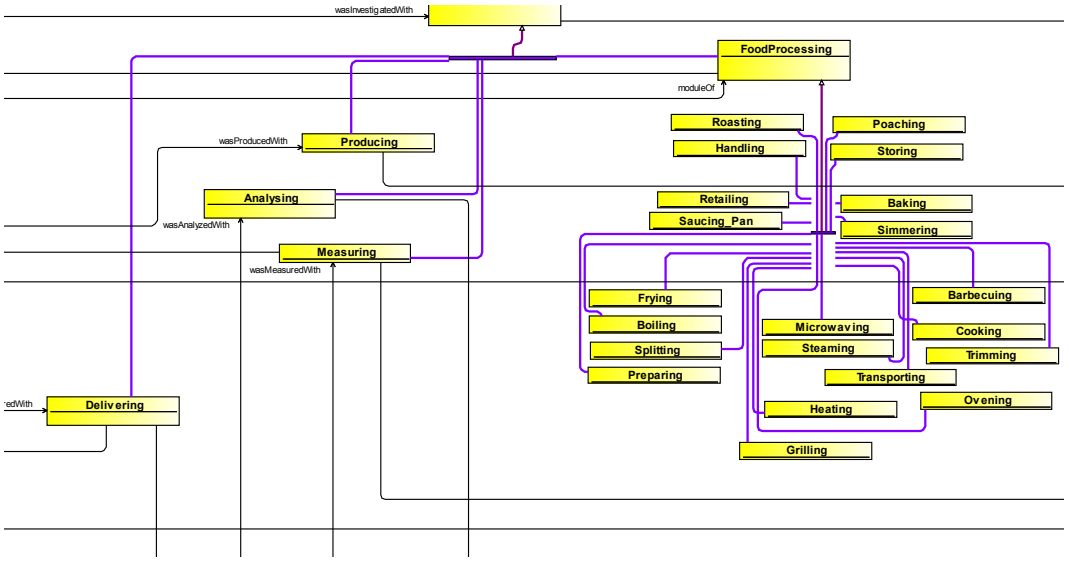


FIGURE 5.4: A class diagram of *prFood*(c): subclass of prov:Activity.

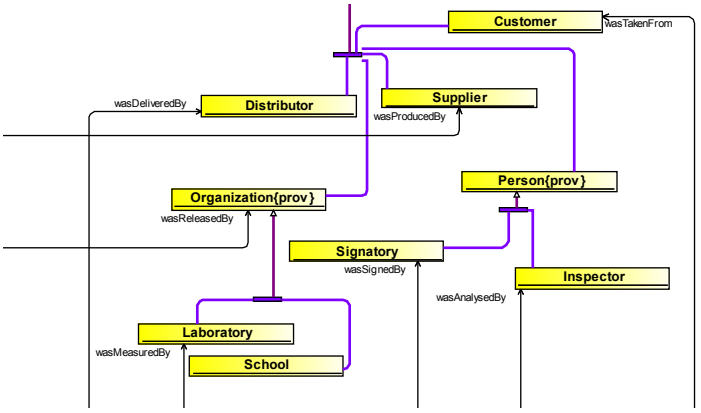


FIGURE 5.5: A class diagram of *prFood*(d): subclass of prov:Agent.

5.2.4 Mapping the legacy ontologies

Many food ontologies already exist, and our mechanism for risk calculation is more powerful if we can easily extend it into specialised food sub-domains. We therefore provide a mapping mechanism by which we can map the classes and properties in *prFood* to equivalent classes and properties in each legacy food ontology. This process allows SPARQL Protocol and RDF Query Language (SPARQL) queries to exploit the legacy food ontologies via our mapping ontology *prFoodMapping*.

Many ontologies in the food domain have been developed to model food knowledge. Pizzuti et al., summarise the ontologies in the domain of food, then propose their ontology to trace and track a food product, known as the Food Track&Trace Ontology, or FTTO [100]. The FTTO ontology aims to integrate and connect the main features of the food traceability domain with the Global Track&Trace System. To achieve their

goal, FTTO includes the representative food concepts in a supply chain, such as the service products, processes and actors involved, within a single hierarchy. The notion of food contamination is missing from FTTO, however. Thus, although one can trace the contaminated food, it is difficult to explain what might be the cause of contamination.

An ontology developed by Markovic et al. aims to monitor food safety by documenting constraints that may be associated with a HACCP plan [40]. Data about time and temperature are captured within their ontology to monitor how meat is treated in the kitchen as a means to provide an alert system if regulations are abused. In addition, their work also extends other existing ontologies with a food safety ontology to model HACCP-based food preparation. Markovic et al.'s constraints are limited to the cooking process in the kitchen alone, however. Consequently, other food processes, such as transportation and storage are not modelled, making it difficult to describe what happened with food before the cooking process.

In a similar vein, Xiangwei and Lin developed an ontology to model HACCP knowledge in a food cold chain[3] [101]. Although their ontology models the information in multiple processes, it is limited to the processes that maintain a low temperature (cold chain). Thus, processes involving a high temperature (cooking), or cross-contamination between food products, are not modelled in this ontology. In addition, as a cold chain is often performed by a food enterprise, this ontology does not focus on the consumer at home, where most of the food contamination is likely to occur [102][103][104].

Overall, most of the legacy food ontologies lack knowledge about quantifying harmful bacteria as a way to assess the risk of contamination, which is a key difference from *prFood*. Moreover, these previous food-related ontologies are independent from each other and already specific for use in the food domain. In contrast, although *prFood*, is also specific to the food domain, it is rooted in PROV-O as its general ontology to describe the provenance of the general things. Expanding PROV-O by adding *prFood* as an extension therefore situates provenance as the backbone of our ontology, rather than the food domain itself. This makes our overall framework much more generalisable and is the main reason to develop *prFood* as an extension of PROV-O, rather than adding or extending previous food-related ontologies.

In developing *prFoodMapping* ontology, we define classes in *prFoodMapping* as the classes from *prFood* ontology. Therefore, the subclasses of *prFoodMapping* are equivalent with classes in the legacy ontologies they have mapped into. For instance, prFood:Food has `http://www.cookReg.org/cook#Meat` as a subclass. Here, cookReg:Meat is an equivalent food with prFood:Food; hence, when we query all classes that have the same type as prFood:Food, it will return all food from another legacy ontology that is a subclass of prFood:Food. Figure 5.6 shows the principle design of *prFoodMapping* ontology.

[3] A cold chain often refers to a process whereby food is chilled or frozen so as to maintain its safety and quality until it arrives at consumers.

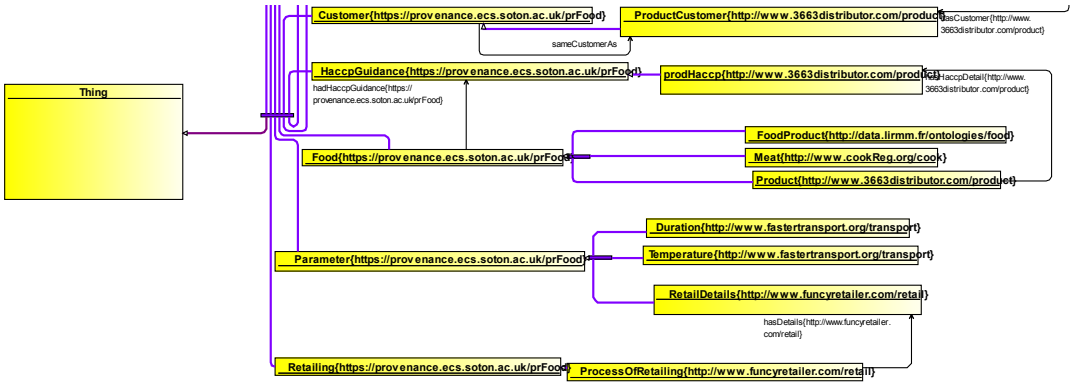


FIGURE 5.6: A class diagram of *prFoodMapping*.

5.2.5 Evaluation of *prFood*

In this section, we present several use cases to reflect the requirements presented in Section 5.2.1. The use cases are also based on the real requirements from EU and UK food regulations. Each use case is validated by SPARQL queries to answer the requirements, validating the design of our ontology *prFood*.

Use Case 1: Can we establish the chain of sellers and buyers for a food product?

Overall, this use case aims to check whether the food operators are retaining their transactional documents in compliance with food regulations. To achieve that, we identify the relevant transactional documents (e.g., invoice, order, etc.) that contain information about the distributors (sellers) and customers (buyers) of the food products. By exploring transactional documents, we are able to build a chain of sellers and buyers of a food product. To extend the result, we also identify all transactional documents from other food legacy ontologies. These documents are the equivalent documents to those captured by *prFood*; thus, they are the subclasses of the transactional document from *prFood* in *prFoodMapping*. Listing 5.1 and Listing 5.2 show the SPARQL queries to answer our first use case.

```
PREFIX prFood:<https://provenance.ecs.soton.ac.uk/food#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT DISTINCT *
WHERE {
    ?invoice    rdf:type                prFood:Invoice   ;
                prFood:wasReleasedBy    ?distributor ;
                prFood:wasTriggeredBy   ?order .
    OPTIONAL { ?order   rdf:type                prFood:Order ;
                        prFood:wasReleasedBy    ?customer . }
}
```

LISTING 5.1: A SPARQL query to return the details of the invoices with *prFood*.

invoice	distributor	order	customer
prFood:INV74683Pro	prFood:3663	prFood:ORD7459	prFood:SDNLat2

TABLE 5.2: A result of SPARQL in Listing 5.1.

In Listing 5.1, we query the instances of class prFood:Invoice and try to find its distributors and customers. To find the distributors of the invoice, we use prFood:wasReleasedBy, which is a subclass of prov:wasAttributedTo to indicate an activity that associates with the food operators. To find the customer of the invoice, we query the order through prFood:wasTriggeredBy, and, subsequently, query the customer of that order by using prFood:wasReleasedBy. The result of Listing 5.1 is shown in Table 5.2, where it shows an instance of prFood:Invoice, prFood:Distributor, prFood:Order, and prFood:Customer.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX prFood:<https://provenance.ecs.soton.ac.uk/prFood#>

SELECT DISTINCT *
WHERE {
    ?invoice rdf:type prFood:Invoice   .
    OPTIONAL{
        ?invoice    prFood:wasReleasedBy    ?distributor ;
                    prFood:wasTriggeredBy   ?order .
        OPTIONAL {  ?order  rdf:type    prFood:Order ;
                                        prFood:wasReleasedBy ?customer . }
    }
}
```

LISTING 5.2: A SPARQL query to return the invoices legacy food ontology via mapping.

invoice	distributor	order	customer
prFood:INV74683Pro	prFood:3663	prFood:ORD7459	prFood:SDNLat2
out3663Product:Invoice3663_5654			

TABLE 5.3: A result of SPARQL in Listing 5.2.

In Listing 5.2, we perform the mapping procedure in which we query all subclasses of prFood:Invoice in *prFoodMapping* (the mapping ontology). These subclasses are the equivalent classes of prFood:Invoice in the legacy food ontologies. In *prFoodMapping*

ontology, http://www.3663distributor.com/product#Invoice3663 is a subclass of https://provenance.ecs.soton.ac.uk/prFood#Invoice (prFood:Invoice). Therefore, line ?invoice rdf:type prFood:Invoice in Listing 5.2 will return all instances of class http://www.3663distributor.com/product#Invoice3663 (i.e., out3663Product:Invoice3663_5654 in Table 5.3). However, the empty cells in Table 5.3 indicate that we simply have not mapped the prFood:Distributor, prFood:Order, and prFood:Customer with their equivalent classes in the ontology http://www.3663distributor.com/product.

Use Case 2: Does an FBO provide consumers with nutritional information

In this use case, we query the nutritional information about the marketed food products. In most cases, the description of the food product comprises the characteristics of the product, including nutritional information. With the same procedure as in Use Case 1, we query *prFood* to find all information about the food product, including its nutritional information. Listing 5.3 shows the SPARQL queries for Use Case 2.

```
PREFIX prFood:<https://provenance.ecs.soton.ac.uk/prFood#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX prov: <https://www.w3.org/ns/prov#>

SELECT  DISTINCT *
WHERE {
    ?food   rdf:type                              prFood:Food ;
            prFood:hadNutritionalInformation      ?nutInfo ;
    OPTIONAL {
        ?nutInfo    rdf:type                 prFood:NutritionalInformation ;
                    ?componentOfNutInfo      ?valueOfNutInfo . }
    FILTER (
        (?componentOfNutInfo  != prFood:NutritionalInformation) &&
        (?componentOfNutInfo  != rdf:type) &&
        (?componentOfNutInfo  != owl:topObjectProperty) &&
        (?componentOfNutInfo  != prov:specializationOf) &&
        (?componentOfNutInfo  != prFood:Information)
    )
}
```

Listing 5.3: A SPARQL query to return nutritional information with *prFood*.

Listing 5.3 shows the SPARQL query to a class that represents the food product (prFood:Food) and has information about its nutritional information (prFood:NutritionalInformation) and its result is shown in Table 5.4. Table 5.4 shows 3 different food, prFood:AnekaRujak, prFood:BatagorCingur, and prFood:CucurKukus. Each of them has an prFood:NutritionalInformation, which contains information about outLirmmFood:calciumPer100g, prFood:energyKCalories, and prFood:energyKJoules (as the datatype properties of class prFood:NutritionalInformation). Note that one of the datatype properties is outLirmmFood:calciumPer100g, which is information from external ontology we incorporate in *prFood* ontology.

food	nutInfo	componentOfNutInfo	valueOfNutInfo
prFood:AnekaRujak	prFood:NutInf_AR	outLirmmFood:calciumPer100g	"50"^^xsd:decimal
prFood:AnekaRujak	prFood:NutInf_AR	prFood:energyKCalories	"450"^^xsd:decimal
prFood:AnekaRujak	prFood:NutInf_AR	prFood:energyKJoules	"300"^^xsd:decimal
prFood:BatagorCingur	prFood:NutInf_BC	outLirmmFood:calciumPer100g	"75"^^xsd:decimal
prFood:BatagorCingur	prFood:NutInf_BC	prFood:energyKCalories	"600"^^xsd:decimal
prFood:BatagorCingur	prFood:NutInf_BC	prFood:energyKJoules	"500"^^xsd:decimal
prFood:CucurKukus	prFood:NutInf_CK	outLirmmFood:calciumPer100g	"85"^^xsd:decimal
prFood:CucurKukus	prFood:NutInf_CK	prFood:energyKCalories	"550"^^xsd:decimal
prFood:CucurKukus	prFood:NutInf_CK	prFood:energyKJoules	"200"^^xsd:decimal

TABLE 5.4: A result of SPARQL in Listing 5.3.

Use Case 3: For each food process, can we see the actions and corrective actions deemed to be appropriate in the HACCP guidance?

In this use case, we query the details about the proper handling of food by food operators, and what they should do in case of an unexpected event. These details are described in the characteristics of a product, and are, most of the time, coherent with HACCP guidance as a standard regulation to protect food from being contaminated. Thus, having a description of the product will explain how food operators should handle and process the product.

We begin by querying all food that are a subclass of prFood:Food. Since we have map an equivalent of prFood:Food from external ontology `http://www.3663distributor.com#product` in our *prFoodMapping* ontology, Listing 5.4 will return the food from the external ontology, `http://www.3663distributor.com#product` too. This is similar with our query in Use Case 1. Once the food that have with HACCP guideline are identified, Table 5.5 is presented as a result of the query. Note that the food and its HACCP guidelines in Table 5.5 are stored as an instance of food in external ontology `http://www.3663distributor.com#product`.

```
PREFIX prFood:<https://provenance.ecs.soton.ac.uk/prFood#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX o3: <http://www.3663distributor.com/product#>

SELECT DISTINCT *
WHERE {
   ?foodEnt     rdfs:subClassOf     prFood:Food .
   ?subHaccp    rdfs:subPropertyOf  prFood:hadHaccpGuidance .
   ?foodIns     rdf:type            ?foodEnt ;
   ?haccpEnt    ?pred               ?haccpIns .

   FILTER (
        (?subHaccp != prFood:hadHaccpGuidance) &&  (?pred != rdf:type)
    )
}
```

Listing 5.4: A SPARQL query to return HACCP details with *prFood* and other legacy food ontologies.

foodEnt	subHaccp	foodIns	haccpEnt	pred	haccpIns
o3: Product	o3: hasHaccpDetail	o3: Product1	o3: HaccpProduct1	o3: HaccpProduct1	"Haccp details for product 1 in 3663 distributor."

Table 5.5: A result of SPARQL in Listing 5.4.

Use Case 4: Can an FBO provide a food enforcement authority with the temperature details?

In this use case, we present how to query the risk factors (temperature and duration) relevant to a retailer's handling of food. Although in this use case we only query the risk factors in the retailing process, it would work in the same way for the other food processes too (e.g., cooking, transporting, etc.). We begin by identifying the class that represents the process in the food supply chain from prFood (e.g., prFood:Retailing). Subsequently, we explore the parameters of the process through the class property. Finally, we explore the risk factors associated with that process through temperature and time properties.

```
PREFIX prFood:<https://provenance.ecs.soton.ac.uk/prFood#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT DISTINCT *
WHERE {
        ?retailing  rdf:type            prFood:Retailing ;
                    prFood:hadParameter ?retailingParameter .

    OPTIONAL {?retailingParameter prFood:duration           ?durationValue .}
    OPTIONAL {?retailingParameter prFood:deviceTemperature  ?deviceTempValue .}
}
```

Listing 5.5: A SPARQL query to return the risk factors from *prFood*.

As we can see in Listing 5.5, we query a class with type prFood:Retailing. This class represents the retailing process in a retailer. Subsequently, we query the parameter

retailing	retailingParameter	durationValue	deviceTempValue
prFood:RetailingProsod1	prFood:ParameterRetailingProsod1	"52"^^xsd:decimal	
prFood:RetailingProsod2	prFood:ParameterRetailingProsod2	"90"^^xsd:decimal	

TABLE 5.6: A result of SPARQL in Listing 5.5.

(prFood:Parameter) used in the retailing process through the property prFood:hadParameter. After we receive the class prFood:Parameter, we explore the duration and temperature used in the retailing process. From Table 5.6, we can see that the process of prFood:RetailingProsod1 and prFood:RetailingProsod2 only have information about the duration and not the temperature of *retailing* process. In this case, the main reason is that the risk model in both prFood:RetailingProsod1 and prFood:RetailingProsod2 does not take temperature as a risk factor when calculating the number of bacteria in those processes.

5.3 *prFood* in *prFrame*

prFood is a domain-specific ontology that is extended from the PROV-O ontology to conceptualise the provenance of food. With *prFood*, the food ecosystem (e.g., production, distribution, consumption, etc.) can be modelled, ultimately representing the provenance of food. We have exercised *prFood* with our limited dataset about food specification, food procurement and food sampling reports; and successfully modelled the food ecosystem to capture the available information from the dataset.

In addition, *prFood* is also able to capture the risk models and the risk factors. This is done by providing risk-related vocabularies that conceptualise risk in food. At this moment, *prFood* only focuses on the risk of contamination; however, other food-related risks can be added later. With *prFood*, it is possible to capture the risk models and their associated risk factors and integrate them alongside the product provenance. This is because *prFood* is an ontology extension from PROV-O, meaning that most of the vocabularies in *prFood* are sub-vocabularies of PROV-O. With this approach, the notions of provenance and risk can be integrated and implemented within *prFrame* as a general framework to support due diligence.

To give the readers an overview of the food ecosystem from our dataset, we present the conceptualisation of the food ecosystem in the general graphs in Figures 5.7, 5.8 and 5.9. Basically, these graphs are the PGs that show the characteristics of food, its microbial sampling reports and its transactional records. We remind the readers that these PGs are constructed as a template that aims to instantiate the provenance records of food with the actual data. For example, name: productABC is an instantiate value of the property of prov:Entity prFood:productCode in Figure 5.7. We refer the

readers to `https://provenance.ecs.soton.ac.uk/prov-template/` for the details of the Provenance Template.

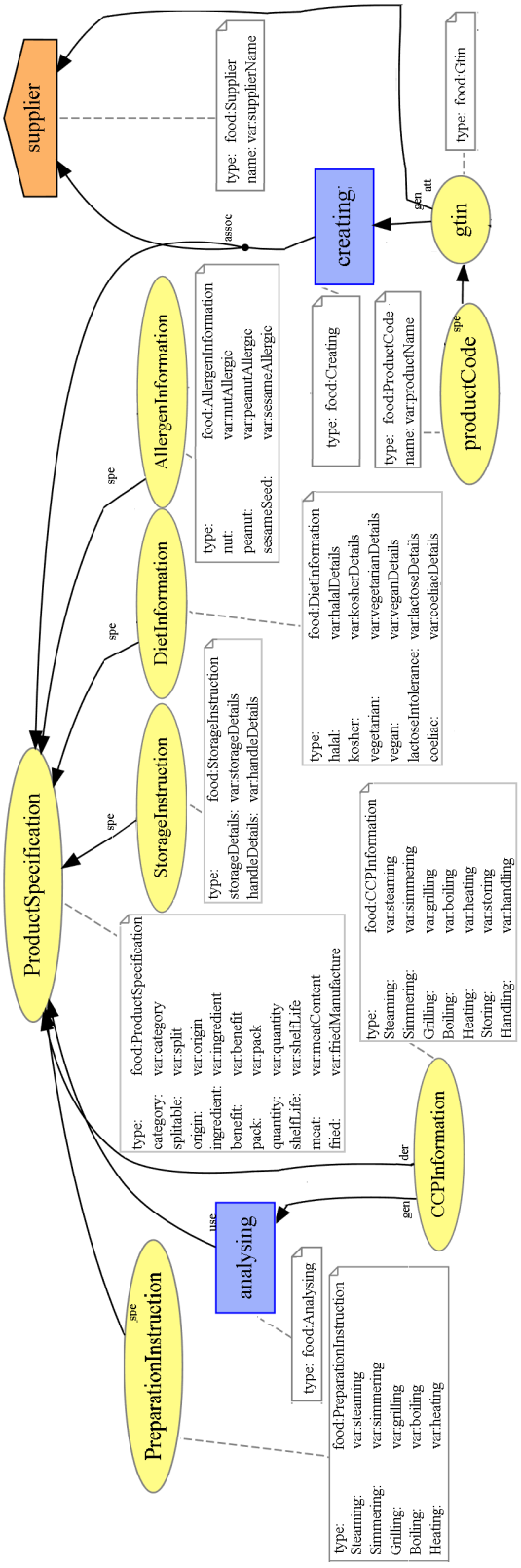


FIGURE 5.7: The provenance template of a food specification.

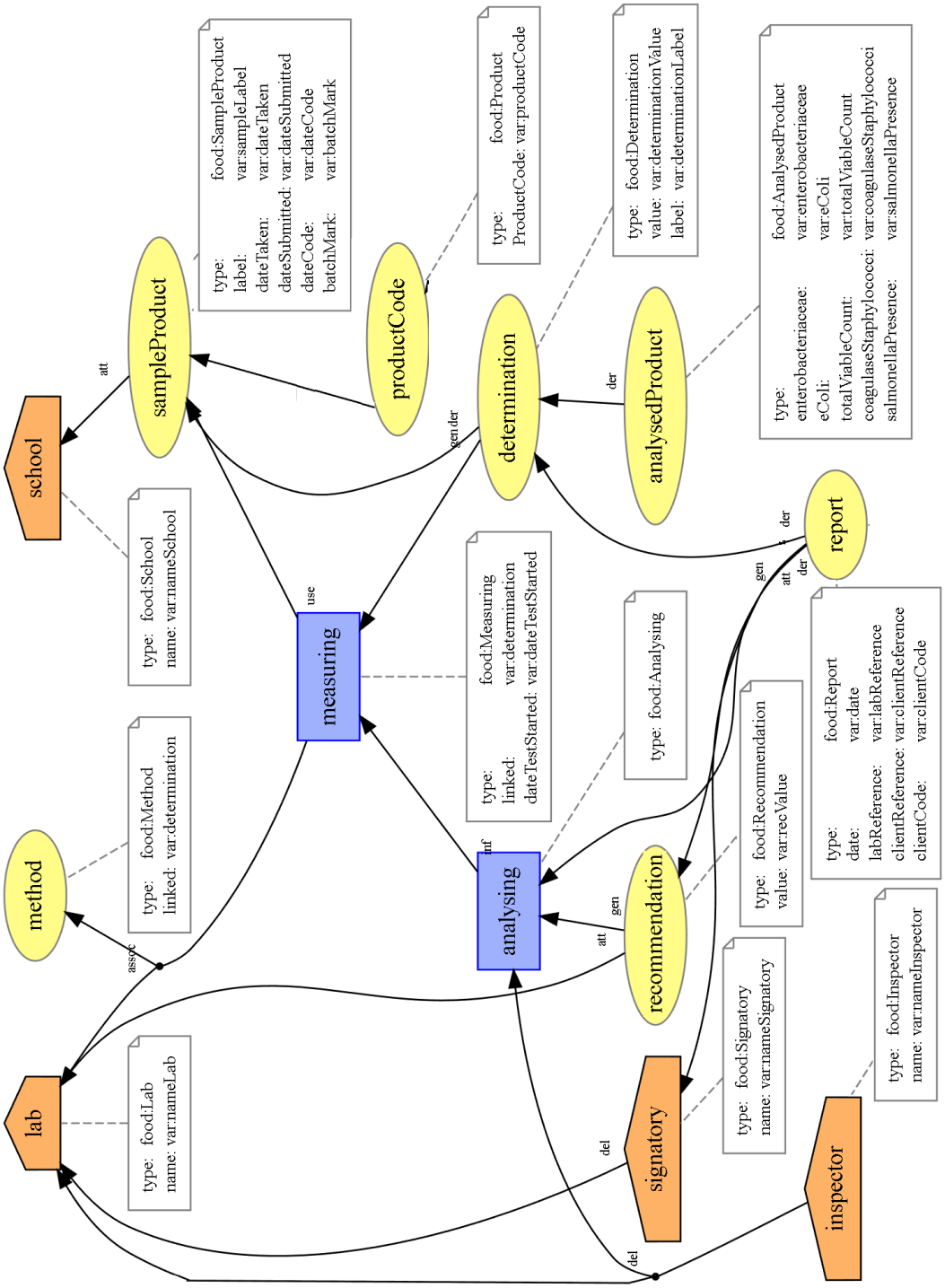


FIGURE 5.8: The provenance template of a microbial sampling report.

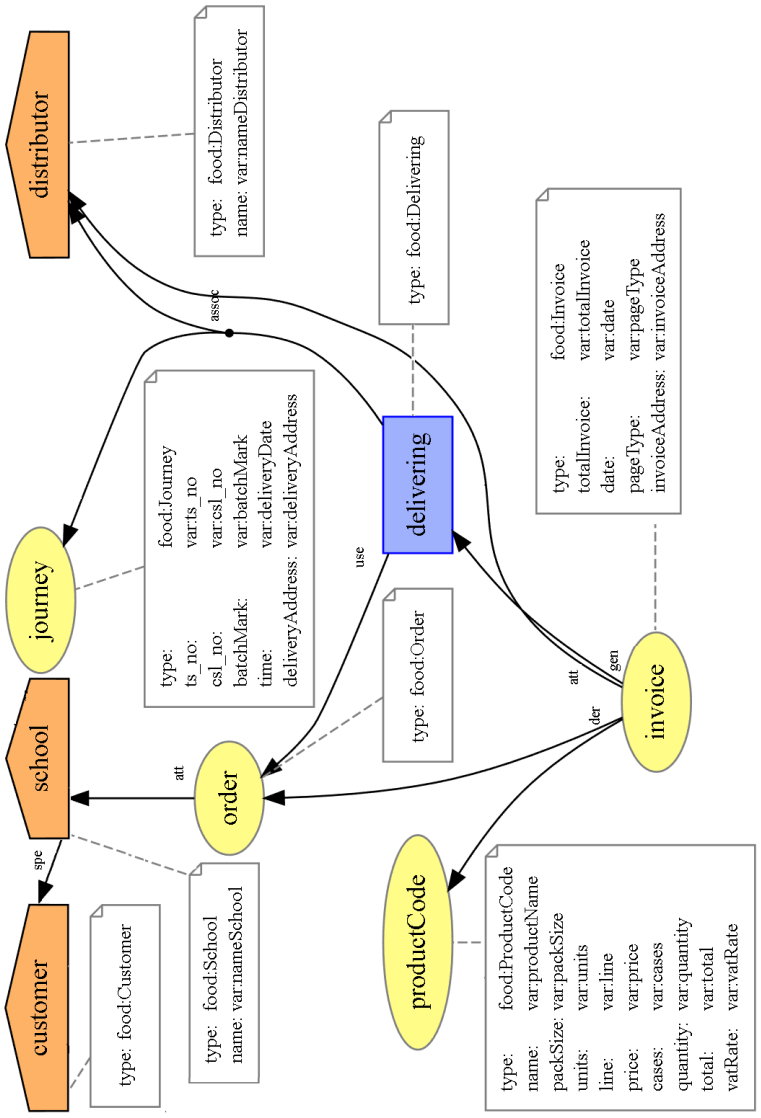


FIGURE 5.9: The provenance template of a food invoice.

Figure 5.7, Figure 5.8 and Figure 5.9 show three provenance templates that capture the type and the relations between food elements. The template in Figure 5.7 aims to describe a food product and its characteristics. The characteristic of food is described by a product specification (prFood:ProductSpecification) that explains the composition of food (prFood:AllergenInformation), intended use of a product (prFood:DietInformation), and how the product should be handled (prFood:PreparationInstruction and prFood:StorageInstruction). In addition, prFood:CCPInformation provides the information about CCPs on the basis of the product's specification. The CCP information is needed in provenance as a way to understand how the food product should be handled to eliminate the presence of micro-organisms.

The template in Figure 5.8 captures the results of microbial sampling, and the agents involved, in the form of a report (prFood:Report). We define several concepts that are

related to microbial sampling, such as prFood:Method (method used in sampling) and prFood:Determination (Microbial determination). Figure 5.9 depicts the distribution of the food product. This template captures the agents that are involved in the food distribution, such as the customer (prFood:Costumer), school (prFood:School) and distributor (prFood:Distributor). In order to understand about potential contamination in the delivery of food, it is necessary to identify the details about the food journey (prFood:Journey), such as temperature the food product experienced, batch mark, location, etc.

There are also several relations used to describe the link between food entities in those templates (See Table 2.2, the definition of relations). Those relations are presented below.

EXAMPLE of GENERATION In Figure 5.7, prFood:Gtin prFood:wasGeneratedBy prFood:Creating, meaning that the existence of GTIN is a result of a creating activity.
EXAMPLE of DERIVATION In Figure 5.9, prFood:Invoice prFood:wasDerivedFrom prFood:Order, meaning that an invoice is created based on an order.
EXAMPLE of USAGE In Figure 5.8, prFood:Measuring prov:used prFood:SampleProduct, meaning an activity of measuring the use of a sample product.
EXAMPLE of ASSOCIATION In Figure 5.8, prFood:Measuring prov:wasAssociatedWith prFood:Lab, meaning that a lab is responsible for conducting a measuring activity.
EXAMPLE of ATTRIBUTION In Figure 5.7, prFood:Gtin prov:wasAttributedTo prFood:Supplier, meaning that a GTIN is ascribed by a supplier.
EXAMPLE of SPECIALISATION In Figure 5.7, prFood:PreperationInstruction is prov:specializationOf prFood:ProductSpecification, meaning that a preparation instruction and a product specification share the same aspects.
EXAMPLE of MEMBERSHIP In Figure 5.9, prFood:Invoice prFood:hadMember prFood:ProductCode, meaning that an invoice can have a collection of product codes.

From this exercise, we demonstrate how PROV Model is able to model the food ecosystem from a limited available dataset. Our limited dataset relates to the food supply chain in Hampshire County that managing kitchens for approximately 600 schools. Those kitchens prepare and provide menus to pupils at schools. There are also several distributors, who deliver the food ingredients to be processed into a meal. Those distributors also get supplied by food suppliers (whether from inside or outside the United Kingdom), who produce the food products. In this context, we identify several food operators (e.g., suppliers, distributors, kitchens, etc.) and requirements from the EU or UK food regulations that they should comply with. At the time *prFood* was constructed we only had schools as a customer, but other types of customer could be added later as new information becomes available. Due to the limited information in our dataset, however, some processes will be added later to construct the complete food supply chain from production to consumption and does not limited the quality of modelling the food supply chain with *prFrame*.

In addition, we show that constructing a product supply chain based on the provenance of a product is possible, and that this carries the various advantages mentioned previously. The result of constructing a product supply chain with provenance is the provenance records, which can be generated and visualised as a PG. Investigating all the processes by visualising them in a graph can help in the identification of peculiar / outlier processes. Finally, this integrated provenance-risk graph provides operators and authorities with an *ease-use* tool to comply with the product's requirements.

5.4 Summary

From this chapter, we gain confidence that *prFrame* can, in principle, be applied in the more specific domain, food. We have demonstrated that *prFrame* is able to model food and its ecosystem by using the more specific ontology, *prFood*. As an ontology, *prFood* conceptualises things about food and its relations by systematically defining them as the classes, object properties, and datatype properties. Moreover, *prFood* is an extension of PROV-O, which means that *prFood* is also able to model the lineage of food from its source to its customers.

prFood also capture a specific risk along the flow of food. Since risk is a concept that is supported by *prFrame*, *prFood* is indisputably able to be used to assess risk within *prFrame*, particularly risk of food contamination, thereby demonstrating due diligence. Here, we believe that the transformation of food caused by food processes also affects the risk of contamination in the final food product. To capture the information about the risk of contamination in the food supply chain, *prFood* is designed to incorporate a general risk framework about bacterial contamination, MPRM. MPRM provides a quantitative approach to assess the risk of contamination in the food processes along the food supply chain. With prFood as an ontology to incorporate MPRM, the reasoning regarding food contamination can be more clear and visible (e.g., where the contamination comes from or ends at).

prFood has also been validated in respect to food regulations that aim to exercise or demonstrate due diligence. This validation is done by performing some SPARQL queries as use cases to request some information and then use it for reasoning. Hence, performing reasoning in the food domain in *prFrame* would only be possible with the help of *prFood*. In the end, the role of *prFood* in modelling the food ecosystem is crucial since it supports the creation of the lineage of food in the form of the food supply chain as well as the risk of contamination. In addition, *prFood* is different from the other food ontologies because it is able to provide a mapping procedure from the existing legacy food ontologies.

Chapter 6

Experimental evaluation of *prFrame* as a case study

In this chapter, we demonstrate *prFrame* through a set of experiments in the specific domain of food, as previously introduced in Chapter 5. Through this, we want to test *prFrame* against various structures in the food supply chain and investigate the result of the *Belief Propagation* inferences in respect to those structures. The intention is to represent the common food supply chains better with branching that reflects the production and distributions of a food product. Our evaluations are concerned with the accuracy of the inference by means of *Belief Propagation* in different structures of the food supply chain. The accuracy of inference is observed based on distance of observed and unobserved variable nodes, observed variable nodes that give the best overall accuracy, and observed variable nodes in certain path. In addition, we try to identify the nodes that are most influential in terms of their ability to affect the overall distribution within the *Factor Graph* (FG).

In the experiment, we introduce observed (obs), unobserved($unObs$), and inferred(inf) variable nodes. An observed variable node is a variable node whose state is known with certainty, while an unobserved variable node is a variable node whose state is unknown. An inferred variable node is an unobserved node, whose states we investigate when performing an inference task. These variable nodes are determined one at the time with a scheme $P(inf|obs)$ for all combinations of all variable nodes in a FG. Note that, $P(inf|obs)$ means that the probability of a single inferred variable node is given by a single observed variable node.

The set of experiments in this chapter contribute to the systematic measurement of accuracy of inference in different structures. In the linear structure, our intuition is test the general supply chain with a set of products that will experience the same processes through the course of that supply chain. This supply chain structure does not consider any branching, such as batching, or the distribution of products at different locations as

a result of branching. To test *prFrame* in a more realistic supply chain with details of batching and branching, we introduce the non-linear supply chain. The construction of the non-linear structure concerns the general processes in the production and distribution of the product supply chain, which can be categorised in one of the structures in *prFrame* (i.e., $-2O$, $-2M$, $O2O$, $M2O_{dirE}$, $O2M_{dirE}$, $M2O_{indE}$, and $O2M_{indE}$). Since we have successfully integrated provenance and risk to enable *prFrame* to assess risk, this chapter shows how *prFrame* performs *Belief Propagation*. The results will be investigated to see whether *prFrame* can potentially be a systematic approach to support due diligence, in particular for risk assessments that can inform subsequent decision making.

To remind the readers, *prFrame* begins by integrating the provenance of food with its risk of contamination in order to model the food supply chain. The process of modelling uses PROV with *prFood* as an extension of PROV-O (Ontology) for the food domain. So as to quantify the risk of contamination, *prFrame* adapts the Modular Process Risk Model (MPRM), which is a general framework of Quantitative Microbial Risk Assessment (QMRA), to estimate the risk of food contamination by undesired colonies of bacteria in food. Ultimately, the result of the modelling can be seen as a food supply chain that shows the set of processes food has gone through (a lineage of food or food provenance). All information is captured in the form of food provenance records, which can be visualised as a graph called a *Provenance Graph* (PG).

A PG with annotated risk models and risk factors will be the basis for the iterative Monte-Carlo (MC) simulation technique to imitate the flow of food products in all of the processes captured in that PG. Each process, with its risk model, will generate a distribution of the predictive output values that takes into account the values of colonies of bacteria as inputs and the associated risk factors. Based on the instantiated predictive output values, *prFrame* constructs the Conditional Probability Tables (CPTs) and will be annotated back to each prov:Activity in the PG. At this point, all the PGs used in this chapter will then be converted into the FGs before performing probabilistic propagation.

The converted FG is, basically, the graphical representation through which *Belief Propagation* will be performed. The results of the inferences arrived at by *Belief Propagation* inform us about the probability distribution of unobserved variable nodes based on the observed nodes across the FG. In other words, how the risk of contamination changes when the updated information about food contamination is revealed in the food supply chain. The updated information relates to the states of food, i.e. whether food is contaminated or not, which are the states in the variable nodes in the FG. Finally, this chapter describes our fourth and fifth contributions, namely a systematic provenance-based factorisation to allow the use of the *Belief Propagation* technique, and a systematic measurement for accuracy of states by means of *Belief Propagation*.

6.1 Method and experiment setup

The experiment aims to evaluate the accuracy of the states inferred by the *Belief Propagation* technique in respect to the risk of food contamination. As mentioned, a state is the existence of bacteria in food after the food processes, namely *absent* or *present*. The state of *absent* indicates the absence of unwanted bacteria in food (uncontaminated food), and the state of *present* indicates the presence of unwanted bacteria in food (contaminated food). In general, the steps in our experimental method and setup are applied to both linear and non-linear food supply chains that are represented as the FGs. Some of those steps are slightly different, however, reflecting changes in the graph topologies and food processes.

6.1.1 Method of experiment

(a) Setup some imaginary structure of PGs with some activities risk factors (e.g., time, temp, etc.). The risk factors are annotated with the parameters for their distributions (e.g., min, max, mean, standard deviation, most likely value, etc). While some risk factors trigger a change of states, others do not.

(b) Based on the PG in step (a), simulate the propagation of the predictive output values (i.e., the colonies of bacteria) estimated by the risk models in each process, taking into account the numeric input values and the risk factors based on past studies or historical data annotated by the PG. Each of these predictive output values is labelled with either *absent* (uncontaminated) or *present* (contaminated) of bacteria. This step is done by executing the provenance-based MC simulation technique.

(c) Construct the CPTs based on the states before and after each process. To remind the reader, the size of a CPT depends on the number of $(states)^{variable\ node}$ combinations in each factor node. For example, if we have four variable nodes connected to a single factor node with two states to be assigned, the size of the CPT in that factor node is $(2)^4 = 16$.

(d) Systematically apply the *Belief Propagation* technique with the *Sum-Product* algorithm to infer the states of an unobserved node, given the states of an observed node.

(e) Since there are two states, *absent* and *present*, the *Sum-Product* algorithm will infer the probability of each of them at the same time with the total probability being 100%. For example, if *Belief Propagation* infers a 45% probability of being *absent*, then the remaining 55% is the probability of being *present*.

(f) Compare the state from MC simulation and the probability of the states from the *Belief Propagation* technique. If the state arrived at by MC simulation is the same as the state with the highest probability generated by the *Belief Propagation* technique, then an inference is categorised as a **CorrectInfer**. For instance, the MC simulation produces *absent*, and *Belief Propagation* produces a higher probability of *absent* than the probability of *present*. Otherwise, it is categorised as **IncorrectInfer**.

(g) Accuracy is measured as follows:

  (i) **CorrectInfer** is defined as when the state predicted by MC simulation *matches* the state inferred by *Belief Propagation* for the same food item.

  (ii) **IncorrectInfer** is defined as when the state predicted by MC simulation *does not match* the state inferred by *Belief Propagation* for the same food item.

  (iii) Accuracy is formulated as $\frac{CorrectInfer}{(CorrectInfer+IncorrectInfer)} * 100\%$

(h) Perform a significance test by comparing groups of the average accuracy. The tests are performed with Analysis Of Variance (ANOVA) test, if each group agrees with the similarity and normally distributed assumptions, or with the Kruskal-Wallis test, if those assumptions are not met.

(i) Calculate a confidence interval between groups of different distances of inferred and observed variable nodes in an FG.

6.1.2 Experimental setup

Overall, the experimental setup for both linear and non-linear food supply chains are similar. One important thing in our setup is our assumption that *there is an unknown factor that changes uncontaminated food to contaminated food* after some processes. This assumption is important because our observation of the actual phenomenon is always limited to some degree. Ignoring this assumption can result in impossibility when constructing the CPT (See Section 4.2 and the construction of the CPT in Figure 4.4). As an example, if there is no event from uncontaminated to contaminated food during the MC simulation, then *Belief Propagation* will assume that it is impossible for uncontaminated food to become contaminated after that food process. This, of course, would be to neglect the fact that uncontaminated food can still be contaminated, which possibly happens in the food supply chain. This assumption is captured as a risk factor with formula $1 * (\alpha)$, where α is configured between 1% - 5.7% according to the UK-wide Survey of Salmonella and Campylobacter Contamination of Fresh and Frozen Chicken on Retail Sale by Food Standards Agency (FSA)[1]. This risk factor is important for risk models that only calculate the *predictive output values* without changing the states. This

[1] https://www.food.gov.uk/sites/default/files/media/document/b180002finreport.pdf

means that the risk models are only applied to food that initially has bacteria present, and will not change the state from uncontaminated to contaminated food. A risk factor with the formula $1 * (\alpha)$ therefore facilitates random changes from an uncontaminated into a contaminated state and the food processes that have this risk factor will produce extra contaminated food after those processes.

(a) Four linear and non-linear factor graphs are presented in Figure 6.1 (linear) and Figure 6.5 (non-linear). Those graphs are the converted FGs based on their PGs, which represent the food supply chains.

(b) The various distributions of the risk factors are configured based on historical data to trigger the change of the states in the food risk models (Table 6.1). We adapt those risk factors from [65] (See Appendix for more details about those risk factors). The instance values of the risk factors are randomly chosen during the iterative MC simulation.

(c) Predicting states accurately by *Belief Propagation* entails three sequential processes, as follows:

  (i) Perform the MC simulation with exactly 1,000 iterations for each of the FGs. These iterations represent the flow of 1,000 food products, which is our general assumption of the quantity of food product in a single batch. The number of iterations varies in the non-linear FGs, however, because of the different structures of the graphs that reflect batching in the distribution of products. For example, we may start with 500 food items in two locations before we combine those into one location with a particular food process.

  (ii) Construct a set of CPTs as the factors for each factor node based on the frequency of the changing states before and after the process.

  (iii) Perform an inference technique with *Belief Propagation* for all combinations of nodes in each FG and compare the results with the states generated by MC simulation.

(d) The three processes in step (c) are repeated 100 times in order to get the average accuracy of the states when *Belief Propagation* infers a node by observing another node. With 100 iterations over 1,000 food products in each FG, there should be sufficient confidence to draw a general conclusion.

Table 6.1 shows a simple version of the risk factors we use in the our experiment. Each process has a set of risk factors (rf) that will change the number of bacteria or change the states. The risk factors are defined by means of probability distributions that best represent their nature. As mentioned, some processes have a risk factor with the formula $1 * (\alpha)$ that serves to change the state from uncontaminated to contaminated food.

initial process (*i*)
rf1: (0.00,1.08,2.08,3.08,4.08,4.64,4.68)
rf2: (0.000,0.600,0.370,0.010,0.006,0.006,0.008)
retailing (*r*)
rf1: truncated_normal (mean=4;sd=2.8;min=-7.2;max=10)
rf2: corellated_uniform (mean=2;max=7)
rf3: 1*(α)
transporting (*t*)
rf1: truncated_normal (mean=4;sd=2.8;min=-7.2;max=10)
rf2: pert (min=0;most=13;max=24)
rf3: truncated_normal (mean=3.72;sd=2.82;min=0;max=rf2-rf1)
rf4: corellated_uniform(min=5;max=240)
rf5: 1*(α)
storing (*s*)
rf1: truncated_normal (mean=4;sd=2.65;min=-6.1;max=21.1)
rf2: pert (min=0;most=2;max=5)
rf3: 1*(α)
preparing (*p*)
rf1: pert (min=0;most=0.1;max=0.15)
rf2: pert (min=0;most=0.1;max=0.15)
rf3: pert (min=0;most=0.1;max=0.15)
rf4: 1*(α)
cooking (*c*)
rf1: pert (min=0.05;most=0.10;max=0.15)
rf2: pert (min=60;most=64;max=65)
rf3: pert (min=0.5;most=1.0;max=1.5)

TABLE 6.1: The risk factors of the food processes.

6.2 Experiments in a linear food supply chain

Our experiment in the linear food supply chains aims to investigate the accuracy of the inferences generated by *Belief Propagation* in a single batch of food products. As described in Section 4.3, a linear chain describes the general processes in the kind of food supply chain where all food must experience the same processes. The *Belief Propagation* technique is performed by observing a single variable node to infer another single variable node for all possible combinations of variable nodes in the linear FGs. Overall, the aim of the linear chain experiment is to demonstrate a single batch of a food product through consecutive food processes. In the linear chain, each of the food products goes to each of the food processes since there no branching in the food supply chain. This is the main difference between some of the structures in a non-linear chain, which we will explore later.

Figure 6.1 illustrates the different topologies of the linear food supply chains. These FGs are converted from the *Provenance Graph* (PG). The rationale behind investigating linear food supply chains here is to establish that the technique works with the more general food supply chain that contain less detail about batching or where food comes from and goes to. All the food product in a linear food supply chain experience the same processes along the chain. In each graph, the "star" symbol represents food from the same origin or the same batch. In this use case, all food comes from the same source and will experience the same processes from the beginning to the end of the chain. The naming of a variable node indicates the food stage as a result of a process with an arrow

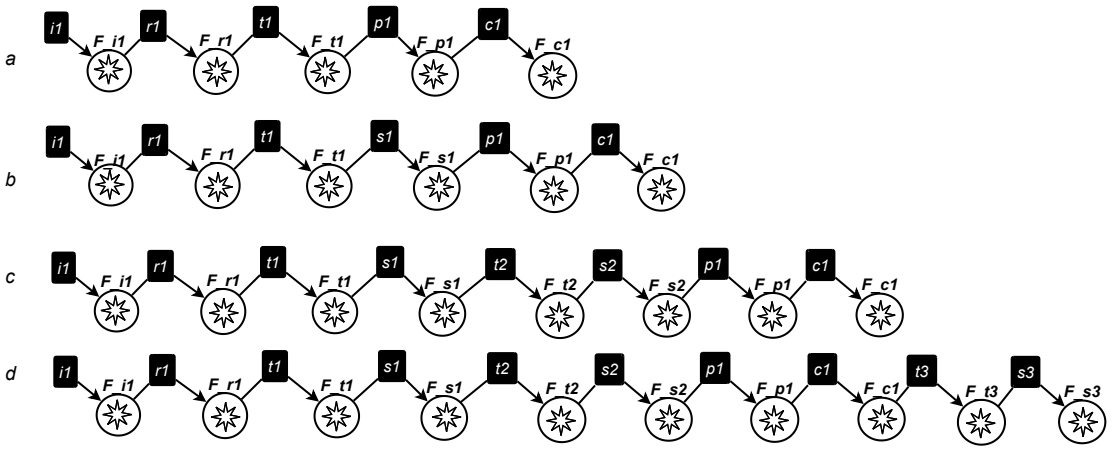


FIGURE 6.1: Four linear factor graphs representing different linear food supply chains.

to that variable node. For example, variable node ***F_r1*** represents a food product that has been processed with a *retailing* process (i.e., ***F*** represents food and ***_r1*** represents a *retailing* process). Thus, its graph representation has one variable node ***F_r1*** and one factor node ***_r1*** with a directed edge pointing to that variable node.

Since it is a linear chain, each factor node (except in factor node *i1*) has exactly two variable nodes connected to it, representing the input and output of food after each process. Each food product is assigned a predicted output value (i.e., colonies of salmonella), which is categorised into a state. With two available states (*absent* and *present*), the uncertainty is determined by $(2)^2$ combinations as described in Section 6.1.1(c) and is represented as a factor in a CPT in each factor node (except in factor node *i1*).

6.2.1 Experiment 1

This experiment focuses on two things: 1) to what extent the accuracy of *Belief Propagation* changes as the food supply chain gets longer. 2) the best food stage to observe in a linear food supply chain. Hence, our first hypothesis is that the accuracy with which *Belief Propagation* is able to determine the states is decreased as the chain gets longer (or the distance between the observed and inferred nodes get wider). Our second hypothesis is that using different variable nodes to infer other unobserved variable nodes has an impact on the accuracy with which *Belief Propagation* is able to determine the states. To test our hypotheses, we present our null hypotheses as follow:

Null Hypothesis 1. Longer chains in a linear *factor graph* have no effect on the accuracy with which *Belief Propagation* is able to infer variable nodes to determine the states.

Null Hypothesis 2. Observing different variable nodes in a linear *factor graph* has no effect on the accuracy of inference with which *Belief Propagation* is able to determine the states.

Results and discussion. The average accuracy with which *Belief Propagation* is able to determine the states is presented in Figure 6.2, where each table is a result of each FG. Each cell in each table shows the average accuracy in respect to determining the states when using the observation of one particular variable node (column) to infer another variable node (row). For example, the average accuracy of inference when observing variable node ***F_r1*** to infer variable node ***F_t1*** ($P(\boldsymbol{F\_t1}|\boldsymbol{F\_r1})$) is 97.9%.

		Observe				
		F_i1	F_r1	F_t1	F_p1	F_c1
Infer	F_i1	100.00	98.05	95.44	93.11	80.81
	F_r1	98.05	100.00	97.9	95.56	78.85
	F_t1	95.44	97.9	100.00	98.16	76.77
	F_p1	93.11	95.56	98.16	100.00	74.91
	F_c1	97.59	97.59	97.59	97.59	100.00

a. The average accuracy of inference in the Factor Graph in Figure 6.1(a)

		Observe					
		F_i1	F_r1	F_t1	F_s1	F_p1	F_c1
Infer	F_i1	100.00	97.79	95.32	92.99	90.53	80.58
	F_r1	97.79	100.00	98.01	95.68	93.22	78.44
	F_t1	95.32	98.02	100.00	98.15	95.69	76.58
	F_s1	92.98	95.68	98.15	100.00	98.03	74.68
	F_p1	90.52	93.22	95.69	98.03	100.00	72.73
	F_c1	97.18	97.18	97.19	97.19	97.19	100.00

b. The average accuracy of inference in the Factor Graph in Figure 6.1(b)

		Observe							
		F_i1	F_r1	F_t1	F_s1	F_t2	F_s2	F_p1	F_c1
Infer	F_i1	100.00	98.13	95.49	93.07	90.7	88.57	86.41	80.02
	F_r1	98.13	100.00	97.85	95.43	93.05	90.93	88.77	78.12
	F_t1	95.49	97.85	100.00	98.06	95.69	93.56	91.41	75.98
	F_s1	93.05	95.42	98.06	100.00	98.12	95.99	93.84	74.07
	F_t2	90.68	93.04	95.68	98.12	100.00	98.37	96.22	72.19
	F_s2	88.56	90.92	93.56	95.99	98.37	100.00	98.33	70.48
	F_p1	86.39	88.76	91.4	93.83	96.2	98.32	100.00	68.75
	F_c1	97.04	97.04	97.04	97	97.04	97.04	97.04	100.00

c. The average accuracy of inference in the Factor Graph in Figure 6.1(c)

		Observe									
		F_i1	F_r1	F_t1	F_s1	F_t2	F_s2	F_p1	F_c1	F_t3	F_s3
Infer	F_i1	100.00	97.9	95.37	93.01	90.52	88.23	86.18	80.23	79.44	79.39
	F_r1	90.45	100.00	97.96	95.6	93.11	90.82	88.77	78.21	76.99	76.81
	F_t1	95.36	97.95	100.00	98.13	95.64	93.35	91.31	76.17	74.76	74.33
	F_s1	93	95.59	98.13	100.00	98.01	95.72	93.67	74.33	72.88	72.16
	F_t2	90.51	93.1	95.64	98.01	100.00	98.21	96.16	72.34	70.99	70.01
	F_s2	88.22	90.81	93.35	95.72	98.21	100.00	98.45	70.53	69.3	68.28
	F_p1	86.17	88.76	91.31	93.67	96.16	98.45	100.00	68.92	67.81	66.8
	F_c1	96.64	96.64	96.64	96.64	96.64	96.64	96.64	100.00	97.69	96.76
	F_t3	93.5	93.5	93.5	93.5	93.5	93.5	93.5	97.35	100.00	97.46
	F_s3	90.45	90.45	90.45	90.45	90.45	90.45	90.45	94.3	97.45	100.00

d. The average accuracy of inference in the Factor Graph in Figure 6.1(d)

FIGURE 6.2: The average accuracy of inference in the linear factor graphs in Figure 6.1.

Overall, it can be seen from Figures 6.2a to 6.2d that the average accuracy decreases as the distance between inferred and observed variable nodes is increased. In other words, the uncertainty increases as the variable nodes become further apart. This increment is produced from the risk factor 1*(α), which aims to change the state *absent* to *present* according to our premise in the experimental setup (Section 6.1.2). Initially, the distribution of food is approximately 79% uncontaminated (*absent*) and 21% contaminated (*present*) in the first variable node ***F_i1***[2]. Incorporating the risk factor 1*(α) in the food processes after variable node ***F_i1*** consistently changes the distribution of the state *absent* and *present* towards equal chance (50%:50%) thereby making the inference after the variable node ***F_i1*** become more uncertain.

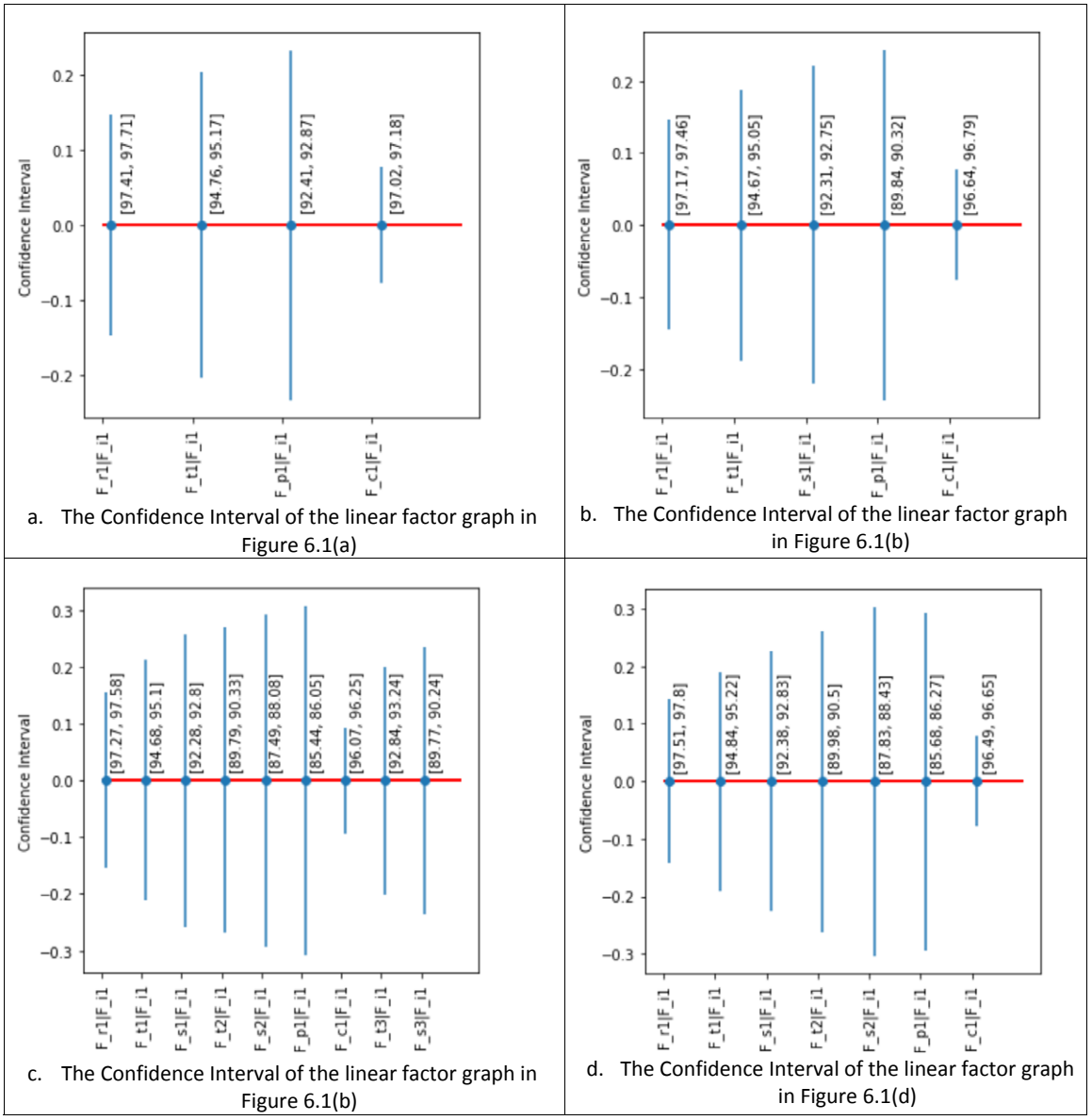
After the *cooking* process, however, almost all contaminated food becomes uncontaminated in the variable node ***F_c1***. Consequently, inferring the states of food in the variable node ***F_c1*** increases the accuracy as the chance of getting uncontaminated food is very high (extreme) in variable node ***F_c1*** (row ***F_c1***). In contrast, inferring the other variable nodes by observing variable node ***F_c1*** does not produce as high accuracy as inferring variable node ***F_c1***, because the proportions of contaminated and uncontaminated food in those variable nodes are not as extreme as when we infer variable node ***F_c1***. To support this finding, we also present the Confidence Interval (CI) in Figure 6.3 to illustrate the decrease of confidence in respect to the accuracy with which *Belief Propagation* is able to determine the states as the chain gets longer.

In Figure 6.3a to 6.3d, the CI is indicated by the length of the blue vertical lines and the mean is indicated by the red horizontal line. The wider the line, the less confident we are in the accuracy with which *Belief Propagation* is able to determine the states. For example, Figure 6.3a shows that the length of CI for *P(**F_r1**|**F_i1**)* is 0.3 (97.71 - 97.41) and its mean is 97.56. It is clear that confidence decreases as the distance between an observed and an unobserved variable node gets longer. It continues until we infer the variable node ***F_c1***, where we are very confident of the accuracy of inference by *Belief Propagation*; obviously, because almost all food is uncontaminated in this variable node.

In the following paragraphs, we focus our discussion on the difference in average accuracy when we compare the different group of datasets from Figure 6.2. There are two reasons for testing the significance of accuracy. First, we want to understand whether the length of the linear chain significantly affects the accuracy of the inference. Second, we want to understand whether observing a different variable node results in a significant difference in accuracy. The result of the significance test is shown in Table 6.2 that shows the *p-values* to reject or not reject the *null hypothesis 1* and *null hypothesis 2*.

Our *null hypothesis 1* is tested by treating all average accuracies in Tables 6.2a, 6.2b, 6.2c and 6.2d as an individual independent group. Since the *p-values* are smaller than 0.05 we conclude that we reject the *null hypothesis 1*. In addition, the average accuracy

[2]Variable node in a FG represents a food stage in a food supply chain.



a. The Confidence Interval of the linear factor graph in Figure 6.1(a)

b. The Confidence Interval of the linear factor graph in Figure 6.1(b)

c. The Confidence Interval of the linear factor graph in Figure 6.1(b)

d. The Confidence Interval of the linear factor graph in Figure 6.1(d)

FIGURE 6.3: The confidence interval in the linear factor graphs in Figure 6.1.

	Linear FG a	**Linear FG b**	**Linear FG c**	**Linear FG d**
Null Hyp.1	*0.0173*			
Null Hyp.2	*1.23e-9*	*7.97e-12*	*2.7e-15*	*0.001*

TABLE 6.2: The *p-values* of significance test in the linear factor graphs in Figure 6.1.

in each table also decreases (average of Figure 6.2a (94.33%) < 6.2b (93.88%) < 6.2c (93.04%) < 6.2d (90.31%)), suggesting that the longer the chain the less accurate the inference is.

To test our *null hypothesis 2*, we compare the groups of average accuracy when observing a different variable node in each linear FG. In other words, the groups are constructed vertically in each table in Figure 6.2. For example in Table 6.2a, the first group is the group of average accuracy when we observe variable node ***F_i1*** and is [98.05, 95.44,

93.11, 97.59] and so on for the other groups ***F_r1***, ***F_t1***, ***F_p1***, ***F_c1***. As a result, each table rejects the *null hypothesis 2*. This result helps us to investigate a product stage (variable node) to achieve the best accuracy across the food supply chain. Finally, by calculating an average in each column in each table, we derive ***F_r1*** as the best single variable node to observe that gives us the highest overall accuracy.

6.2.2 Experiment 2

In Figure 6.2, we notice that performing an inference task with the *cooking* process (a factor node *c1*) generates asymmetric accuracy in each table as it changes the distribution of the states *absent* and *present* drastically. This certainly goes on to significantly change the distribution of contaminated and uncontaminated food in the food stage after the *cooking* process. Hence, we perform Experiment 2 to verify whether the *cooking* process is the most influential node to change the distribution of contaminated and uncontaminated food in the linear food supply chain.

In this experiment, we focus on investigating the accuracy of risk propagation without this process; thus, the food stage after *cooking* process, variable node ***F_c1***, will be excluded from all the FGs in Figure 6.1. With the same motivation as in the first experiment, we present similar hypotheses that we validate in the linear FGs without the *cooking* process. We therefore present our *null hypotheses* as follows:

Null Hypothesis 3. In a linear *factor graph* without a *cooking* process, there is no difference in the accuracy with which *Belief Propagation* is able to infer variable nodes to determine the food states as the chain gets longer.

Null Hypothesis 4. In a linear *factor graph* without a *cooking* process, there is no difference in the accuracy with which *Belief Propagation* is able to infer variable nodes to determine the food states when observing each different node.

Results and discussion. In this experiment, we exclude the variable node ***F_c1*** and the factor nodes whose directed edges point to variable node ***F_c1*** in each linear FG from our Experiment 1. As configured in the cooking risk model in Table 6.1, this process transforms the contaminated food to uncontaminated food with a binomial distribution (85%-95%. See *cooking* process *rf1*[3]). This means that the odds of food that is going through this process being uncontaminated are between 85% and 95%. As a result, the majority of food becomes uncontaminated food after this process. With the same method and experimental setup as in the previous experiment, Figure 6.4 is generated.

The results in Figure 6.4 show that the accuracy is symmetrical in the linear chains without factor node *c1* or a *cooking* process. The result is similar to our first experiment, where the accuracies by *Belief Propagation* in the variable nodes before variable

[3]Each iteration in the MC simulation will randomly choose a single number from a pert distribution in the *rf1 cooking* process. Further, the chosen number will determine the state of food with a binomial distribution.

Infer \ Observe	F_i1	F_r1	F_t1	F_p1
F_i1	100.00	97.8	95.2	92.8
F_r1	97.8	100.00	97.9	95.4
F_t1	95.2	97.9	100.00	98.1
F_p1	92.8	95.4	98.1	100.00

a. The average accuracy of inference in the Factor Graph in Figure 6.1(a) without a cooking process

Infer \ Observe	F_i1	F_r1	F_t1	F_s1	F_p1
F_i1	100.00	98.1	95.7	93.2	90.93
F_r1	98.1	100.00	98.1	95.6	93.34
F_t1	95.7	98.1	100.00	98	95.73
F_s1	93.2	95.6	98	100.00	98.24
F_p1	90.9	93.3	95.7	98.2	100.00

b. The average accuracy of inference in the Factor Graph in Figure 6.1(b) without a cooking process

Infer \ Observe	F_i1	F_r1	F_t1	F_s1	F_t2	F_s2	F_p1
F_i1	100.00	98	95.5	93.1	90.98	88.7	86.3
F_r1	98	100.00	98	95.6	93.49	91.3	88.82
F_t1	95.5	98	100.00	98.1	95.99	93.8	91.31
F_s1	93.1	95.6	98.1	100.00	98.43	96.2	93.76
F_t2	91	93.5	96	98.4	100.00	98.3	95.83
F_s2	88.7	91.3	93.8	96.2	98.26	100.00	98.07
F_p1	86.3	88.8	91.3	93.8	95.83	98.1	100.00

c. The average accuracy of inference in the Factor Graph in Figure 6.1(c) without a cooking process

Infer \ Observe	F_i1	F_r1	F_t1	F_s1	F_t2	F_s2	F_p1	F_t3	F_s3
F_i1	100.00	98	95.4	93	90.69	88.4	86.42	84.09	82.1
F_r1	98	100.00	97.9	95.5	93.19	90.9	88.92	86.59	84.5
F_t1	95.4	97.9	100.00	98.1	95.75	93.5	91.48	89.15	87
F_s1	93	95.5	98.1	100.00	98.19	95.9	93.92	91.59	89.5
F_t2	90.7	93.2	95.8	98.2	100.00	98.2	96.23	93.9	91.8
F_s2	88.4	90.9	93.5	95.9	98.24	100.00	98.49	96.16	94
F_p1	86.4	88.9	91.5	93.9	96.23	98.5	100.00	98.17	96.1
F_t3	84.1	86.6	89.2	91.6	93.9	96.2	98.17	100.00	98.4
F_s3	82	84.5	87	89.5	91.78	94	96.05	98.38	100.00

d. The average accuracy of inference in the Factor Graph in Figure 6.1(d) without a cooking process

FIGURE 6.4: The average accuracy of inference in the linear factor graphs in Figure 6.1 without the *cooking* process.

node ***F_c1*** were also symmetrical. This suggests that the cooking process is the most influential process in respect to changing the distribution of contaminated and uncontaminated food. In addition, calculating the confidence interval to see the decrease in confidence of accuracy as the chain gets longer generates similar results as in Figure 6.3, although without the confidence interval for ***F_c1|F_i1*** since we exclude variable node ***F_c1*** in this experiment.

Testing our null hypotheses with the same procedure as in Experiment 1 results in failure to reject our null hypotheses as shown in Table 6.3. Essentially, this means that there is no significant difference in the accuracy with which *Belief Propagation* is able to infer variable nodes to determine the food states as the chain gets longer in the linear FG (*null*

	Linear FG a without c1	**Linear FG b without c1**	**Linear FG c without c1**	**Linear FG d without c1**
Null Hyp.3	*0.059*			
Null Hyp.4	*0.764*	*0.733*	*0.372*	*0.254*

TABLE 6.3: The *p-values* of significance test in the linear factor graphs without *cooking* process in Figure 6.1.

hypothesis 3). Since variable node ***F_c1*** is the most influential variable node in respect to changing the distribution, removing it from the supply chain reduces the opportunity for food states to change, and this is reflected in the subsequent insignificant differences in the distribution of food states. Nonetheless, the *p-value* of this test is close to the threshold (smaller than 0.05 to reject *null hypothesis 3*).

Testing our *null hypothesis 4* returns the same result as for *null hypothesis 3* in each of the linear FGs (Table 6.3). All the *p-values* are bigger than the threshold required to reject the null hypothesis, although those *p-values* gradually decrease. This suggests that even though there is a difference in the accuracy of the states determined by *Belief Propagation* between the linear FGs exists, this difference is not significant because the most influential variable node has been excluded.

Our intuition in respect to the result of Experiment 2 is that the *cooking* process can be considered as the most important process in respect to changing the distribution of contaminated and uncontaminated food. If the proportion of contaminated food is high, therefore, having a process like *cooking* can make food a lot safer for consumption. If the proportion of contaminated food is lower or below the threshold considered to be safe, however than having a *cooking* process will not be necessary, especially if the process is costly to be perform in the food supply chain.

6.3 Experiments in a non-linear food supply chain

Our non-linear experiments involve non-linear FGs that represent food products that have different batches in the food supply chains. The experiments in this section have a similar methodology, experimental setup, risk models and risk factors to the experiments with the linear FGs (See Table 6.1 for the list of the risk factors). The intention of the experiments with the non-linear FGs is to demonstrate the operation of our technique within more realistic food supply chains with different lines / paths / batches of production and distribution. The non-linear chain therefore reproduces more detail of the food production and distribution life cycle. The main difference from the linear food supply chain is that not all food products exist in each food process in the food supply chains.

During our investigation, most well-known food processes (e.g., cooking, cutting, cleaning, washing, etc.) fall into one of the structures in *prFrame* ($-2O$, $-2M$, $M2O_{dirE}$, $O2M_{dirE}$, $M2O_{indE}$, and $O2M_{indE}$). To determine which food process belongs to which

structure, we have to read the description of each food process identified in a food supply chain. Figure 6.5 shows a non-linear food supply chain with different structures of branching.

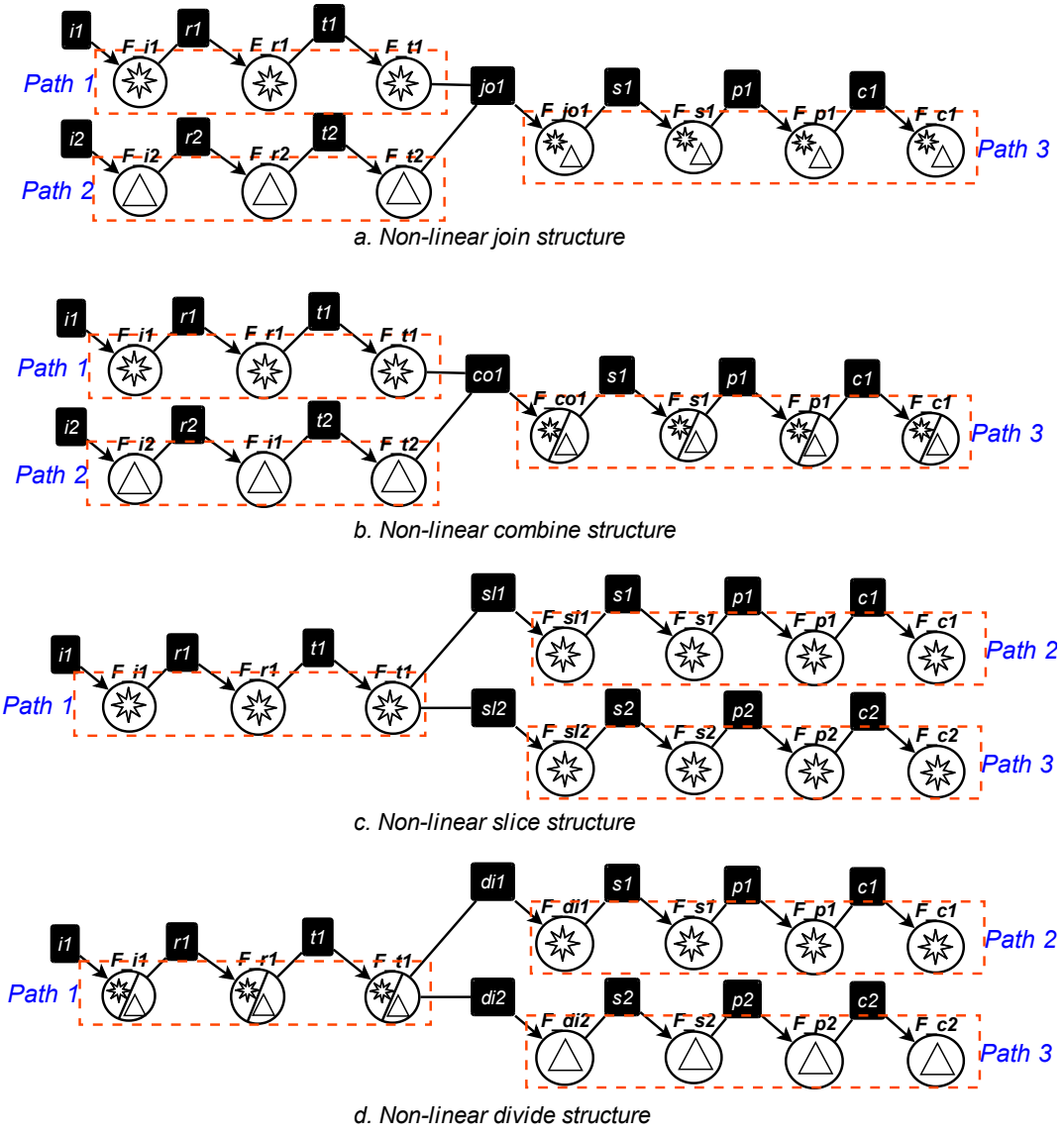


FIGURE 6.5: Four non-linear factor graphs.

Figure 6.5a and 6.5b show the *M2O* structure (*joining* and *combining* processes) of two origins (paths) of the food products. Here, we assume that those paths represent two batches of the food products with the symbols "**star**" and "**triangle**". On the one hand, the *joining* process refers to when all the food in one batch is physically mixed with other food from another batch; thus, the quantity of the output is the same as the quantity in each input. On the other hand, the *combining* process only collects the batches of food products into one location or as a new single batch; thus, the quantity of the output is the sum of the quantities in each input.

Figures 6.5c and 6.5d show the $O2M$ structure (*slicing* and *dividing* processes) of two origins (paths) of the food products. Similar to the *joining* process, the *slicing* process physically slices or tears food into smaller pieces, which can be put into several batches. Thus, the quantity of each output is the same as the quantity of the input. The *dividing* process is also similar to the *combining* process, since it divides or relocates a single batch into several batches. Thus, the quantity of input is the sum of the quantities in each output.

Since the chain is non-linear, some of the factor nodes have more than two variable nodes connected to them based on the method (c) (Section 6.1.1). For example, the combine and join processes in Figure 6.5b have a factor node that holds the combination of $(2)^3$ (three variable nodes with two states).

6.3.1 Experiment 3

For the non-linear experiment, we modify our hypotheses because the structures in the non-linear chain have an impact on the accuracy of the inferences made by *Belief Propagation* when observing different variable nodes to determine the food states. Using the same methodology and experimental setup as in the first experiment, we present Figure 6.7 for the average accuracy in each non-linear FG previously shown in Figure 6.5 to check our null hypotheses as follows:

Null Hypothesis 5. There is no difference in the accuracy of the inferences made by *Belief Propagation* to determine the food states in the different structures of the non-linear *factor graph*s.

Null Hypothesis 6. There is no difference in the accuracy of the inferences made by *Belief Propagation* to determine the food states when observing each different variable node in the non-linear *factor graph*s.

Results and discussion. Similar to the first experiment, the accuracy has deteriorated since the distance between the observed and inferred variable nodes is further apart because of the presence of the risk factor $1^*(\alpha)$ in some food processes. In Figure 6.7a, 6.7b, 6.7c, some of the cells do not have values for average accuracy because food following some of the processes does not go through all of the other processes. For example in Figure 6.6a (a combine structure), it is not possible to determine the accuracy in observing ***F_i2*** to infer ***F_i1*** ($P(\boldsymbol{F\_i1}|\boldsymbol{F\_i2})$) since the food in ***F_i2*** does not exist in ***F_i1***. Also, the *cooking* process increases the accuracy for the same reason described in the linear experiment. This is also supported by the confidence interval shown in Figure 6.6.

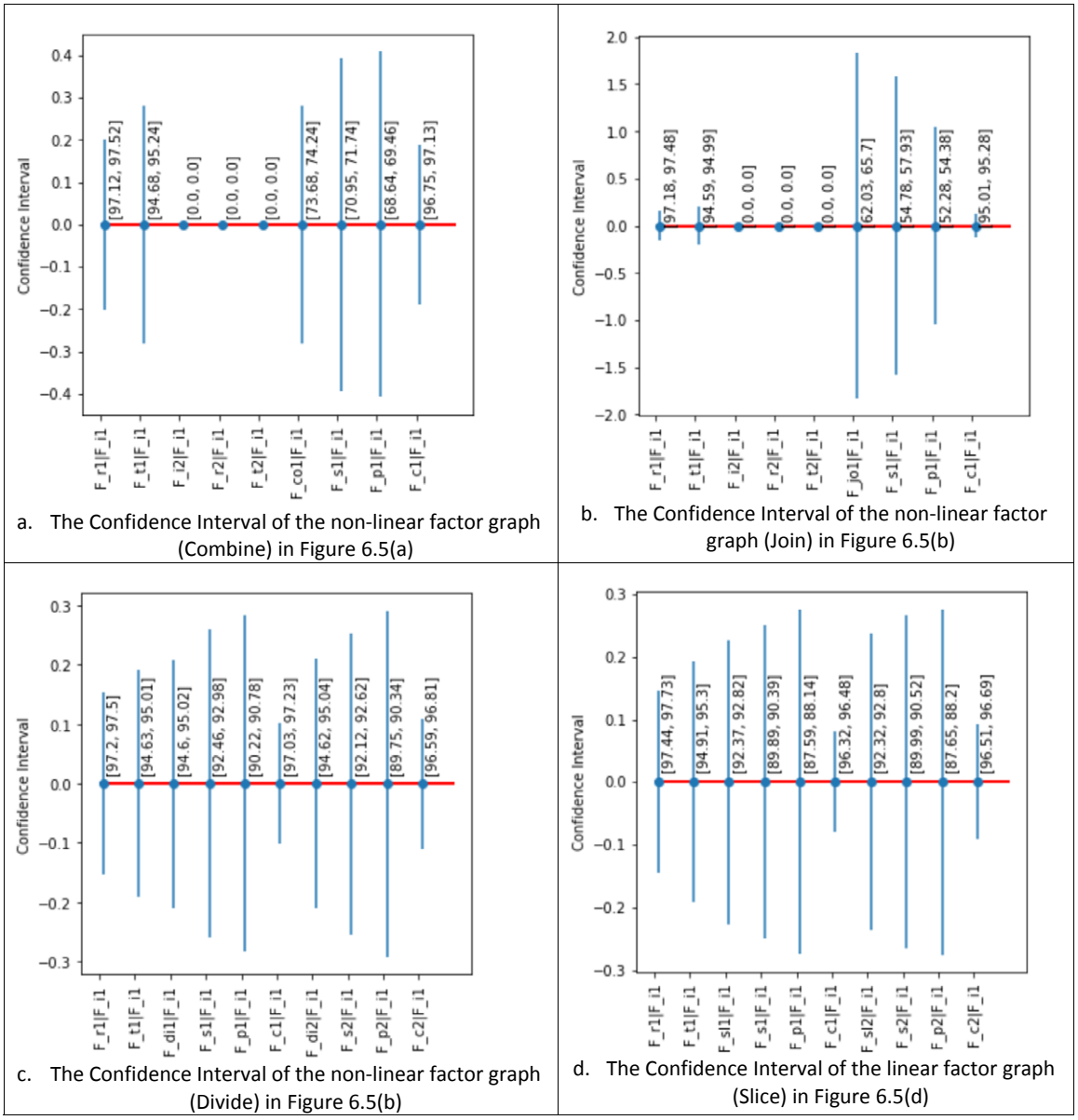
In Figures 6.6a, 6.6b, 6.6c and 6.6d, our conclusion regarding the confidence interval in the non-linear food supply chains is similar to that for linear food supply chains. The graph in each cell shows that the longer the distance, the less confident we are

about the accuracy of *Belief Propagation*. The different result was found for the join structure, however, where the confidence in observing variable node ***F_i1*** to infer the nodes after variable node ***F_jo1*** is decreased. This is caused by the increasing number of instances of contaminated food after the *joining* process, which changes the distribution to be almost equal (around 50% for both contaminated and uncontaminated food). This represents the lowest confidence we can have about the accuracy of inference by *Belief Propagation*. After the process of joining, the formula $1^*(\alpha)$ changes the distribution away from almost equal to more extreme as the number of instances of contaminated food increases to become similar to the distribution in variable node ***F_i1***. As a result, we gain more confidence about the accuracy by *Belief Propagation*, particularly when inferring the variable node ***F_c1***.

Again, we performed an ANOVA and Kruskal-Wallis tests to find the significance of any differences between different groups of accuracy. We present the result of those tests in Table 6.4, where we validate our set of hypotheses in this experiment.

	Non-linear (Combine)	**Non-linear (Join)**	**Non-linear (Divide)**	**Non-linear (Slice)**
Null Hyp.5	*5.149e-07*			
Null Hyp.6	*0.955*	*0.607*	*4.565e-04*	*1.04e-05*

TABLE 6.4: The *p-values* significance test in the non-linear factor graphs in Figure 6.5.



a. The Confidence Interval of the non-linear factor graph (Combine) in Figure 6.5(a)

b. The Confidence Interval of the non-linear factor graph (Join) in Figure 6.5(b)

c. The Confidence Interval of the non-linear factor graph (Divide) in Figure 6.5(b)

d. The Confidence Interval of the linear factor graph (Slice) in Figure 6.5(d)

FIGURE 6.6: The confidence interval of the accuracy based on the distance of inferred and observed variable nodes in the non-linear factor graphs in Figure 6.5.

Infer \ Observe	F_i1	F_r1	F_t1	F_i2	F_r2	F_t2	F_co1	F_s1	F_p1	F_c1
F_i1	100.00	98.28	95.92	-	-	-	79.79	79.79	79.79	79.8
F_r1	98.28	100.00	98.62	-	-	-	77.09	77.09	77.11	77.11
F_t1	95.9	98.61	100.00	-	-	-	74.71	74.71	74.74	74.74
F_i2	-	-	-	100.00	98.72	96.49	79.79	79.79	79.8	79.8
F_r2	-	-	-	98.72	100.00	98.77	77.51	77.51	77.51	77.51
F_t2	-	-	-	96.5	98.78	100.00	75.28	75.28	75.27	75.27
F_co1	74.69	74.69	74.71	75.3	75.3	75.28	100.00	98.43	96.04	77.06
F_s1	72.05	72.05	72.08	72.78	72.78	72.77	98.43	100.00	98.62	75.02
F_p1	69.73	69.73	69.78	70.26	70.26	70.25	96.02	98.59	100.00	73.13
F_c1	97.9	97.9	97.91	97.92	97.92	97.92	97.91	97.91	97.92	100.00

a. The average accuracy of inference in the non-linear Combine structure in Figure 6.5(a)

Infer \ Observe	F_i1	F_r1	F_t1	F_i2	F_r2	F_t2	F_jo1	F_s1	F_p1	F_c1
F_i1	100.00	97.82	95.27	-	-	-	79.4	79.4	79.4	79.4
F_r1	97.82	100.00	97.95	-	-	-	76.71	76.71	76.71	76.71
F_t1	95.27	97.95	100.00	-	-	-	74.16	74.16	74.16	74.16
F_i2	-	-	-	100.00	97.8	95.35	79.4	79.4	79.4	79.4
F_r2	-	-	-	97.8	100.00	98.05	76.7	76.7	76.7	76.7
F_t2	-	-	-	95.35	98.05	100.00	74.25	74.25	74.25	74.25
F_jo1	64.18	71.7	78.19	64.71	70.69	76.79	100.00	98.94	97.28	47.55
F_s1	56.64	64.4	72.79	57.16	64.72	71.22	98.94	100.00	98.84	48.89
F_p1	53.6	57.08	64.26	53.86	57.86	63.56	97.28	98.84	100.00	50.56
F_c1	95.62	95.62	95.62	95.62	95.62	95.62	95.62	95.62	95.62	100.00

b. The average accuracy of inference in the non-linear Join structure in Figure 6.5(b)

Infer \ Observe	F_i1	F_r1	F_t1	F_di1	F_s1	F_p1	F_c1	F_di2	F_s2	F_p2	F_c2
F_i1	100.00	97.84	95.3	95.29	93.18	90.95	80.41	95.31	92.83	90.5	80.6
F_r1	97.84	100.00	97.96	97.98	95.87	93.64	78.21	97.95	95.48	93.14	78.59
F_t1	95.3	97.96	100.00	100	98.39	96.17	76.19	100	98.03	95.69	76.63
F_di1	95.29	97.98	100	100.00	98.39	96.17	76.19	-	-	-	-
F_s1	93.18	95.87	98.39	98.39	100.00	98.28	74.5	-	-	-	-
F_p1	90.95	93.64	96.17	96.17	98.28	100.00	72.64	-	-	-	-
F_c1	97.62	97.62	97.62	97.62	97.62	97.62	100.00	-	-	-	-
F_di2	95.31	97.95	100	-	-	-	-	100.00	98.03	95.69	76.63
F_s2	92.83	95.48	98.03	-	-	-	-	98.03	100.00	98.16	74.7
F_p2	90.5	93.14	95.69	-	-	-	-	95.69	98.16	100.00	72.8
F_c2	97.19	97.19	97.19	-	-	-	-	97.19	97.19	97.19	100.00

c. The average accuracy of inference in the non-linear Divide structure in Figure 6.5(c)

Infer \ Observe	F_i1	F_r1	F_t1	F_sl1	F_s1	F_p1	F_c1	F_sl2	F_s2	F_p2	F_c2
F_i1	100.00	98.08	95.58	93.06	90.59	88.31	80.45	93.03	90.71	88.37	80.32
F_r1	98.08	100.00	98.01	95.49	93.02	90.73	78.51	95.45	93.13	90.79	78.43
F_t1	95.58	98.01	100.00	97.98	95.51	93.23	76.47	97.94	95.63	93.29	76.47
F_sl1	93.06	95.49	97.98	100.00	98.03	95.75	74.53	95.59	93.43	91.24	74.01
F_s1	90.59	93.02	95.51	98.03	100.00	98.22	72.6	93.29	91.28	89.28	71.58
F_p1	88.31	90.73	93.23	95.75	98.22	100.00	70.82	91.2	89.33	87.47	69.34
F_c1	96.89	96.89	96.89	96.89	96.89	96.89	100.00	96.89	96.89	96.89	96.89
F_sl2	93.03	95.45	97.94	95.59	93.29	91.2	73.97	100.00	98.18	95.85	74.35
F_s2	90.71	93.13	95.63	93.43	91.28	89.33	71.7	98.18	100.00	98.16	72.52
F_p2	88.37	90.79	93.29	91.24	89.28	87.47	69.4	95.85	98.16	100.00	70.68
F_c2	97.08	97.08	97.08	97.08	97.08	97.08	97.08	97.08	97.08	97.08	100.00

d. The average accuracy of inference in the non-linear Divide structure in Figure 6.5(d)

FIGURE 6.7: The average accuracy of inference in the non-linear factor graphs in Figure 6.5.

It is clear from Table 6.4 that there is a significant difference in the accuracy of the states determined by *Belief Propagation* between different structures of the non-linear FGs (*p-value* = 5.149e-07). The rejection of the *null hypothesis 5* is expected as the dataset about the average accuracy in each table in Figure 6.7 is quite different from the others, both in the number of datasets per table and the accuracy of the states determined by *Belief Propagation*. For example, the datasets of average accuracy in Tables 6.7a, 6.7b and 6.7c have some missing data because not all food products exist in each process; hence, we cannot calculate the accuracy for those inferences. This difference contributes to the rejection of the *null hypothesis 5*.

In testing the *null hypothesis 6*, we find that, with the Combine and Join structures (*v-structure* in Figure 6.5a and 6.5b), observing different nodes results in no significant difference in accuracy of the determination of states. This suggests that observing a different variable node in the *v-structure* graphs does not change the overall accuracy significantly. The observation of the best variable node in these structures may therefore depend on the other aspects, such as the financial aspect, which is not in the scope of our experiment. On the other hand, the significance tests in the Divide and Slice structures result in rejection of the *null hypothesis 6*, which suggests that these structures can affect our choice in observing a variable node for better accuracy across the food supply chain.

In addition, we also compare the average accuracy between different paths in the non-linear FGs. This aims to explore the accuracy of the states determined by *Belief Propagation* based on the path in the food supply chain. Our motivation in this experiment is that sampling based on any variable node in a particular path may result in the same accuracy in a different path. This is because food usually travels in a batch and food products in the same batch and same path are often treated equally. Here, we hypothesise that the accuracy of inferences made by *Belief Propagation* to determine the food states in different structures of the non-linear FG depend on which path is observed. The result of this experiment is presented in 6.8.

Figure 6.8 shows the difference in the average between non-linear structures of the FGs in Figure 6.5. We can see that the Combine and Join structures have a similar pattern. Observing a variable node to infer another variable node in the same path always produces the higher accuracy, especially before the Combine or Join processes (*path1* and *path 2* in Figure 6.8a and 6.8b). The same trend is also shown in the Divide and Slice structures (*path1* in Figure 6.8c and 6.8d as the path before those two branches).

In general, the accuracy in the Join structure is less than that in the Combine structure in *v-structure*. This is expected since a joining process drastically changes the distribution of contaminated and uncontaminated food while a combining process only collects the food products into the same location without drastically changing the distribution. In contrast, the accuracy in the Divide structure is higher than in the Slice structure even though the dividing process only divides the food products into different locations

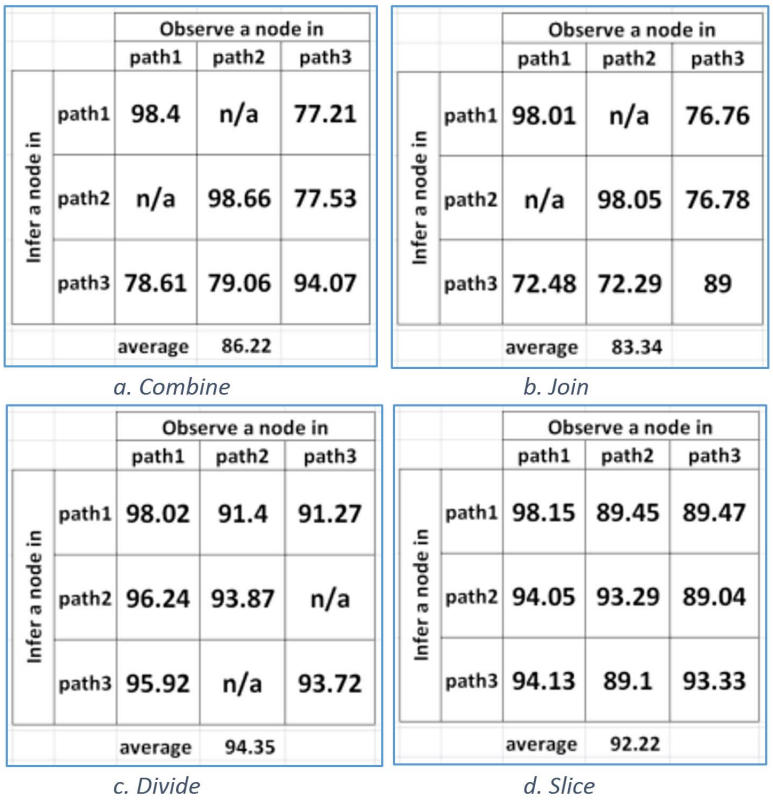


FIGURE 6.8: The average accuracy based on the path observed in the non-linear factor graphs in Figure 6.5.

without changing the distribution of contaminated and uncontaminated food. The process of slicing changes the distribution of bacteria in each food to be smaller in each slice of food. Thus, even though the quantity of the food products is doubled through the *slicing* process, the colonies of bacteria are reducing in each food product.

Finally, based on the average accuracy for each structure, we conclude that a food supply chain with a Join structure is the one in which food states can be least accurately determined by *Belief Propagation* technique, while a food supply chain with a Divide structure is the one in which food states can be determined with the highest accuracy using *Belief Propagation*. This means that a Join structure has the highest uncertainty (hence, is riskier) compared to the other structures.

6.3.2 Experiment 4

Our last experiment, Experiment 4, is similar to our Experiment 2, where we investigated the accuracy of inference by *Belief Propagation* to determine the states in the food supply chain without the *cooking* process (*c1*) that affects the distribution in food stage ***F_c1***. In this experiment, however, we deal with non-linear food supply chains. We present our null hypotheses to be tested as follows:

Null Hypothesis 7. There is no difference in the accuracy of the states inferred by *Belief Propagation* for different structures of non-linear *factor graph*s without a cooking process.

Null Hypothesis 8. There is no difference in the accuracy of the states inferred by *Belief Propagation* when observing each different node in a non-linear *factor graph* without a cooking process.

Results and discussion. As expected, Figure 6.9 shows a similar pattern as that in Figure 6.4, where the average accuracies are symmetrical in each non-linear FG. This confirms that the *cooking* process is the most influential process in both the linear and non-linear food supply chains in our experiments.

To test the *null hypotheses*, we perform significance tests (ANOVA or Kruskal-Wallis) between each structure (*null hypothesis 7*) and between an observed variable node (*null hypothesis 8*). The result of testing *null hypothesis 7* is the rejection of the null hypothesis, in other words suggesting a significantly different accuracy of *Belief Propagation* in each structure. In contrast, *null hypothesis 8* could not be rejected. This suggests that there was no significant difference in *Belief Propagation* when observing a different variable node in each of the structures. These results are presented in Table 6.5

	Non-linear (Combine)	**Non-linear (Join)**	**Non-linear (Divide)**	**Non-linear (Slice)**
Null Hyp.8	*3.44457197378442e-10*			
Null Hyp.9	*0.248*	*0.819*	*0.235*	*0.058*

TABLE 6.5: The *p-values* significance test in the non-linear factor graphs in Figure 6.5 without a *cooking* process.

Finally, we also experiment with the average accuracy between different paths in a non-linear FGs. Essentially, this has the same purposes as in Experiment 2, but without a cooking process in the non-linear food supply chains. The result of this experiment is presented in 6.10.

		Observe								
		F_i1	F_r1	F_t1	F_i2	F_r2	F_t2	F_co1	F_s1	F_p1
Infer	F_i1	100.00	98.24	95.75	-	-	-	95.75	93.24	86.61
	F_r1	98.24	100.00	98.5	-	-	-	98.5	95.99	92.65
	F_t1	95.75	98.5	100.00	-	-	-	100	98.48	96.21
	F_i2	-	-	-	100.00	98.31	95.54	95.54	92.94	85.92
	F_r2	-	-	-	98.31	100.00	98.22	98.22	95.62	91.37
	F_t2	-	-	-	95.54	98.22	100.00	100	98.4	95.9
	F_co1	95.75	98.5	100	95.54	98.22	100	100.00	98.44	96.06
	F_s1	93.24	95.99	98.48	92.94	95.62	98.4	98.44	100.00	98.61
	F_p1	90.97	93.72	96.21	90.45	93.13	95.9	96.06	98.61	100.00

a. The average accuracy of inference in the non-linear Combine structure in Figure 6.5(a) without the cooking process

		Observe								
		F_i1	F_r1	F_t1	F_i2	F_r2	F_t2	F_jo1	F_s1	F_p1
Infer	F_i1	100.00	98.1	95.47	-	-	-	79.4	79.4	79.4
	F_r1	98.1	100.00	97.88	-	-	-	76.99	76.99	76.99
	F_t1	95.47	97.88	100.00	-	-	-	74.37	74.37	74.37
	F_i2	-	-	-	100.00	97.81	95.5	79.4	79.4	79.4
	F_r2	-	-	-	97.81	100.00	98.2	76.7	76.7	76.7
	F_t2	-	-	-	95.5	98.2	100.00	74.4	74.4	74.4
	F_jo1	64.07	72.37	77.86	64.06	73.93	77.6	100.00	98.84	97.18
	F_s1	58.03	63.56	71.92	58.04	65.23	71.34	98.84	100.00	98.84
	F_p1	54.96	57.93	63.78	54.96	57.19	63.77	97.18	98.84	100.00

b. The average accuracy of inference in the non-linear Join structure in Figure 6.5(b) without the cooking process

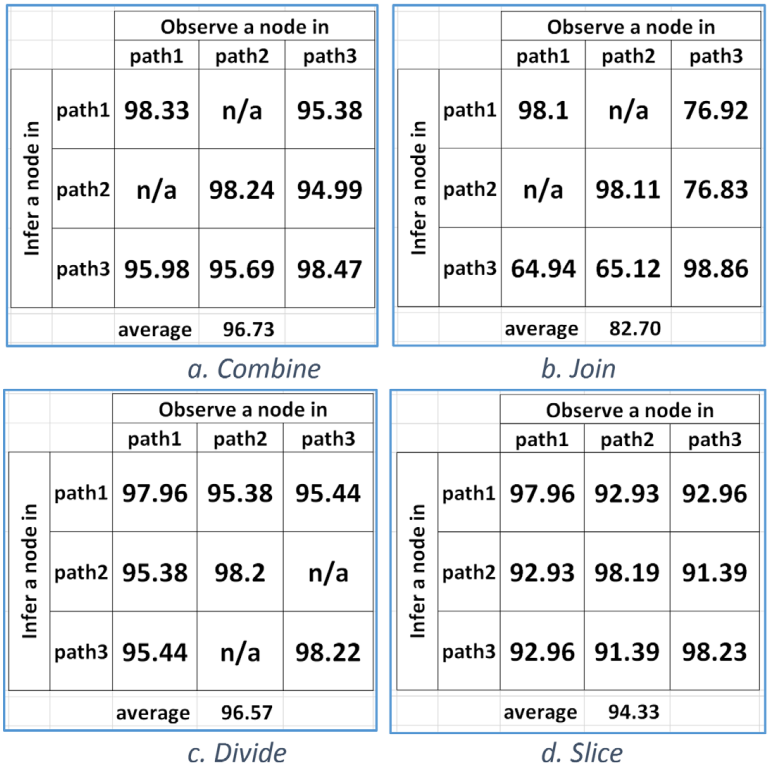
		Observe								
		F_i1	F_r1	F_t1	F_di1	F_s1	F_p1	F_di2	F_s2	F_p2
Infer	F_i1	100.00	97.9	95.17	95.16	92.65	90.35	95.18	92.83	90.41
	F_r1	97.9	100.00	97.77	97.79	95.28	92.98	97.75	95.4	92.98
	F_t1	95.17	97.77	100.00	100	97.99	95.7	100	98.15	95.73
	F_di1	95.16	97.79	100	100.00	97.99	95.7	-	-	-
	F_s1	92.65	95.28	97.99	97.99	100.00	98.21	-	-	-
	F_p1	90.35	92.98	95.7	95.7	98.21	100.00	-	-	-
	F_di2	95.18	97.75	100	-	-	-	100.00	98.15	95.73
	F_s2	92.83	95.4	98.15	-	-	-	98.15	100.00	98.09
	F_p2	90.41	92.98	95.73	-	-	-	95.73	98.09	100.00

c. The average accuracy of inference in the non-linear Divide structure in Figure 6.5(c) without the cooking process

		Observe								
		F_i1	F_r1	F_t1	F_sl1	F_s1	F_p1	F_sl2	F_s2	F_p2
Infer	F_i1	100.00	97.78	95.16	92.7	90.15	87.88	92.69	90.18	87.94
	F_r1	97.78	100.00	97.88	95.42	92.88	90.6	95.41	92.91	90.67
	F_t1	95.16	97.88	100.00	98.04	95.5	93.22	98.03	95.53	93.29
	F_sl1	92.7	95.42	98.04	100.00	97.95	95.68	95.76	93.43	91.35
	F_s1	90.15	92.88	95.5	97.95	100.00	98.23	93.39	91.25	89.32
	F_p1	87.88	90.6	93.22	95.68	98.23	100.00	91.28	89.28	87.49
	F_sl2	92.69	95.41	98.03	95.76	93.39	91.28	100.00	98	95.76
	F_s2	90.18	92.91	95.53	93.43	91.25	89.28	98	100.00	98.26
	F_p2	87.94	90.67	93.29	91.35	89.32	87.49	95.76	98.26	100.00

d. The average accuracy of inference in the non-linear Divide structure in Figure 6.5(d) without the cooking process

FIGURE 6.9: The average accuracy of inference in the non-linear factor graphs without the *cooking* process in Figure 6.5.



a. Combine

Infer a node in	Observe a node in path1	Observe a node in path2	Observe a node in path3
path1	98.33	n/a	95.38
path2	n/a	98.24	94.99
path3	95.98	95.69	98.47
average		96.73	

b. Join

Infer a node in	Observe a node in path1	Observe a node in path2	Observe a node in path3
path1	98.1	n/a	76.92
path2	n/a	98.11	76.83
path3	64.94	65.12	98.86
average		82.70	

c. Divide

Infer a node in	Observe a node in path1	Observe a node in path2	Observe a node in path3
path1	97.96	95.38	95.44
path2	95.38	98.2	n/a
path3	95.44	n/a	98.22
average		96.57	

d. Slice

Infer a node in	Observe a node in path1	Observe a node in path2	Observe a node in path3
path1	97.96	92.93	92.96
path2	92.93	98.19	91.39
path3	92.96	91.39	98.23
average		94.33	

FIGURE 6.10: The average accuracy based on path observed in the non-linear factor graphs without a *cooking* process (***c1***) in Figure 6.5.

6.4 Summary

In this chapter, we performed some experiments with *prFrame* in respect to the different structures of FGs. Generally, the experiments serve two aims. First, to combine each scientific approach in Chapter 3 systematically as a single completed general framework, *prFrame*. Second, to see if *prFrame* can provide us with intuitive and relevant results over imaginary scenarios with different possibilities of product supply chains. With the historical datasets from the literature of our domain of interest, food, we establish the experimental setup and run the experimental method. Note that we also establish an assumption, which is common in the food domain (i.e., the possibility of uncontaminated food becoming contaminated).

Our experiment with linear structures has demonstrated that the risk of contamination can be propagated throughout the food supply chain. *prFrame* is able to show the changes in risk caused by the different processes, based on the states observed in the food supply chain. Importantly, the results are intuitive and relevant to what would be expected on the basis of the risk factors and assumptions. In addition, the accuracy of the states predicted by *Belief Propagation* follows the general expectations in respect to food contamination.

The experiments with non-linear food supply chains convey a similar conclusion to those for the linear food supply chain, which provides further support to the contention that *prFrame* can be used as a systematic risk assessment approach for demonstrating due diligence. *prFrame* has demonstrated its effectiveness through the results in the range of realistic structures, batches and paths in the non-linear food supply chains. Finally, the experiments also provide statistical support for our claim in the form of significance levels and confidence intervals.

Chapter 7

Discussion and Evaluation

In this chapter, we discuss and evaluate our systematic framework to assess risk, *prFrame*, based on the results of our experiments and the general intuition of the product supply chain. Our discussion comprises all of the techniques we have presented in previous chapters and how they are combined in *prFrame*. In addition, we evaluate the framework based on our case study in the specific domain, food, to see the applicability of the framework in the real world.

7.1 *prFrame* framework

To remind the readers, the three techniques we described and explained earlier in Chapter 3 are our approaches to assess the risk in a product supply chain scientifically and systematically. We refer to this systematic mechanism to assess risk as the *prFrame* framework. Overall, this framework encapsulates the notion of *Provenance*, *Risk* and *Probabilistic Graphical Modelling* (PGM), which are the three pillars in *prFrame*. Provenance contributes to the modelling of a product supply chain by capturing the lineage of a product. The notion of risk provides a mechanism to calculate the risk of undesired events across the provenance-based supply chain. Finally, the concept of PGM administers the fundamental theories of probability and graphs to perform probabilistic propagation as an inference task based on the *Provenance Graph* (PG) representation.

prFrame begins with a given PG that is a result of the integration process between the provenance of a product and the risk models with their associated risk factors. The integration aims to overlay the provenance of the product with risk models to explore and understand a potential risk across the product supply chain. In the end, it generates a PG with annotated risk information on it that models the lineage of a product and its potential risk.

As a result of the integration, the integrated provenance-risk graph is expected to hold the necessary information about the risk, such as the risk models and the risk factors. That information is captured as parameters to represent what had happened to a product and how to calculate its risk in every process of its supply chain. The risk calculation is performed by using the iterative Monte-Carlo (MC) simulation to consider variability and uncertainty in risk calculation from the past or historical data. Here, the description of the past event about the risk factors is in the form of a distribution. At the end of the simulation, the distribution of the predicted output values are generated from which Conditional Probability Tables (CPTs) will be constructed for each process in the product supply chain.

The generated CPTs are annotated back to the initial PG before the graph is converted into a *Factor Graph* (FG). The conversion from a PG to an FG allows efficient inferences by means of the *Belief Propagation* technique, where the risks are passed between the variable and factor nodes in the FG. The principle of the conversion follows the concept of *d-separation* in order to preserve the probability distribution among variable nodes. Depending on the structure in the PG ($-2O$, $-2M$, $O2O$, $O2M$, $M2O$, and $M2M$), a prov:Activity (product process) may get converted into a single or several factor nodes. Also, a prov:Entity (product stage) is converted to a variable node in the *one-to-one* mapping (a single prov:Entity for a single variable node).

Once the converted FG is ready, the inference task can be performed. The inference is achieved by propagating the risk across the FG. This probabilistic propagation is handled by the *Belief Propagation* technique, which works on the principle of the *Message Passing* algorithm to pass messages forwards and backwards along the FG. Since the messages are the probability distribution between nodes, the *Sum-Product* algorithm is used to propagate the probabilities appropriately. Finally, this technique allows us to infer the unobserved variable nodes with the partial information available in the observed nodes, which helps us to identify the risk in the product supply chain.

7.2 Provenance as the core for *prFrame*

Incorporating the concept of provenance in *prFrame* helps the modelling process to be more credible and accountable because the modelling will be based on the actual events instead of the general processes or locations of the product supply chain. For example, the general product supply chain may only indicate the locations to which a product is travelling to, but lack of information about what the processes are in those locations and what risk is involved in those processes. Moreover, any change in the product supply chain that can alter its structure is also captured in the provenance records. The ability to capture the actual events at actual places makes a provenance-aware system a powerful tool to promote trust in subsequent reasoning and inferences derived in respect to risk.

As a result, the provenance-based product supply chain by *prFrame* has considerable benefits compared to merely a general description of the product supply chain.

In modelling the product supply chain based on a product's provenance, *prFrame* focuses on the processes that a product has passed through. This approach modelling with provenance is called a *process-centred provenance*, where actions or activities are the main concern when capturing the product's provenance. Modelling the provenance by capturing the processes ultimately helps *prFrame* to identify the most influential processes in the product supply chain. Each process is modelled with PROV Model as prov:Activity, which uses (prov:used) and generates (prov:wasGeneratedBy) a product as a prov:Entity in the existing of the derivation (prov:wasDerivedFrom) of the *used* and *generated* prov:Entity. In addition, all the details about the processes and their risk (a risk model and its risk factors) are annotated in prov:Activity and will be used as the important inputs for risk calculation later. Eventually, modelling the product supply chain with the *process-centred provenance* approach will generate a product supply chain with a set of processes as the main focus to describe the lineage of product.

In this research, the naming of prov:Activity and prov:Entity are designed to intuitively represent the development of a product *stage-by-stage*. For example, prov:Entity in Figure 2.5 shows the stages of food before it becomes its final product. Those stages are **cooked meat**, **boiled spaghetti**, **cooked sauce meat** and **spaghetti with cooked sauce meat**. In the same Figure, the naming of prov:Activity is designed to be an action verb (i.e., *cooking*, *boiling* and *mixing*), and the *generated entities* of each process are named after those actions (i.e., **cooked**, **boiled** and **mixed**). This intuitively produces a PG with a set of actions and stages, making the identification of the stages and their associated process is easier. In addition, this procedure allows us to apply an inference rule through the use of *usage*, *generation* and *derivation* relations. The kind of inference is administered by PROV-CONSTRAINT[1] as one of the building blocks in PROV (See section 2.2.1).

With this approach, the product supply chain in *prFrame* is a representation of the lineage of product, which only captures the actual processes the product has gone through. In addition, new processes, which may not be expected to happen before, are captured in the provenance records if those processes are actually happen. This use of provenance can therefore serve as a verification tool to validate the actual product supply chain against the description of the product supply chain described in the documentation.

Another advantage of provenance in *prFrame* is to determine where responsibility lies for processes. In the product supply chain, the same process may be performed by different operators at different locations. In this case, *prFrame* will distinguish those processes allowing identification of who is responsible for which process. This means that if an

[1] `http://www.w3.org/TR/2013/REC-prov-constraints-20130430/`. Accessed: 28 June,2020

undesired event occurs, the right operators can be identified quickly by querying the provenance records.

In addition, the provenance of a product recorded through *prFrame* will help the operators and enforcement authorities comply with regulations. We have seen this in some of our exercises in the food domain when we developed the *prFood* ontology in Section 5.2. Basically, *prFood* helps *prFrame* conceptualise the domain of interest, (i.e., food), by providing the terminologies regarding the guidance and regulations. *prFood* is also an important *hub* to map the other legacy ontologies, making data exploration possible at a massive scale. We also evaluate this ontology by checking it against some regulations in its domain. These include documentation, HACCP-based monitoring, and traceability with the concept of an *one-up/one-down* check.

7.3 Risk assessment with *prFrame*

The notion of risk assessment in *prFrame* helps to identify the higher risk processes in the system, such as online money transactions, news productions or product supply chains. Risk assessment can be a tool to increase awareness of undesired events. In this sense, risk can be seen as something that is measurable and its measurement is often modelled by experts as risk models. By using these risk models, operators and product authorities have some insights about the risks of failure in the system they are involved in. Utilising risk assessment in the *prFrame* will therefore support the notion of due diligence, because measuring risk can be considered as a necessary action to prevent an undesirable event from happening. Moreover, by utilising a risk model to assess risk, one is more likely to be able to target the actual cause of a problem as opposed to just guessing it. This is more preferable as it is based on a methodologically proven approach, which can be used as a piece of evidence to demonstrate due diligence.

By integrating differing levels of risk with the provenance records *prFrame* is able to understand how risk develops throughout the lineage of a product. To remind the readers, *prFrame* is concerned with the development of risk in a product supply chain; therefore, the concerned risk is a risk of product failure throughout the product lineage. In this section, the lineage of a product from its production to its consumption consists of processes that are modelled with the PROV as a set of prov:Activity. In each prov:Activity, the risk models and their risk factors are annotated by using PROV-O and the specific ontology as its extension. In the case of food, *prFrame* uses the *prFood* ontology to capture all the information necessary to construct the food supply chain and the risk of contaminations at all the points within it.

Through this approach, we gain the benefit of having the product supply chain with its associated risk in the standardised and general form of the provenance records. Figure 7.1 shows an example of integrating the provenance of a product and its risks to construct

the product supply chain. In this example, we take the details of the risk assessment from WHO/FAO in their report about *Salmonella* contamination in broiler chicken. Note that this example uses the same risk factors as in Table 6.1 when we experimented with the accuracy of *prFrame* in Chapter 6.

Figure 7.1 shows the integration of chicken's provenance with the risk of *salmonella* contamination across it. The type of each prov:Entity is captured as the properties in prov:type to indicate the stage of chickens. For example, prFood:RetailedFood indicates that the chickens in this stage are the result of a *retailing* process which is described in prFood:stageDetails. In modelling, *prFrame* will represent a prov:Entity as a stage of chicken after a particular process (prov:Activity) and it is named after that process. Appendix A defines all the terms in Figure 7.1.

As mentioned, the risk models and their associated risk factors are both captured in the prov:Activity. All risk models are captured in prFood:modelCode and the basic module of MPRM is captured in prFood:moduleOf. For example, a *storing* process, **s1**, is modelled with the risk model grw01b and this process causes the growth of *salmonella* (prFood:Growth). In addition, the associated risk factors of this process are captured in prFood:homeDurDist and prFood:homeTempDist, which are the inputs for the risk model grw01b to calculate the distribution of predictive output values of food process **s1**. The result of the calculation will be an input to the process **p1** and this process continues until the end of the supply chain. This repetitive process with a large number of chickens is one of the reasons for using the Monte-Carlo (MC) simulation.

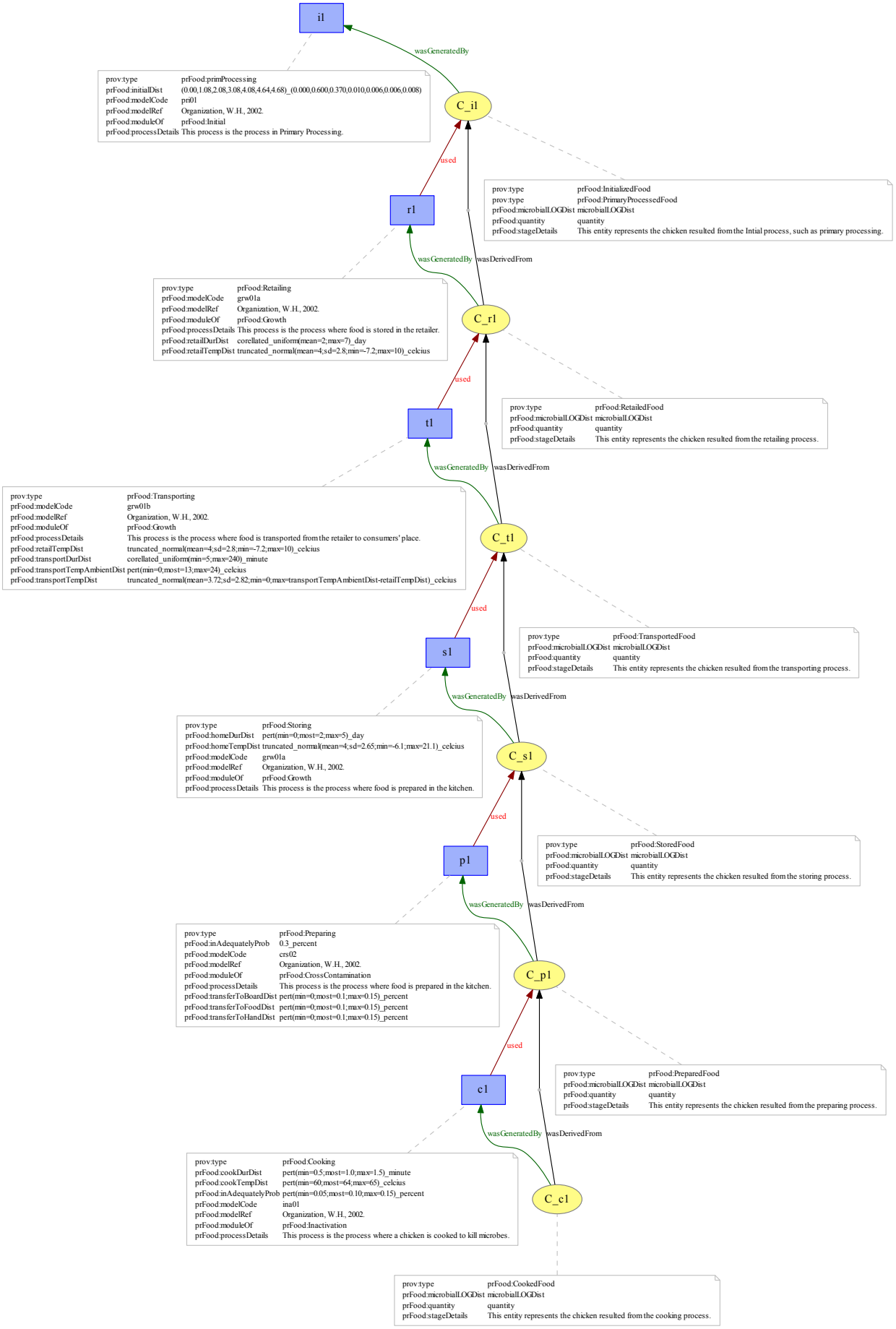


FIGURE 7.1: A provenance graph of the chicken supply chain with the risk of *salmonella* contamination.

The PG in Figure 7.1 is the input for the iterative MC simulation to instantiate the distribution of predictive output values. The use of the distribution in assessing the risk is important in order to represent uncertainty and variability, which are the two aspects inherited from risk. In this risk context, uncertainty can be seen as the lack of perfect knowledge resulting from limited observations of the parameter value; this lack of knowledge may be reduced by further measurements. A variability, on the other hand, can be seen as a true heterogeneity of the population that is a consequence of the measurement (e.g., quantitative measurement). Thus, the use of a probability distribution in assessing risk is deemed as a good approach because it represents uncertainty (with a probability of occurrence) and variability (with a range of possible values). Further, as explained in Section 4.2, the Conditional Probability Table (CPT) is attached to its prov:Activity.

From our exercise, we can see that *prFrame* helps the existing risk models, developed by their experts, to be integrated with the provenance of a product. Since the integration relies on the provenance of a product, the result of the risk calculation becomes more relevant and contextual because the result is derived from the actual events of the product. Moreover, identifying the cause of the higher risk is easier because provenance captures the possible risk factors in each process. To remind the readers, a risk factor is anything that contributes to the development of risk. It has a direct influence on a risk model, which is a mathematical expression to estimate risk. The cause of the higher risk calculated by a risk model can therefore be tracked down to its risk factors. Moreover, the specific product operators of the higher risk can be identified as all the processes with a higher risk that are captured in the provenance records.

prFrame is designed to be more general than risk ontology. That is because the risk is domain-specific, even if risks are similar in principle, they can be differently defined or modelled by different users. At present, *prFrame* framework only considers the risk of food contamination in the food supply chain. Therefore, the semantic of *prFood*, currently only aims to capture food-contamination-related information. However, *prFood* can always be enriched by adding additional semantics that considers a different type of risk in the food supply chain, for example, risk in food quality, food waste, carbon footprint, supply/demand availability, etc. In other words, an additional type of risks can be incorporated in *prFood* as long as we can quantify them through a set of risk models. Another consideration to think about when adding a new type of risk is where the information of risk factor should be annotated/captured in, as risk factor can be both annotated in prov:Entity, prov:Activity, and prov:Agent. In fact, ontology can be extended to capture any information; hence, it is a powerful tool to capture information for risk analysis.

In our use cases, we demonstrate the risk of food contamination and this type of risk comes after certain action or food process. Therefore, we associate the risk models and risk factors by annotating them in prov:Activity, rather than in prov:Entity. Moreover, the food supply chain is often depicted as a set of processes from raw material to the final

product. This is suitable when modelling the product's provenance with *process-centred provenance*, where actions are the main concern to result in product's stage/phase. With this reason, it is suitable to capture the product's process as prov:Activity and product's stage as a risk model.

Finally, calculating risk in *prFrame* framework considers the structure and the rich of semantic in the set of ontologies. *prFrame* framework has *prFrame* ontology that concerns about the stages and the structure of the product supply chain. We have seen it through prFrame:Stage, that represents each phase in a product's lifetime. On the other hand, the risk of a product is captured in the product's process through the rich of semantic in specific product ontology. As mentioned, the risk is more domain-specific and not a part of *prFrame* ontology to give more flexibility to another risk domain-specific ontologies (such as *prFood* for food) to capture their risk.

7.4 Propagation of risk in *prFrame*

prFrame has been developed to assess risk quantitatively, which often refers to a Quantitative Risk Assessment (QRA). QRA provides a solid and scientific approach to measure risk through mathematical expressions such as a risk model and its inputs as risk factors. Because of uncertainty and variability in risk assessment, the probability distributions are often used to represent risk, and MC simulation is performed to take into account all the possible values in those distributions.

This approach alone is limited to the additional data actually available, however (dataset used in MC approach is derived from the past studies). Even with additional data that represents the most actual and relevant information, MC simulation does not significantly update the result of the risk assessment, because this technique simply relies on the distributions of the predicted output values and risk factors. In addition, the input-output interaction of MC simulation only works in one direction (forward assessment); making a prediction or inference of the risk under uncertainty before the process (backward assessment) is more difficult. To address this limitation, *prFrame* incorporates an inference technique adopted from the PGM domain to be able to assess risk in both directions, forwards and backwards.

Inference is about learning properties of the world from data that we observe. As we have seen the inference technique that utilises the structure of a PG (through *usage*, *generation* and *derivation*) in Section 7.2, the inference technique in this section quantifies the probabilities based on the structure of the provenance-based *Factor Graph* (FG) through PGM. This is based on the recognition that it is natural to have some limitations in an observation due to lack of understanding, data availability, or some phenomenon that is difficult to explain. The result of inference is therefore often represented as a

probability distribution that reflects the inherent randomness (i.e., variability) and uncertainty. Because of this ubiquitous and fundamental uncertainty about the truth, it is necessary to allow an inference system to consider different possibilities as a model of the world.

When uncertainty is involved in an inference task, one may conclude that some events (e.g., rain or not, traffic jam or not, etc.) are more likely to happen than others. Generally, uncertainty is often followed by an adaption of our beliefs regarding the occurrence of an event. For example, we are more likely to believe (to some degree) that person ***A*** has murdered person ***B*** if we have sensed a rivalry between them. This degree of belief increases when evidence (e.g., knife, gun, etc.) is discovered with person ***A***'s fingerprints on it. This subjective attitude towards a certain idea or concept can be described as a degree of belief and can be represented as a probability (chance or odds) distribution. This term is interchangeable with the term confidence in the sense that if we have a higher degree of belief, we are more confident that a certain event will occur as expected. In contrast, we can say that the more uncertain we are in respect to the occurrence of an event, the lower the degree of belief we have; hence we have less confidence. We have seen this phenomenon in the distribution of predictive output values generated by the risk models and the range of values in the risk factors in our experiments in the previous chapter (Chapter 6). Basically, those probability distributions map our degree of belief and can be used to establish assumptions and perform an inference task on the basis of those assumptions.

In our experiment, we also investigate the accuracy of *Belief Propagation* as our inference technique. Here, accuracy is defined as the correctness of outputs appropriate to the context where it is represented that can affect the degree of belief in an idea or concept. In our experiment, it reflects the correctness between the output values predicted by MC simulation and the states inferred by *Belief Propagation*. Generally speaking, the highest accuracy is shown as 100% correctness and the lowest is shown as 0%. For instance, the average accuracy of $P(\boldsymbol{F\_i1}|\boldsymbol{F\_r1})$ is higher than $P(\boldsymbol{F\_i1}|\boldsymbol{F\_t1})$ based on the result of our experiment (Figure 6.2a). As mentioned, we conclude that the higher accuracies also represent higher belief and confidence; and thus less uncertainty.

The final objective of *prFrame*, as a systematic approach in demonstrating due diligence, is to assess risk based on the provenance of a product across its supply chain. Since risk can change, we investigate whether utilising *Belief Propagation* as a provenance-based inference technique, produces accountable, credible and relevant results that agree with general intuition. To produce credible and accountable results, *prFrame* should have a set of sufficient evidence that can be trusted or believed as providing an explicable answer. Those aspects are supported by the notion of provenance to capture and record the history or lineage of a product. Moreover, a relevant result means that it agrees intuitively within the context; hence, the approach, i.e., *prFrame*, is potentially useful and applicable in solving real-world challenges.

In performing an inference, *prFrame* aims to derive a conclusion based on our observation of the states. Intuitively, the more observations, the more relevant and contextual the result of inference is. Our illustration of being late to school in Chapter 1 shows that intuitively. We remind the readers that the observations we are referring to here are observations of the states (e.g., *absent* or *present*, *being late* or *not late*, etc.), and not observations of the details of the risk factors (e.g., distribution of temperature, speed, minutes, etc.). The observations are used as the evidence in Bayes' Theorem, where Conditional Probability Distribution (CPD) is used to find the probability distribution of unobserved random variables (i.e., late to school), given the observed random variables (i.e., accident on the road). By capturing and representing those random variables in a PG, *prFrame* can produce more credible, accountable and relevant inference results. Ultimately, those results provide better insights because they are based on more actual facts captured in the provenance records. Hence, they are intuitively and contextually relevant for decision-making procedures and finding a reason for (reasoning) or a cause of a certain event.

We tested *prFrame* through a set of experiments in Chapter 6 to see if it can produce more intuitive results. Specifically, we evaluated the probability distribution of a set of variable nodes (unobserved), given the values of another set of variable nodes (observed). This evaluation relies on the *Belief Propagation*, as a general *Message Passing* technique to propagate the distributions across the FG by manipulating the *sum rule* and the *product rule* (through the *Sum-Product* algorithm). As we know, these are the basic rules mathematically to calculate the probability distribution efficiently. That efficiency is achieved by factorising the joint distribution between those random variables in order to query the conditional distribution that allows reasoning patterns (e.g., explanation, prediction, inter-causal reasoning, etc.).

The first intuition to be checked relies upon the concept that our belief in something (prior belief) is changed when we start looking at the data (i.e., observing the evidence). Depending on the observed value in the variable nodes, uncertainty may increase or decrease, which feeds through to a change in our belief as well. For example, in Figure 6.1a, our belief that variable node ***F_p1*** has the state of *absent* is increased when we observe that variable node ***F_r1*** has the state of *absent*. The increment is from [71%:21%][2] without any observation, to [94%:6%] by observing variable node ***F_r1*** as *absent*. In addition, in Figure 6.5b, observing variable node ***F_t1*** as having a state of *absent* will increase our belief that variable node ***F_jo1*** will also have a state of *absent*. Although belief in the state of *absent* for variable node ***F_jo1*** increases (from [43%:57%] without observation, to [53%:47%] by observing ***F_t1*** with *absent*), uncertainty remains relatively high because the prior distribution of *absent* and *present*, without observations, was quite close. The experiment also includes some processes that change the distribution dramatically, such as the *cooking* process. Since those processes produce such an

[2] In this format the first percentage figure represents our belief that the variable node has a state *absent*, while the second represents our belief that the variable node has a state *present*.

extreme distribution, performing inference in a variable node ***F_c1*** will produce high accuracy (higher than 96% in Figure 6.2 and Figure 6.7); hence, increasing our belief and confidence about the occurrence of an event (whether it is *present* or *absent*).

From this simple exercise, we conclude that the closer the probability to a *fifty-fifty* (50%:50%) chance, the more uncertainty we exhibit and the more our belief in the occurrence of an event deteriorates. This can be validated through the Confidence Interval (CI) in Figure 6.3 and Figure 6.6. The lower accuracy relates to lower confidence (wider CI) because of higher uncertainty. Note that, the *fifty-fifty* chance is based on the two states available for the variable nodes, namely *absent* and *present*. In the context of the product supply chain, this experiment shows that the risk of product failure can change based on the data observed throughout its production and distribution. Thus, it is necessary and important to explore how each process in the product supply chain affects the distribution of a product in order to establish some preventive actions or take decisions to mitigate the risk. For example, a regular inspection or monitoring system (e.g., sensor) is suggested to be implemented in the processes that bring more uncertainty (i.e., a process with a distribution of a product failure 50%:50%).

We also take our experiment further through measuring the accuracy of inference by *Belief Propagation*. As described in Chapter 6, the accuracy is measured by comparing the predictive values generated by MC simulation with the inference result from *Belief Propagation*. Here, we investigate the accuracy of states when distances between observed and unobserved variable nodes are wider when performing an inference task with *Belief Propagation*. The results reveal that the wider the distance, the less accurate the inference of states by *Belief Propagation* (see Figure 6.2, 6.4, 6.7, 6.9). In other words, the more uncertainty in between observed and inferred nodes, the less accurate the inference is. Intuitively, this means that the longer the distance between the source of information and the location of inference, the less accurate the result of inference by *Belief Propagation*. It also implies that a ripple of uncertainty travels through the product supply chain. This ripple of uncertainty accumulates and increases risk as more uncertainty exists in between the observed and inferred nodes.

In this experiment, we also investigate the structure of the graph. As we know, the structures of the graph we are concerned with are either linear or non-linear, reflecting the different ways in which the product supply chain is constructed. We therefore observed how risk is developed or changed based on the structure of the graph. We started with the linear structure because it is a simple structure in the product supply chain, capturing only the general processes that the product must have gone through. Specifically in the food domain, most of the food risk models have been developed by experts to measure the risk of contamination in a linear food supply chain. In a linear chain, each variable node has exactly one directed edge pointing to it, which provides us with the causal effect to answer our reasoning questions. Basically, the inference in a linear graph is an extension of the MC-simulated QRA with an ability to assess

risk in any direction. In addition, the experiment in a linear graph guarantees that a product must undergo all of the processes described in the product supply chain. This means that if there are 1,000 products at the beginning of the chain, then those 1,000 products will go through the same processes in the product supply chain. In this case, the factorisation is easier since the quantities of the product in all processes are equal.

In the non-linear graphs, meanwhile, the factorisation is slightly different from the linear graph because the quantity of the products may not be equal between the input and output of the processes. The non-linear or branching structure can result from the nature of the product processes or the distribution of the product in its supply chain. We discussed this when we differentiated the non-linear *dirE* and *indE* structure in Section 4.3. In short, the *dirE* structure represents an equal quantity of input and output for the processes, while in the *indE* structure outputs may not equal inputs.

Non-linear structures are common in many actual product supply chains, since the products are processed and distributed to many locations in many batches. As a consequence, the PGs will often show the non-linear structures because provenance always captures the actual product supply chain. Subsequently, factoring the joint distribution between the inputs and outputs of a product is challenging since the quantity of products between the input and the output of the processes are not equal with the non-linear structure. Hence, *prFrame* is developed to consider the type, location and batch of a product across the product supply chain. With those considerations, *prFrame* is able to calculate and propagate risk appropriately.

Our experiment with the non-linear structures also aims to identify which path we should investigate further in the non-linear supply chain. In the actual supply chain, the products in one path may not be treated equally in the other paths, even when both of the paths have the same processes. Clearly, this is because different product operators may apply different parameters or configurations to the product they handle. These parameters are captured and annotated in the prov:Activity as the risk factors in the PG. Consequently, those processes may produce different distributions, which then produce different values in the CPTs. In addition, the notion of *d-separation* in probabilistic propagation can block the transfer of information from one variable node to the other variable nodes. Therefore, we may not interested to observe the variable nodes that do not contribute to the result in the inferred variable nodes. Based on this fact, we experiment with the different batches and paths in the product supply chain to identify which process in which path is the most suitable to observe. In this case, the most suitable process is perhaps the process that requires the least cost and resources to investigate compared to another path with a similarly accurate result. For example, if observing a product in stage A to infer its stage in stage C ($P(C|A)$) returns the same result as observing a product in stage B to infer its state in stage C ($P(C|B)$), a product operator may chose the cheaper observation between A or B.

Our last experiment is to investigate the processes that change the distribution significantly. The existence of these processes is important to reduce the uncertainty and hence increase the accuracy of prediction. This is because the distributions after these processes are in favour of a particular state. In the product supply chain, this type of process holds less uncertainty. As an illustration from our experiment, performing inference in the variable node ***F_c1*** and ***F_c2*** will produce a higher accuracy since food after the *cooking* process will mostly be uncontaminated. Intuitively, if we know that the uncertainty is low, we tend not to require monitoring or inspection at that point. In reality, it always depends on how crucial the process is to reduce risk in the specific domain. In addition, it also depends on whether the product operators are willing to take a risk by not monitoring these processes.

7.5 Expected and actual behaviour in the food supply chain

As we mentioned previously in Section 7.2, incorporating provenance in the product supply chain can potentially re-structure its topology as provenance records the actual events or incidents. Therefore, we can compare the product supply chain as planned against the provenance-based product supply chain (the actual product supply chain). In our experiment in Chapter 6, we intentionally design our food supply chain as simple as possible, yet covering the basic structures of food production and distribution. From our experiment, the $M2O_{dirE}$ procedure is the riskiest operation compare with the other structures. However, we only show the obvious (i.e., *joining*) in our experiment, while in practice, there can be many $M2O_{dirE}$, even as a sub-process of the main process. For example, *boiling* different type of food in the same pan will need to consider *boiling* process (to calculate the reduction number of bacteria) and *joining* process (to calculate the addition of bacteria that survive between food). This event obviously changes the structure of the graph topology to be more complex, as well as using more complex risk model (e.g., a risk model in *boiling*, a risk model in *joining*, or a risk model in *boiling* and *joining* as a single risk model). The fact that our experiment highly depends on the dataset available at that time, the structure in our experiment underestimates the actual structures of the food supply chain.

In addition, the risk factor used in the experiment assuming how people behave in an ideal circumstance, even with a certain proportion of contamination as a random contaminated food (See Section 6.1.2, when we set $\alpha = 1\% - 5.7\%$ as a random variable to inject contaminated food in our experiment). However, with the change in food menu or variation of food, it certainly changes how people prepare a certain/specific meal. Perhaps, a certain food needs to be prepared in a specific manner that does not reduce the number of bacteria as good as preparing it with a conventional manner. Therefore our $\alpha = 1\% - 5.7\%$ may need to be re-considered for different behaviour. Table 7.1 illustrates how our experiment may underestimating the food supply chain.

Food process	Expected behaviour	Actual behaviour	Possible change of risk for actual behaviour
Cooking	Temperature sensor in specific location	Temperature sensor too far	Higher than expected
Transporting	Delivery with in 20 minutes in summer	45 minutes in delivery in summer	Higher than expected
Transporting	Different temperature for different food	Same temperature for all food	Higher than expected
Storing	Lowest position for meat in a refrigerator	Meat above another food	Higher than expected
Storing	Once open, finish food within 2 days	Finish food within 5 days	Higher than expected
Storing	Control condensation in a storage	Lack of control of condensation	Higher than expected
Preparing	Do not wash meat	Meat is washed	Higher than expected

TABLE 7.1: The potential different between expected and actual behaviour in the food supply chain ecosystem.

Based on the aforementioned paragraphs, our experiments hold the assumption that we mostly underestimate the food supply chain (structurally and assumption). Although we start with underestimate structure and assumption, the use of provenance can gradually rectify the topology of the food supply chain and a set of assumptions holds in that supply chain. However, as the topology becomes more complex, the resource to collect more data and develop suitable risk model becomes more expensive.

7.6 Summary

In principle, the effectiveness of the methodology within this framework lies when new evidence can easily be incorporated to estimate the actual risk more accurately. Overall, *prFrame* comprises three major techniques, which have been discussed in the previous sections. These are incorporating the risk models and their risk factors with the provenance of a product, the MC simulation based on a PG, and a conversion from a PG into an FG (a bipartite graph containing nodes for variables and factors).

These techniques result in the establishment of *prFrame* to reason, estimate and understand risk across a supply chain for which we have only partial knowledge. It can provide a basis for operators and product authorities to develop a rationale for control procedures. Indeed, our discussions with them show that random inspections are costly in terms of resources, and a rationale needs to be developed on how best to inspect the product in its supply chain. Since inspecting and monitoring the product and managing risk is a key part of demonstrating due diligence, regulators and operators are constantly on the lookout for better ways to measure, track and analyse risk in the product supply chain.

In addition, provenance also supports the simple inference through the use of *usage*, *generation* and *derivation*. This feature in *prFrame* helps to construct the product supply chain, where the derivation of a product is the result of the generation of consecutive processes. In principle, therefore, provenance improves accountability and trust in the result of an inference task. As a basis to construct the product supply chain, provenance offers the dynamic construction of the product supply chain. For example, if a product

has for any reason gone through a process that was not originally in its supply chain, the updated supply chain with the new process is captured in the provenance records. This is because the notion of provenance always captures anything that happens to a product. As we demonstrated in our experiment (Chapter 6), the structure of the graph (linear and non-linear) can affect the result of an inference task, the quality of the result by performing *Belief Propagation* is better to use the most relevant and contextual graph that represents the product supply chain.

Finally, *prFrame* can potentially be a tool to support decision making. Generally speaking, it is often preferred to take a decision based on a scientific and systematic principle because the validity of that decision can then be accounted for. *prFrame* is therefore built to be a scientific and systematic approach. It uses three scientific pillars, Provenance, Risk and PGM to assess risk. In particular, the use of provenance can improve decision making because the evidence to make a decision is based on actual events in the past.

Chapter 8

Conclusions and Future works

In this research, we have discussed our systematic approach, *prFrame*, with the aim of demonstrating the concept of due diligence. Our review of due diligence shows that this term has a broader meaning and has been defined differently in many domains, but we conclude that due diligence can be described as meaning that everything that might lead to negative outcomes and could be anticipated, must be identified and controlled. This description promotes the concept of due diligence as a set of precautions or preventive actions, which essentially, becomes the primary reason to develop *prFrame*.

The above definition of due diligence motivates us to provide the product operators and enforcement authorities with a mechanism to understand their products' processes and have an overview of the risk around it. In this sense, our research only concerns due diligence in the product supply chain because of our interest in identifying and assessing the risk of product failure in its supply chain. We present the scientific approaches in *prFrame* as *Provenance*, *Risk* and *Probabilistic Graphical Modelling* (PGM), which then become the three pillars of *prFrame*.

Our modelling of the product supply chain in *prFrame* adapts PROV as a standard language with the concepts and their relations to facilitate the machine-processable data model for provenance. PROV is supported by a general provenance ontology PROV-O (Ontology) as one of its serialisation formats. Our specific ontology, *prFood*, can be seen as a sub-ontology of PROV-O in the food domain to derive the lineage of food as the food supply chain. In particular, the development of *prFood* also aims to measure risk across the food supply chain automatically. By this approach, we address our first research question *"How can we model the provenance of a product to support its regulation and its risk assessment?"*

We identify a focus on the risk of contamination by undesired bacteria across the food supply chain because the prevention of such an event is of interest to food authorities as a way of protecting consumers (hence, complying with due diligence) and can also save

time, money and energy. We introduce a case study in the food domain, where a general Quantitative Microbial Risk Assessment (QMRA) framework, called Modular Process Risk Model (MPRM) is used as a guideline to assess the risk of contamination. All of the information is annotated in the PG, which is essentially the product supply chain. With this approach, we address our second research question *"How can we overlay a product supply chain, based on its provenance, with the existing risk models?"*

Since risk is related to uncertainty and variability, the last pillar in *prFrame*, PGM, helps us to address our third research question, *"How can Belief Propagation be augmented with provenance?"* The propagation is done by incorporating the Monte-Carlo (MC) simulation to construct a Conditional Probability Tables (CPTs) in each prov:Activity as a basis to propagate the risk with *Belief Propagation*. *Belief Propagation*, as a *Message Passing* technique, uses *Sum-Product* algorithm to perform probabilistic propagation over the *Factor Graph* (FG); thus, the PG must be converted to an FG, and the factorisation of the joint distribution then takes place. Through our demonstration and validation of the case studies in Chapter 6 and Chapter 7, the results show that the change of risk intuitively represents the actual world. Thus, we conclude that *prFrame* can potentially be used as a systematic and scientific methodology to solve a real-world problem. This ends our conclusion by addressing our final research question, *"Is the approach relevant to the common product supply chain?"*

In designing *prFrame* framework, our motivation is to model a provenance-based supply chain to improve the relevancy in decision making. Moreover, the *process-centre provenance* approach is suitable when modelling a set of processes. The main driver for this approach in our case study is that the risk we concern emerges after certain activities, which we model in prov:Activity. Also, the *prFrame* ontology is designed to be more general to give more flexibility for other risk-specific ontologies to define their own risk. With this design principle, *prFrame* potentially be used in another domain than the product supply chains, such as emergency response, medical care, banking, block-chain, or crime. Because of its generality, *prFrame* framework should be able to be integrated with the more risk-specific domain.

Following our conclusion, we provide some recommendation as followed:

(i) Apply *prFrame* in different domains, with different risk models and risk factors specific to those domains. In principle, the methodology in *prFrame* is general enough to be used in multiple domains. To demonstrate and validate *prFrame*, we subjectively choose food as the domain of interest because of the limited dataset we already have in hand (See Chapter 7.2 where we conceptualise food based on the dataset and Chapter 6 where we experiment with *Belief Propagation* based on the dataset of salmonella in the broiler chicken). To establish with more confidence that *prFrame* can serve as a general framework, it may need to be evaluated and validated for several domains.

(ii) Propagation with more states, or continuous variables. In our use case, because the chosen bacteria (i.e., salmonella) is tested (or sampled) with only *absent* and *present*, the value of the bacteria in food does not matter. It is therefore proposed that the experiment be extended by increasing the number of states. This would greatly increase the complexity of the CPT, however, and the computational cost will also increase as the inputs and outputs of all combinations of states and variable nodes need to be calculated. In addition, performing *Belief Propagation* with a continuous distribution as opposed to the discrete distribution (with a predefined number of states as in this research) may produce a more relevant and contextual result for risk assessment.

(iii) Reduce the uncertainty by experimenting with different types of distributions of the risk factors. In general, *prFrame* attempts to reduce this uncertainty by gathering more evidence by observing the stages of a product (prov:Entity in the PG or variable node in the FG). In-depth research needs to be conducted, however, to reduce the uncertainty by observing the distribution of the risk factors (e.g., time, temperature, etc.) in a process (prov:Activity). Intuitively, we may think that if our distribution of risk factors becomes narrower or more precise, it should reduce the uncertainty in performing an inference. This investigation will have a significant impact in terms of the need to relocate sensors to collect the dataset.

(iv) A decision was taken to consider multiple aspects (e.g., structure of the graph, financial, available operators, etc.) *prFrame* is developed to provide a systematic approach to risk assessment and propagation. Furthermore, it is expected to enable decisions based on risk assessment. There is another aspect that needs to be taken into account before making a decision, however. That is a consequence. In this research, we do not provide an in-depth account of the notion of consequences with the risk assessment. In practice, before a decision is taken, it is important to associate the consequence with the outcome of the risk assessment.

(v) Sporadically and permanently investigations (e.g., sampling, etc.). Demonstrating due diligence with *prFrame* is based on the risk that each process conveys in the product supply chain. Hence, we intend to investigate the processes with high risk according to the result of the risk assessment. In this matter, further investigation may be beneficial to categorize the investigation process into where the best to do a sporadic or more permanent investigation the product supply chain.

Appendix A

Appendix A

A.1 A provenance of spaghetti with cooked sauce meat

```
document

default <http://provenance.ecs.soton.ac.uk/prov#>
prefix vargen <http://openprovenance.org/vargen#>
prefix tmpl <http://openprovenance.org/tmpl#>
prefix var <http://openprovenance.org/var#>
prefix prFood <https://provenance.ecs.soton.ac.uk/prFood#>

activity(cooking,-,-,[prov:type = "prFood:cooking" %% xsd:string])
activity(mixing_a,-,-,[prov:type = "prFood:mixing" %% xsd:string])
activity(mixing_b,-,-,[prov:type = "prFood:mixing" %% xsd:string])
activity(boiling,-,-,[prov:type = "prFood:boiling" %% xsd:string])

entity(raw_meat,[prov:type = "prFood:raw_meat" %% xsd:string, prFood:stageDetails
= "raw_meat" %% xsd:string])
entity(spaghetti,[prov:type = "prFood:spaghetti" %% xsd:string,
prFood:stageDetails = "spaghetti" %% xsd:string])
entity(cooked_meat,[prov:type = "prFood:cooked_meat" %% xsd:string,
prFood:stageDetails = "cooked_meat" %% xsd:string])
entity(sauce,[prov:type = "prFood:sauce" %% xsd:string, prFood:stageDetails =
"stageDetails" %% xsd:string])
entity(boiled_spaghetti,[prov:type = "prFood:boiled_spaghetti" %% xsd:string,
prFood:stageDetails = "boiled_spaghetti" %% xsd:string])
entity(cooked_sauce_meat,[prov:type = "prFood:cooked_sauce_meat" %% xsd:string,
prFood:stageDetails = "cooked_sauce_meat" %% xsd:string])
entity(served_spaghetti,[prov:type = "prFood:served_spaghetti" %% xsd:string,
prFood:stageDetails = "served_spaghetti" %% xsd:string])

wasGeneratedBy(cooked_meat,cooking,-)
wasGeneratedBy(boiled_spaghetti,boiling,-)
wasGeneratedBy(cooked_sauce_meat,mixing_a,-)
wasGeneratedBy(served_spaghetti,mixing_b,-)
```

```
used(cooking,raw_meat,-)
used(boiling,spaghetti,-)
used(mixing_a,sauce,-)
used(mixing_a,cooked_meat,-)
used(mixing_b,cooked_sauce_meat,-)
used(mixing_b,boiled_spaghetti,-)

wasDerivedFrom(cooked_meat, raw_meat)
wasDerivedFrom(boiled_spaghetti, spaghetti)
wasDerivedFrom(cooked_sauce_meat, sauce)
wasDerivedFrom(cooked_sauce_meat, cooked_meat)
wasDerivedFrom(served_spaghetti, cooked_sauce_meat)
wasDerivedFrom(served_spaghetti, boiled_spaghetti)

endDocument
```

Listing A.1: A provenance record in a spaghetti supply chain.

A.2 Defined terms of chicken supply chain in Figure 7.1

Attributes / Properties	**Description**
prov:type	An attribute to provide further typing information for any construct with an optional set of attribute-value pairs.
prFood:initialDist	An attribute to define initial distribution of bacteria in prov:Activity.
prFood:modelCode	An attribute to capture the coded risk model in prov:Activity that affects the distribution of bacteria.
prFood:modelRef	An attribute to capture the reference of where the risk model come from.
prFood:moduleOf	An attribute to capture the basic module(s) of MPRM.
prFood:processDetails	An attribute to describe the details of a process that is represented as prov:Activity.
prFood:retailDurDist	An attribute to capture the specific risk model in the retailing process regarding its duration.
prFood:retailTempDist	An attribute to capture the specific risk model in the retailing process regarding its temperature.
prFood:transportDurDist	An attribute to capture the specific risk model in the transporting process regarding its duration.
prFood:transportTempAmbientDist	An attribute to capture the specific risk model in the transporting process regarding its ambient temperature.
prFood:transportTempDist	An attribute to capture the specific risk model in the transporting process regarding its temperature.
prFood:homeDurDist	An attribute to capture the specific risk model in the storing process at home regarding its duration.
prFood:homeTempDist	An attribute to capture the specific risk model in the storing process at home regarding its temperature.
prFood:transferToBoardDist	An attribute to capture the specific risk model in preparing process at the kitchen regarding the transformation of bacteria to a chopping board.
prFood:transferToFoodDist	An attribute to capture the specific risk model in preparing process at the kitchen regarding the transformation of bacteria to food.
prFood:transferToHandDist	An attribute to capture the specific risk model in preparing process at the kitchen regarding the transformation of bacteria to the human hands.

prFood:cookDurDist	An attribute to capture the specific risk model in cooking process regarding its duration.
prFood:cookTempDist	An attribute to capture the specific risk model in cooking process regarding its temperature.
prFood:inAdequatelyProb	An attribute to capture the specific risk model in cooking process regarding the probability of inadequately cooking.
prFood:primProcessing	A property to describe that the type of prov:Activity is a primary processing of food.
prFood:Retailing	A property to describe that the type of prov:Activity is a retailing of food.
prFood:Transporting	A property to describe that the type of prov:Activity is a transporting of food.
prFood:Storing	A property to describe that the type of prov:Activity is a storing of food.
prFood:Preparing	A property to describe that the type of prov:Activity is a preparing of food.
prFood:Cooking	A property to describe that the type of prov:Activity is a cooking of food.
prFood:Initial	A property to describe the initial process of the product supply chain.
prFood:Growth	A property to describe the growth of bacteria as one of the MPRM module.
prFood:CrossContamination	A property to describe the cross contamination of bacteria as one of the MPRM module.
prFood:Inactivation	A property to describe the reduce number of bacteria as one of the MPRM module.
prFood:InitializedFood	A property to describe the initial stage in food in prov:Entity after food process (prov:Activity).
prFood:PrimaryProcsssedFood	A property to describe the stage in food after primary processing process.
prFood:RetailedFood	A property to describe the stage in food after retailing process.
prFood:TransportedFood	A property to describe the stage in food after transporting process.
prFood:StoredFood	A property to describe the stage in food after storing process.
prFood:PreparedFood	A property to describe the stage in food after preparing process.
prFood:CookedFood	A property to describe the stage in food after cookingprocess.
prFood:microbialLOGDIst	An attribute to capture the distribution of bacteria after a certain food process.
prFood:quantity	An attribute to capture the quantity of food after a certain food process.
prFood:stageDetails	An attribute to capture the description of the food stage.

TABLE A.1: The defined terms of the chicken supply chain in Figure 7.1.

A.3 Risk model and risk factor in experiment

Description	Variable	Unit	Distribution or Equation		
Prevalence	Prev			Min	Max
			Fixed value	0	1
Concentration	Conc	MPN/bird	Cumulative		

FIGURE A.1: A risk model and its associated risk factors after *primary* process.

Description	Variable	Unit	Distribution or Equation				
Transport temperature	T_pr	degree C		Min	Max		
			Uniform				
Transport time	t_pr	hours		Min	Max	CF	
			Correlated uniform			-0.75	
Minimum growth temperature	Tmin_pr	degree C	Constant	10			
Salt concentration	Slt_pr	%	Constant	1.9			
Log growth per hour	LGR_pr	log/hr	=EXP(-6.2251-(0.0114*Slt_pr) +(0.3234*T_pr) +(0.002*(Slt_pr*T_pr)) -(0.0085*(Slt_pr*Slt_pr)) -(0.0045*T_pr*T_pr)))				
Total log growth at retail	LG_pr	log	=IF(T_pr<Tmin_pr,0,t_pr*LGR_pr)				

FIGURE A.2: A risk model and its associated risk factors during the *transporting* process from a plant to a retail.

Description	Variable	Unit	Distribution or Equation				
Retail temperature	Rtl_Temp	degree C		Mean	SD	Min	Max
			Truncated Normal	4	2.8	-7.2	10
Retail time	Rtl_Time	days		Mean	Max	CF	
			Correlated Uniform	2	7	-0.75	
Minimum growth temperature	MGT	degree C	Constant	10			
Salt concentration	NaCl	%	Constant	1.9			
Log growth per hour	LogSGR_Rtl	log/hr	=EXP(-6.2251 -(0.0114*NaCl) +(0.3234*Rtl_Temp) +(0.002*(NaCl*Rtl_Temp)) -(0.0085*(NaCl*NaCl)) -(0.0045*(Rtl_Temp*Rtl_Temp)))				
Total Log growth at retail	Rtl_growth	log	=IF(Rtl_Temp<MGT.0.Rtl_Time*24*LogSGR_Rtl)				

FIGURE A.3: A risk model and its associated risk factors during *storing* process in a retail.

Description	Variable	Unit	Distribution or Equation
Ambient temperature during transport	Trans_Temp	degree C	Pert Min ML Max 0 13 24
Maximum change in temperature during transport	TransMax	degree C	= Trans_Temp -Rtl_Temp
Potential change in temperature during transport	Trans_DTemp1	degree C	Truncated Normal Mean SD Min Max 3.72 2.82 0 TransMax
Change in temperature during transport	Trans_Dtemp2	degree C	=IF(Trans_Temp -Rtl_Temp<=0,0,Trans_DTemp1
Chicken temperature after transport	Post_Trans_Temp	degree C	=Rtl_Temp +Trans_DTemp2
Average transport temperature	Avg_Trans_Temp	degree C	=Average(Rtl_Temp, Post_Trans_Temp)
Transport time	Trans_Time	Minutes	Correlated Cumulative Min Max CF 5 240 -0.75
Log growth per hour	LogSGR_Trans	log/hr	=EXP(-6.2251 -(0.0114*NaCl) +(0.3234*Avg_Trans_Temp) +(0.002*(NaCL*Avg_Trans_Temp (0.0085*(NaCL*NaCL)) -(0.0045*(Avg_Trans_Temp* Avg_Trans_Temp)))
Total log growth during transport	Trans_growth	log	=IF(Avg_Trans_Temp<MGT,0,Trans_Time/60*LogSGR_Trans)

FIGURE A.4: A risk model and its associated risk factors during the *transporting* process from a retail to a customer house.

Description	Variable	Unit	Distribution or Equation
Home storage temperature	Home_Temp	degree C	Mean SD Min Max Truncated Normal 4 2.65 -6.1 21.1
Home storage time	Home_Time	days	Min ML Max CF Correlated PERT 0 2 5 -0.75
Log growth per hour	LogSGR_Home	log/hr	=EXP(-6.2251 -(0.0114*NaCL) +(0.3234*Home_Temp) +(0.002*(NaCl*Home_Temp)) -(0.0085*(NaCl*NaCL)) -(0.0045*(Home_Temp*Home_Temp)))
Total log growth in home	Home_growth	log	+IF(Home_Temp<MGT,0,Home_Time*24*LogSGR_Home)
Total log growth in storage, transport and home	Growth	log	Rtl_growth + Trans-growth + Home_growth

FIGURE A.5: A risk model and its associated risk factors during *storing* process in a customer house.

Description	Variable	Unit	Distribution or equation			
Probability of inadequate cooking	Prob_AC	—		Min	ML	Max
			Pert	0.05	0.10	0.15
Adequately cooked?	AC	—	=binomial(1,1-Prod_AC)			
Proportion o0f cells in areas that permit a chance of survival	Prop_Prot			Min	ML	Max
			Pert	0.10	0.16	0.20
Log number of cells with chance of survival	Num_Prot	log cells	=IF(Conc=0,0,LOG10(10^Conc*Prop_Prot))			
Exposure time at exposure temperature for cells in “protected area”	Time_Prot	minutes		Min	ML	Max
			Pert	0.50	1.00	1.50
Exposure temperature during cooking in “protected areas”	Temp_Prot	degree C		Min	ML	Max
			Pert	60	64	65
D-value (at this temperature)	D_Prot	minutes	=10^(-0.139*Temp_Prot +8.58)			
log reduction in “protected area”	Prto_LR	log	=IF(AC=1,”death”,Time_Prot / D_Prot)			

FIGURE A.6: A risk model and its associated risk factors during the *cooking* process in a kitchen.

Description	Variable	Unit	Distribution or equation			
Number of organisms on bird	Num	cells	=IF(Conc=0,0,10^Conc)			
Chickens ⇒ Hands						
Transfer from chicken to hands?	XCH	—	=IF(Num=0,0,1)			
Proportion transferred from chicken	Pop_CH	proportion	Pert	Min 0	ML 0.1	Max 0.15
Number on hands	Num_H	cell	=IF(XCH=0,0,Num*Prop_CH)			
Number left on chicken	Num_C1	cell	=Num -Num_H			
Hands ⇒ Other food						
Probability that hands are not washed	HW_Prob	—	Beta	alpha 64	beta 46	
Hands not washed?	HW	—	=binomial(1,HW_Prob)			
Proportion transferred from hands	Prop_HF	—	Pert	Min 0.00	ML 0.10	Max 0.15
Number on other foods via hands	Num_OF1	—	=IF(HW=0,0,Num_H*Prop_HF)			
Chickens ⇒ Board						
Transfer from chicken to board	XCB	—	=IF(Num=0,0,1)			
Proportion transferred from chicken to board	Prop_CB	proportion	Pert	Min 0	ML 0.1	Max 0.15
Number on board	Num_B	cell	=IF(XCB=0,0,Num*Prop_CB)			
Number left on chicken		cell	=NUm_C1 -Num_B			
Board ⇒ Other food						
Probability that board is used for other foods	Brd_use_Prob	—	Beta	alpha 66	beta 44	
Boards used for other food?	Brd_use	—	=binomial(1,Brd_use_Prob)			
Proportion transferred from board	Prop_BF	—	Pert	Min 0.00	ML 0.10	Max 0.15
Number on other foods from chicken via board	Num_OF2	—	=IF(Brd_use=0,0,Num_B*Prop_BF)			
Number ingested via-cross-contamination	Num_XC	cell	=Num_OF1 +Num_OF2			
Ingestion via cross-contamination?	—	—	+IF(Num_XC=0,0,1)			

FIGURE A.7: A risk model and its associated risk factors during the *preparing* process in a kitchen.

A.4 Type of distributions

A.4.1 Uniform Distribution

The uniform distribution is also called rectangular distribution and defined by two parameters, min (minimum) and max (maximum) values.

A.4.2 Normal Distribution

The normal distribution is a very common continuous probability distribution in nature. It shape represents bell-curve and is defined with μ (mean) and σ (standard deviation).

A.4.3 Truncated Normal Distribution

The truncated normal distribution is defined is similar with the normal distribution with a range to limit the distribution to an upper, lower or double truncated distribution. This distribution has four parameters, namely μ (mean), σ (standard deviation), low (lower), and up (upper) values.

A.4.4 Triangular Distribution

The triangular distribution is a continues probability distribution with three parameters, min (minimum), max (maximum), and mode (most common) values.

A.4.5 Pert Distribution

The pert distribution is similar with the triangular distribution but smoother. It is defined with at least three parameters, max (maximum), mode (most common), and λ (shape of parameter, optional) values.

Bibliography

[1] Food Standards Agency, "Food Law Code of Practice (England)-April 2015," Food Standards Agency, Report, April 2015. [Online]. Available: https://www.food.gov.uk/sites/default/files/Food%20Law%20Code%20of%20Practice%20-%202015_1.pdf

[2] A. Eves and P. Dervisi, "Experiences of the implementation and operation of hazard analysis critical control points in the food service sector," *International Journal of Hospitality Management*, vol. 24, no. 1, p. 3–19, March 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0278431904000350

[3] International Organization for Standarization, "ISO 9000 - Quality management," http://www.iso.org/iso/home/standards/management-standards/iso_9000.htm, accessed: 2016-05-17.

[4] C. Harland, R. Brenchley, and H. Walker, "Risk in supply networks," *Journal of Purchasing and Supply Management*, vol. 9, no. 2, pp. 51–62, March 2003, supply Chain Management: Selected Papers from the European Operat ions Management Association (EurOMA) 8th International Annual Conference. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1478409203000049

[5] T. Aven, J. Vinnem, and H. Wiencke, "A decision framework for risk management, with application to the offshore oil and gas industry," *Reliability Engineering & System Safety*, vol. 92, no. 4, pp. 433–448, April 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0951832006000093

[6] S. Kaplan and B. J. Garrick, "On the quantitative definition of risk," *Risk Analysis*, vol. 1, no. 1, pp. 11–27, 1981. [Online]. Available: http://dx.doi.org/10.1111/j.1539-6924.1981.tb01350.x

[7] R. Ojha, A. Ghadge, M. K. Tiwari, and U. S. Bititci, "Bayesian network modelling for supply chain risk propagation," *International Journal of Production Research*, vol. 56, no. 17, pp. 5795–5819, 2018. [Online]. Available: https://doi.org/10.1080/00207543.2018.1467059

[8] L. Moreau, P. Groth, J. Cheney, T. Lebo, and S. Miles, "The rationale of prov," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 35, no. 4, pp. 235–257, December 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570826815000177

[9] D. Bhagwat, L. Chiticariu, W.-C. Tan, and G. Vijayvargiya, "An annotation management system for relational databases," *The VLDB Journal*, vol. 14, no. 4, pp. 373–396, 2005.

[10] P. Agrawal, O. Benjelloun, A. D. Sarma, C. Hayworth, S. Nabar, T. Sugihara, and J. Widom, "Trio: A system for data, uncertainty, and lineage," *Proc. of VLDB 2006 (demonstration description)*, 2006.

[11] C. Sar and P. Cao, "Lineage file system," *Online at http://crypto.stanford.edu/cao/lineage.html*, pp. 411–414, 2005.

[12] B. J. Frey, F. R. Kschischang, H.-A. Loeliger, and N. Wiberg, "Factor graphs and algorithms."

[13] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE transactions on Information Theory*, vol. 46, no. 2, pp. 325–343, 2000.

[14] B. M. Beamon, "Supply chain design and analysis:: Models and methods," *International Journal of Production Economics*, vol. 55, no. 3, pp. 281 – 294, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925527398000796

[15] J. T. Mentzer, W. DeWitt, J. S. Keebler, S. Min, N. W. Nix, C. D. Smith, and Z. G. Zacharia, "Defining supply chain management," *Journal of Business logistics*, vol. 22, no. 2, pp. 1–25, 2001.

[16] E. Holleran, M. E. Bredahl, and L. Zaibet, "Private incentives for adopting food safety and quality assurance," *Food Policy*, vol. 24, pp. 669–683, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306919299000718

[17] European Parliament and the Council, "Regulation (EC) No 178/2002 of The European Parliament and of The Council," European Parliament and the Council, Report, January 2002. [Online]. Available: http://www.food.gov.uk/sites/default/files/multimedia/pdfs/1782002ecregulation.pdf

[18] A. Regattieri, M. Gamberi, and R. Manzini, "Traceability of food products:general framework and experimental evidence," *Journal of Food Engineering*, vol. 81, p. 347–356, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0260877406006893

[19] R. Loader and J. E. Hobbs, "Strategic responses to food safety legislation," *Food Policy*, vol. 24, no. 6, p. 685–706, December 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306919299000731

[20] M. D. Pierson, *HACCP: principles and applications*. Springer Science & Busines, 1992.

[21] L. Moreau, P. Groth, S. Miles, J. Vazquez-Salceda, J. Ibbotson, S. Jiang, S. Munroe, O. Rana, A. Schreiber, V. Tan *et al.*, "The provenance of electronic data," *Communications of the ACM*, vol. 51, no. 4, p. 52, 2008.

[22] P. Groth and L. Moreau, "Recording process documentation for provenance," *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, no. 9, pp. 1246–1259, 2008.

[23] M. Ali and L. Moreau, *A Provenance-Based Policy Control Framework for Cloud Services*. Cham: Springer International Publishing, 2015, pp. 127–138. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-16462-5_10

[24] M. D. Allen, A. Chapman, L. Seligman, and B. Blaustein, "Provenance for collaboration: Detecting suspicious behaviors and assessing trust in information," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*, Oct 2011, pp. 342–351.

[25] C. Goble, "Position Statement: Musings on Provenance, Workflow and (Semantic Web) Annotations for Bioinformatics," in *Workshop on Data Derivation and Provenance, Chicago*, vol. 3, 2002, accessed: 2016-06-27.

[26] J. Golbeck and J. Hendler, "A semantic web approach to the provenance challenge," *Concurrency and Computation : Practice and Experience*, vol. 20, no. 5, pp. 432–439, April 2008. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/cpe.1238/full

[27] W. D. Pence, L. Chiappetti, C. G. Page, R. A. Shaw, and E. Stobie, "Definition of the flexible image transport system (fits), version 3.0," *Astronomy & Astrophysics*, vol. 524, p. A42, 2010.

[28] R. G. Fegeas, J. L. Cascio, and R. A. Lazar, "An overview of fips 173, the spatial data transfer standard," *Cartography and Geographic Information Systems*, vol. 19, no. 5, pp. 278–293, 1992.

[29] L. Moreau and P. Missier, "PROV-DM: The PROV Data Model,," World Wide Web Consortium, W3C Recommendation REC-prov-dm-20130430, April 2013. [Online]. Available: https://www.w3.org/TR/prov-dm/

[30] L. Moreau and P. Groth, *Provenance: An Introduction to PROV*, ser. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool Publishers, 2013. [Online]. Available: https://books.google.co.uk/books?id=8aBeAQAAQBAJ

[31] T. D. Nies, "Constraints of the PROV Data Model,," World Wide Web Consortium, W3C Recommendation REC-prov-constraints-20130430, April 2013. [Online]. Available: https://www.w3.org/TR/prov-constraints/

[32] Y. Gil and S. Miles, "PROV Model Primer,," World Wide Web Consortium, W3C Note NOTE-prov-primer-20130430, April 2013. [Online]. Available: https://www.w3.org/TR/prov-primer/

[33] M. Markovic, P. Edwards, and D. Corsar, "Sc-prov: A provenance vocabulary for social computation," in *International Provenance and Annotation Workshop*. Springer, 2014, pp. 285–287.

[34] D. Garijo and Y. Gil, "Augmenting prov with plans in p-plan: scientific processes as linked data." CEUR Workshop Proceedings, 2012.

[35] H. S. Packer, L. Drăgan, and L. Moreau, *An Auditable Reputation Service for Collective Adaptive Systems*. Cham: Springer International Publishing, August 2014, no. 4, pp. 159–184. [Online]. Available: url="http://dx.doi.org/10.1007/978-3-319-08681-1_8"

[36] T. D. Huynh, M. Ebden, M. Venanzi, S. D. Ramchurn, S. Roberts, and L. Moreau, "Interpretation of crowdsourced activities using provenance network analysis," in *First AAAI Conference on Human Computation and Crowdsourcing*. AAAI Press, 2013. [Online]. Available: http://www.aaai.org/ocs/index.php/HCOMP/HCOMP13/paper/view/7388

[37] S. D. Ramchurn, T. D. Huynh, M. Venanzi, and B. Shi, "Collabmap: Crowdsourcing maps for emergency planning," in *Proceedings of the 5th Annual ACM Web Science Conference*, ser. WebSci '13. New York, NY, USA: ACM, 2013, pp. 326–335. [Online]. Available: http://dl.acm.org/citation.cfm?id=2464508

[38] S. D. Ramchurn, T. D. Huynh, Y. Ikuno, J. Flann, F. Wu, L. Moreau, N. R. Jennings, J. E. Fischer, W. Jiang, T. Rodden, E. Simpson, S. Reece, and S. J. Roberts, "Hac-er a disaster response system based on human-agent collectives," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, ser. AAMAS '15. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 533–541. [Online]. Available: http://dl.acm.org/citation.cfm?id=2772947

[39] L. Philip, A. Chorley, J. Farrington, and P. Edwards, "Data provenance, evidence-based policy assessment, and e-social science," in *Third international conference on e-social science*. Citeseer, 2007.

[40] M. Markovic, P. Edwards, M. Kollingbaum, and A. Rowe, *Modelling Provenance of Sensor Data for Food Safety Compliance Checking*. Cham:

Springer International Publishing, 2016, pp. 134–145. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-40593-3_11

[41] B. V. Batlajery, M. Weal, A. Chapman, and L. Moreau, "prfood: Ontology principles for provenance and risk in the food domain," in *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*. IEEE, 2018, pp. 17–24.

[42] ——, "Belief propagation through provenance graphs," in *International Provenance and Annotation Workshop*. Springer, 2018, pp. 145–157.

[43] D. Corsar, M. Markovic, and P. Edwards, "Capturing the provenance of internet of things deployments," in *International Provenance and Annotation Workshop*. Springer, 2018, pp. 196–199.

[44] C. Burnett, L. Chen, P. Edwards, and T. J. Norman, "Traac: trust and risk aware access control," in *2014 Twelfth Annual International Conference on Privacy, Security and Trust*. IEEE, 2014, pp. 371–378.

[45] M. Markovic, P. Edwards, and D. Corsar, "A role for provenance in social computation," in *Proceedings of the First International Workshop on Crowdsourcing the Semantic Web-CrowdSem 2013*. CEUR-WS, 2013.

[46] C. Baillie, P. Edwards, and E. Pignotti, "Qual: A provenance-aware quality model," *Journal of Data and Information Quality (JDIQ)*, vol. 5, no. 3, p. 12, 2015.

[47] ——, "Quality assessment, provenance, and the web of linked sensor data," in *International Provenance and Annotation Workshop*. Springer, 2012, pp. 220–222.

[48] D. Corsar and P. Edwards, "Enhancing open data with provenance," *Digital Futures*, 2012.

[49] D. Corsar, P. Edwards, N. Velaga, J. Nelson, and J. Z. Pan, "Exploring provenance in a linked data ecosystem," in *International Provenance and Annotation Workshop*. Springer, 2012, pp. 226–228.

[50] S. Amland, "Risk based testing and metrics: Risk analysis fundamentals and metrics for software testing including a financial application case study," *Journal of Systems and Software*, vol. 53, no. 3, pp. 287–295, September 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0164121200000194

[51] M. N. Faisal, D. Banwet, and R. Shankar, "Information risks management in supply chains: an assessment and mitigation framework," *Journal of Enterprise Information Management*, vol. 20, no. 6, pp. 677–699, October 2007. [Online]. Available: http://www.emeraldinsight.com/doi/abs/10.1108/17410390710830727

[52] Y. Chen, R. L. Probert, and D. P. Sims, "Specification-based regression test selection with risk analysis," in *Proceedings of the 2002 conference of the Centre for Advanced Studies on Collaborative research*, 2002, p. 1.

[53] A. M. Lammerding and A. Fazil, "Hazard identification and exposure assessment for microbial food safety risk assessment," *International Journal of Food Microbiology*, vol. 58, no. 3, p. 147–157, July 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168160500002695

[54] Joint FAO and World Health Organization and others, "Application of risk analysis to food standards issues: report of the Joint FA," Food and Agriculture Organization and World Health Organization, Report, 1995, accessed: 2016-05-05. [Online]. Available: http://www.fao.org/docrep/008/ae922e/ae922e00.HTM

[55] S. B. Dennis, J. Kause, M. Losikoff, D. L. Engeljohn, and R. L. Buchanan, "Using risk analysis for microbial food safety regulatory decision making," in *Microbial risk analysis of foods*. American Society of Microbiology, 2008, pp. 137–175.

[56] M. Alhomidi and M. Reed, "Risk assessment and analysis through population-based attack graph modelling," in *Internet Security (WorldCIS), 2013 World Congress on*, December 2013, pp. 19–24. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6751011&tag=1

[57] S. Henson and J. Caswell, "Food safety regulation: an overview of contemporary issues," *Food Policy*, vol. 24, no. 6, pp. 589–603, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S030691929900072X

[58] A. Lammerding, "Using microbiological risk assessment (mra) in food safety management," ILSI Europe, Tech. Rep., 2007.

[59] A. M. Fazil, *A primer on risk assessment modelling: focus on seafood products*. Food & Agriculture Org., 2005, no. 462.

[60] J. Bassett, M. Nauta, R. Lindqvist, and M. Zwietering, "Tools for microbiological risk assessment," ILSI Europe, Tech. Rep., 2012.

[61] G. E. Apostolakis, "How useful is quantitative risk assessment?" *Risk Analysis: An International Journal*, vol. 24, no. 3, pp. 515–520, 2004.

[62] S. Pudar, G. Manimaran, and C.-C. Liu, "Penet: A practical method and tool for integrated modeling of security attacks and countermeasures," *Computers & Security*, vol. 28, no. 8, pp. 754 – 771, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404809000522

[63] T. J. Linsmeier and N. D. Pearson, "Value at risk," *Financial Analysts Journal*, pp. 47–67, 2000.

[64] M. Raugas, J. Ulrich, R. Faux, S. Finkelstein, and C. Cabot, "Cyberv@ r," 2013.

[65] W. H. Organization, *Risk assessments of Salmonella in eggs and broiler chickens*. Food & Agriculture Org., 2002, vol. 2.

[66] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*, ser. Adaptive computation and machine learning. MIT Press, 2009. [Online]. Available: https://books.google.co.uk/books?id=7dzpHCHzNQ4C

[67] A. Ankan and A. Panda, *Mastering Probabilistic Graphical Models Using Python*. Packt Publishing Ltd, 2015.

[68] M. Van Steen, "Graph theory and complex networks," *An introduction*, vol. 144, 2010.

[69] R. J. Wilson, *Introduction to graph theory*. Pearson Education India, 1979.

[70] D. Koller, N. Friedman, L. Getoor, and B. Taskar, *Introduction to statistical relational learning*. MIT Press, 2007.

[71] M. Newman, *Networks: an introduction*. Oxford university press, 2010.

[72] S. L. Lauritzen, *Graphical models*. Clarendon Press, 1996, vol. 17.

[73] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier, 2014.

[74] M. H. Cohen, "The unknown and the unknowable-managing sustained uncertainty," *Western Journal of Nursing Research*, vol. 15, no. 1, pp. 77–96, 1993.

[75] J. Pearl, *Causality: Models, Reasoning and Inference*, 2nd ed. New York, NY, USA: Cambridge University Press, 2009.

[76] P. Vieira, A. Costa, and J. Macedo, "A comparison of opportunistic connection datasets," *International Journal of Distributed Systems and Technologies (IJDST)*, vol. 4, no. 3, pp. 31–46, 2013.

[77] B. M. Jedynak and S. Khudanpur, "Maximum likelihood set for estimating a probability mass function," *Neural computation*, vol. 17, no. 7, pp. 1508–1530, 2005.

[78] T. P. Oscar, "A quantitative risk assessment model for salmonella and whole chickens," *International Journal of Food Microbiology*, vol. 93, no. 2, pp. 231–247, June 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168160503006081

[79] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[80] P. Garbolino and F. Taroni, "Evaluation of scientific evidence using bayesian networks," *Forensic Science International*, vol. 125, no. 2, pp. 149 – 155, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0379073801006429

[81] J. Pearl, *Reverend Bayes on inference engines: A distributed hierarchical approach.*

[82] A. Chapman, B. Blaustein, and C. Elsaesser, "Provenance-based belief," in *3rd USENIX Workshop on the Theory and Practice of Provenance*, 2010.

[83] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498–519, Feb 2001.

[84] World Health Organization and Agriculture Organization of the United Nations, "Exposure assessment of microbiological hazards in food," ATM Forum Contribution 94-0735R1, 2008. [Online]. Available: http://www.who.int/foodsafety/publications/micro/MRA7.pdf

[85] INTERNATIONAL LIFE SCIENCE INSTITUTE, "Impact of Microbial Distributions on Food Safety," http://ilsi.eu/wp-content/uploads/sites/3/2016/06/Microbial-Distribution-2010.pdf, accessed: 2017-06-22.

[86] A. S. R. Duarte, "The interpretation of quantitative microbial data: meeting the demands of quantitative microbiological risk assessment," 2013.

[87] O. I. Aruoma, "The impact of food regulation on the food supply chain," *Toxicology*, vol. 221, no. 1, pp. 119–127, April 2006, nutraceuticals and Functional Foods Regulations in the United States and Around The World. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0300483X0600045X

[88] Ministry of Higher Education, Training and Employment Creation, Namibia and The Food and Agriculture Organization of the United Nations, *A Handbook for Namibian Volunteer Leaders*. Ministry of Higher Education, Training and Employment Creation, Namibia and The Food and Agriculture Organization of the United Nations, 2004, accessed: 2016-05-05. [Online]. Available: ftp://ftp.fao.org/docrep/fao/008/a0104e/a0104e.pdf

[89] M. Nauta, "A modular process risk model structure for quantitative microbiological risk assessment and its application in an exposure assessment of bacillus cereus in a repfed," accessed: 2016-05-05. [Online]. Available: http://hdl.handle.net/10029/261555

[90] M. Thakur and C. R. Hurburgh, "Framework for implementing traceability system in the bulk grain supply chain," *Journal of Food Engineering*,

vol. 95, no. 4, p. 617–626, December 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0260877409003264

[91] J. E. Ehiri and G. P. Morris, "Food safety control strategies: A critical review of traditional approaches," *International Journal of Environmental Health Research*, vol. 4, no. 4, pp. 254–263, 1994. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/09603129409356824

[92] M. M. Aung and Y. S. Chang, "Traceability in a food supply chain: Safety and quality perspectives," *Food Control*, vol. 39, no. Supplement C, pp. 172 – 184, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0956713513005811

[93] G. Senneset, E. Forås, and K. M. Fremme, "Challenges regarding implementation of electronic chain traceability," *British Food Journal*, vol. 109, no. 10, pp. 805–818, 2007.

[94] European Commission: Health & Consumer Protection Directorate-Generate, "Preliminary Report: Risk assessment of food borne bacterial pathogens: Quantitative methodology relevant for human exposure assessment," Food Standards Agency, Preliminary Report, February 2002. [Online]. Available: http://www.food.gov.uk/sites/default/files/multimedia/pdfs/fsa1782002guidance.pdf

[95] W. B. McNab, "A general framework illustrating an approach to quantitative microbial food safety risk assessment," *Journal of food protection*, vol. 61, no. 9, pp. 1216–1228, 1998.

[96] H. M. Marks, M. E. Coleman, C.-T. J. Lin, and T. Roberts, "Topics in microbial risk assessment: dynamic flow tree process," *Risk Analysis*, vol. 18, no. 3, pp. 309–328, 1998.

[97] M. H. Cassin, A. M. Lammerding, E. C. Todd, W. Ross, and R. S. McColl, "Quantitative risk assessment for escherichia coli o157: H7 in ground beef hamburgers," *International journal of food microbiology*, vol. 41, no. 1, pp. 21–44, 1998.

[98] J. Dawber, B. Horn, and J. Brown, "Performing a quantitative microbial risk analysis using second-order monte carlo simulation," 2009.

[99] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: principles and methods," *Data & knowledge engineering*, vol. 25, no. 1-2, pp. 161–197, 1998.

[100] T. Pizzuti, G. Mirabelli, M. A. Sanz-Bobi, and F. Goméz-Gonzaléz, "Food track & trace ontology for helping the food traceability control," *Journal of Food Engineering*, vol. 120, pp. 17–30, 2014.

[101] M. Xiangwei and Z. Lin, "An ontology development for haccp knowledge description and sharing in food cold chain," 2015.

[102] C. Byrd-Bredbenner, J. Berning, J. Martin-Biggers, and V. Quick, "Food safety in home kitchens: a synthesis of the literature," *International journal of environmental research and public health*, vol. 10, no. 9, pp. 4060–4085, 2013.

[103] C. Griffith, D. Worsfold, and R. Mitchell, "Food preparation, risk communication and the consumer," *Food control*, vol. 9, no. 4, pp. 225–232, 1998.

[104] L. J. Kagan, A. E. Aiello, and E. Larson, "The role of the home environment in the transmission of infectious diseases," *Journal of community health*, vol. 27, no. 4, pp. 247–267, 2002.