TALLUR, GAYATRI, M.S. Uncertain Data Integration with Probabilities. (2013)
Directed by Dr. Fereidoon Sadri. 48 pp.

Real world applications that deal with information extraction, such as business intelligence software or sensor data management, must often process data provided with varying degrees of uncertainty. Uncertainty can result from multiple or inconsistent sources, as well as approximate schema mappings. Modeling, managing and integrating uncertain data from multiple sources has been an active area of research in recent years [6][7][1][2]. In particular, data integration systems free the user from the tedious tasks of finding relevant data sources, interacting with each source in isolation using its corresponding interface and combining data from multiple sources by providing a uniform query interface to gain access to the integrated information [5].

Previous work has integrated uncertain data using representation models such as the *possible worlds* and *probabilistic relations* [12][1][2]. We extend this work by determining the probabilities of possible worlds of an *extended probabilistic relation*. We also present an algorithm to determine when a given extended probabilistic relation can be obtained by the integration of two probabilistic relations and give the decomposed pairs of probabilistic relations.

UNCERTAIN DATA INTEGRATION

WITH PROBABILITIES

by

Gayatri Tallur

A Thesis Submitted to
the Faculty of The Graduate School at
The University of North Carolina at Greensboro
in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Greensboro
2013

Approved by

Dr. Fereidoon Sadri
Committee Chair

APPROVAL PAGE

This thesis written by GAYATRI TALLUR has been approved by the following committee of the Faculty of The Graduate School at The University of North Carolina at Greensboro.

Committee Chair Dr. Fereidoon Sadri

Committee Members Dr. Nancy Green

Dr. Jing Deng

November 7, 2013
Date of Acceptance by Committee

November 7, 2013
Date of Final Oral Examination

ACKNOWLEDGEMENTS

I sincerely thank my advisor, Dr. Fereidoon Sadri, for his abundant guidance and support throughout the course of this research without which this thesis would not have been possible. I am very thankful to him for giving me the opportunity to work with him and believing in me. I thoroughly enjoyed working under him. I would also like to thank Dr. Jing Deng and Dr. Nancy Green for their valuable guidance and feedback.

I am indebted to my husband for his constant encouragement, support and love. I am very grateful to my mother and my family members for their unconditional support and care. I want to thank Nina Revankar and her family for their ample love and concern during my study. Nina’s spontaneous gestures of help on those busy days really made a big difference, and I am deeply indebted to her for it.

I would like to express my gratitude to my friends at School for cheering me up and encouraging me all through. I am highly grateful to all my dear friends in Greensboro for keeping my life outside School fun at all times, and for their endless concern and support throughout.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Page

CHAPTER I

INTRODUCTION

Real world applications that deal with information extraction, such as sensor data management, Optical Character Recognition (OCR), data mining in social networks, deduplication and data cleaning or even business intelligence software, must often process data provided with varying degrees of certainty. Useful information is usually obtained from the available data by tracing the relevant data sources, interacting with each source in isolation using its corresponding interface and combining data from all the sources. With the growing number of such applications today, it is important that the user be able to pose complex queries and retrieve information in a very efficient and scalable manner. Given the imprecise nature of information in the real world, processing and integrating uncertain data continues to be a challenging area of research.

1.1 Sources of Uncertainty in Data

In the context of information retrieval systems, uncertainty could manifest itself for many reasons including it being the outcome of *flawed data*, or *missing knowledge*. While flawed data can result from recording errors during the process of data collection or entry, missing knowledge can result from the inability to fill gaps in the collected data. Both these issues can be addressed by enumerating all possibilities for the corrupt or missing data and assigning a degree of likelihood to them. Thus, information retrieval systems that operate on uncertain data glean useful information from all available and enumerated data to provide results that are *most likely* to be true. In the ensuing lines, we categorize the sources of uncertainty and provide examples for each of them.

1.1.1 Measurement Errors

Consider the example of a sensor management system that must process the readings from two sensors reporting the temperature for the same city. If one recorded a value of 70F and the other recorded 72F, then the sensor data can be classified as *Uncertain* simply because they are not matching precisely.

1.1.2 Multiple or Inconsistent Sources

Consider the example of two students S1 and S2 who provide information regarding the courses that another student Bob has registered for in the current semester. Student S1 claims that Bob has enrolled for CS100 or CS101 while student S2 claims that Bob is enrolled for CS101 or CS102. Clearly, the courses that Bob is registered for can be classified as *Uncertain*. Along similar lines, sources that provide data that is deemed inconsistent also lead to uncertainty.

1.1.3 Approximate Schema Mapping

Consider the example of two schemas, *Student (Name, SSN, Marks)* and *Grad-Student (Name, ID, Grades)*. If every *Grade* in *Grad-Student.Grades* maps uniquely to every *Marks* in *Student.Marks*, then this one-to-one mapping ensures definite results. However, if each *Grade* in *Grad-Student.Grades* maps to a range of *Marks* in *Grad-Student.Marks*, then this approximate mapping introduces *uncertainty*.

For every source of uncertainty discussed above, uncertain data cannot be processed within the confines of a traditional database information retrieval system as easily. Not only is it more complicated to process, but it is also less efficient [11]. Processing such data begins with modeling uncertain data differently. In fact, research in recent times has focused on addressing the modeling techniques and managing such data [1][6][7].

1.2 Modeling Uncertain Data

Traditional databases do not allow scope for handling information retrieval errors resulting from flawed data or missing knowledge. A strong need is felt for a database that can model uncertain data and allow multiple values based on user-defined confidence levels. The ensuing sections discuss the different modeling techniques that such *uncertain databases* could use to represent uncertain data.

1.2.1 Possible Worlds Model

The possible worlds model has been widely accepted as a conceptual model of uncertain information. In this model, the information represented by each source is distributed over many traditional database instances, each instance being a possible state of the real world [1]. Each possible world is simply a traditional database containing data without any uncertainty.

Table 1. Possible Worlds Model for Representing Uncertain Data

Possible World $\{D_1\}$

Tuple	Location	Temperature
t_1	Greensboro	70F

Possible World $\{D_2\}$

Tuple	Location	Temperature
r_1	Greensboro	72F

Invalid Possible World $\{D_3\}$

Tuple	Location	Temperature
t_1	Greensboro	70F
r_1	Greensboro	72F

Invalid Possible World $\{D_4\}$

Φ

Table 1 shows the four database instances for the example presented in Section 1.1.1. From Table 1 however, Possible World $\{D_3\}$ is invalid because it is improbable that a single location has different temperature readings simultaneously. Furthermore, Possible World $\{D_4\}$ is invalid because the place must have a temperature reading associated with it. Thus, the relevant possible worlds are simply *{D_1}* and *{D_2}*.

As the amount of uncertain data increases, the number of possible worlds also increases exponentially. The resulting representation for uncertain data becomes unwieldy and processing time becomes unacceptably large. Therefore, we resort to the *probabilistic relation* model which serves as a more compact representation, by helping avoid enumerating all the possible worlds.

1.2.2 Probabilistic Relation Model (pr-relation)

The uncertain database that chooses to model its data using the *possible worlds* model uses multiple schemas to represent all the possible enumerations of uncertain data. In contrast, the *probabilistic relation* model [3] uses only one schema with one additional attribute known as the *Event* attribute (E). Unique *atomic* events across all possible worlds are expressed using Boolean variables, or *event* variables, and are combined using logic expressions to create *complex* events that are assigned to the attribute E for all tuples in the schema. In other words, the attribute E for every tuple in the schema is simply a Boolean *True* or *False* value. For the example presented in Section 1.1.1, the atomic events "*The temperature in Greensboro is 70F*" and "*The temperature in Greensboro is 72F*" are represented by the event variable **x** and **y** respectively. The atomic event **y** may be interpreted as **¬x** for purposes of simplification. The truth value for **x** represents the corresponding possible worlds.

Table 2 shows the probabilistic relation for the example discussed above. The individual rows or *tuples* are simply represented by the variables $\mathbf{t_1}$ and $\mathbf{r_1}$ respectively. Thus, when **x** is True, only $\mathbf{t_1}$ is True representing Possible World {D_1} and when **x** is False, only $\mathbf{r_1}$ is True representing Possible World {D_2}.

The probabilistic relations model is equivalent to the possible worlds model, and the work of [13] formally shows that in fact the probabilistic relation model can represent any possible worlds set.

Table 2. Probabilistic Relation Model for Representing Uncertain Data

Tuple	*Location*	*Temperature*	*Event Attribute (E)*
$\mathbf{t_1}$	Greensboro	70F	x
$\mathbf{r_1}$	Greensboro	72F	¬x

An uncertain database can utilize the modeling techniques described in Section 1.2.1 and 1.2.2, and assign probabilities to every possible world representing the degree of belonging to the database. The sum of these probabilities over all possible worlds in the database should be equal to 1. Such a database is then known as a *Probabilistic Database*. A Probabilistic Database for the example shown in Table 1, for all possible worlds with equal probabilities, might be depicted as $P(D_1) = 0.5$ and $P(D_2) = 0.5$.

1.3 Managing Uncertain Data

Managing uncertain data refers to the set of operations that are used to store data, modify it and extract useful information. Operations such as indexing, join processing and query evaluation must be redesigned to handle uncertain data properly. Each of these is an active area of research and query processing for probabilistic databases is still, in fact, in its infancy. Little is known about which queries can be evaluated in polynomial time, and the few existing evaluation methods employ expensive main-memory algorithms [10].

1.4 Data Integration

Often, applications need to retrieve consolidated information using data stored in *multiple* uncertain databases that have the same schema. The process of consolidating all the available data across various databases is known as *Data Integration*. Data integration is important because integrating multiple sources of uncertain data can help resolve some uncertainty, yielding more accurate results than any of the individual sources [4]. The importance of information integration for uncertain data has been realized in recent years. In fact, [1] makes the following relevant observation:

> While in traditional database management managing uncertainty and lineage seems like a nice feature, in data integration it becomes a necessity.

The result of integration is useful only to the extent that the information it produces can be trusted. Hence, providing a confidence value to the integrated information is a necessity in many applications [1]. This work concentrates on integrating two probabilistic databases and determining the probabilities of the possible worlds in the integration. The challenge here lies in accurately determining the probabilities.

1.5 Contributions from This Work

The work of [1] has developed methods to integrate uncertain databases with and without known associated probabilities using the possible worlds model. On the other hand, the work of [2] has developed a method to integrate uncertain databases using the probabilistic relations model without considering the associated probabilities. Our work extends the work of [2] towards uncertain data integration in the following manner.

- We present two methods to determine the probabilities of the possible worlds in the integration of two uncertain databases associated with a known set of probabilities.

- We show that both these methods are equivalent.
- We give sufficient conditions that an extended probabilistic relation can be obtained by the integration of two probabilistic relations.
- We present the decomposition algorithm that determines if a given extended probabilistic relation can be obtained by the integration of two probabilistic relations and gives the decomposed pairs of probabilistic relations.
- Given the result of integration whose decomposition leads to multiple pairs of probabilistic relations, we show that all pairs are equivalent.

This work is organized as follows. Chapter II discusses all relevant previous work. Chapters III and IV present our work and results. Finally, Chapter V presents the conclusions and the scope for future research.

CHAPTER II

BACKGROUND INFORMATION

This chapter summarizes the work done so far towards integrating data from uncertain databases. Towards this end, we discuss the use of the Possible Worlds model to integrate information from two uncertain databases across two different scenarios, as presented in the work of [1] – firstly, when the information is not associated with known probabilities and secondly, when it is. We also highlight the use of the Probabilistic Relation model to integrate information from uncertain databases when the information is not associated with probabilities, as presented in the work of [2]. In this work, we extend these ideas towards using the Probabilistic Relation model to integrate information from uncertain databases associated with known probabilities to determine the probabilities of the result of integration.

2.1 Data Integration using the Possible Worlds Model without probabilities

The work of [12] uses the well-known possible worlds model to represent and integrate uncertain information from two uncertain sources using superset-containment. The work of [1] uses the same model to represent and integrate uncertain information. It introduces a simple logic based approach for doing the integration. Since [1] forms the basis for this research, we summarize the procedure below.

Given an uncertain source U with T(U) representing the finite set of tuples of U, a propositional variable t_i is assigned to each tuple in T(U). A formula corresponding to each possible world (D) for a source is built by conjuncting all variables t_i where the corresponding tuple is in D_j, and conjuncting $\neg t_i$ where the corresponding tuple is not in D_j. The formula corresponding to the uncertain database U is then the disjunction of the formulae corresponding to the possible worlds of U.

The formula f corresponding to the uncertain database resulting from integrating $U_1 \ldots U_n$ is obtained by conjuncting the formulae of the databases: $f = f_1 \wedge \ldots \wedge f_n$. This procedure is best demonstrated through an example we present next. Let us consider the two friends of Bill who are providing information about his course registrations during Fall 2013. We refer to the first friend as Source S1 and the second one as Source S2. Let S1 state that Bill is taking CS100 or CS101 (but not both). Let S2 state that Bill is taking CS101 or CS102 (but not both). The corresponding possible worlds for this example are:

Table 3. Possible Worlds Model for Sources S1 and S2

D_1

Student	Course
Bill	CS100

S1

D_2

Student	Course
Bill	CS101

D_3

Student	Course
Bill	CS101

S2

D_4

Student	Course
Bill	CS102

Let variable t_1 and t_2 correspond to each of tuples (Bill, CS100) and (Bill, CS101) respectively. Then the formula for the first possible world, second possible world, and the database are, respectively,

$$t_1 \wedge \neg t_2,\ \neg t_1 \wedge t_2,\ \text{and}\ (t_1 \wedge \neg t_2) \vee (\neg t_1 \wedge t_2)$$

Let t_2 and t_3 correspond to (Bill, CS101) and (Bill, CS102) respectively. Then the formula corresponding to the uncertain database representing S2 is as shown next.

$$(t_2 \wedge \neg t_3) \vee (\neg t_2 \wedge t_3)$$

The integration is then obtained as,

$$[(t_1 \wedge \neg t_2) \vee (\neg t_1 \wedge t_2)] \quad \wedge \quad [(t_2 \wedge \neg t_3) \vee (\neg t_2 \wedge t_3)]$$

Simplifying this Boolean expression yields

$$(\neg t_1 \wedge t_2 \wedge \neg t_3) \qquad \vee \qquad (t_1 \wedge \neg t_2 \wedge t_3)$$

We interpret this to mean that the two possible worlds upon integration are, *(Bill registered for CS101)* or *(Bill registered for both CS100 and CS102).*

2.2 Data Integration using the Possible Worlds Model with probabilities

The integration approach developed in Section 2.1 is extended to deal with integrating uncertain information associated with *known probabilities* and determine the probabilities of the possible worlds in the integration [1]. Given a probabilistic uncertain database U with PW(U) = $\{D_1, \ldots, D_m\}$, it is convenient to associate a probabilistic event e_i with each possible world D_i. Intuitively, if e_i represents the event where the value of the uncertain database U is equal to D_i, then the probability of e_i, $P(e_i) = p_i$.

The work of [1] shows that the *probabilistic consistency constraint* has to be satisfied for performing uncertain data integration. It states that the sum of probabilities of the possible worlds corresponding to the first source should be equal to the sum of probabilities of the possible worlds corresponding to the second source for the possible worlds that are integrating. When possible worlds from different sources satisfy the consistency constraints, then the probabilities of integration are obtained in terms of the probabilities of the individual sources by using conditional probability : $P(e_j \wedge e_k) = P(e_j|e_k) * P(e_k)$. If e_j and e_k are inconsistent, then $P(e_j|e_k) = 0$.

If possible worlds D and D’ are connected in the consistency graph, and not connected to any other nodes, then $P(D) = P(D')$ and $P(D \wedge D') = P(D) = P(D')$, otherwise the probability is obtained by distributing the sum according to the pairwise product of probabilities of underlying possible worlds as shown in the ensuing example.

Consider the possible worlds of two sources shown in Table 4 and Table 5.

Table 4. Possible Worlds of Source S1

D_1

Student	Course
Bill	CS101

D_2

Student	Course
Bill	CS101
Bill	CS103

D_3

Student	Course
Bill	CS103

Let the probabilities of the possible worlds in the two sources be $P(D_1) = 0.3$, $P(D_2) = 0.5$, $P(D_3) = 0.2$, $P(D_1') = 0.35$, $P(D_2') = 0.45$, $P(D_3') = 0.05$, and $P(D_4') = 0.15$.

The lines in Figure 1 connecting S1 to S2 represent the possible worlds that are consistent and can integrate. There are two connected components in this consistency graph. The possible worlds in the result of integration are shown on right. The probabilistic consistency constraints $P(D_1) + P(D_2) = P(D_1') + P(D_2')$ and $P(D_3) = P(D_3') + P(D_4')$ are satisfied.

The probabilities of the integrated possible worlds are calculated in the following way.

$$P(D_3 \wedge D_3') = P(D_3 \mid D_3') * P(D_3') = P(D_3') = 0.05,$$

$$P(D_3 \wedge D_4') = P(D_3 \mid D_4') * P(D_4') = P(D_4') = 0.15$$

Table 5. Possible Worlds of source S2

D_1'

Student	Course
Bill	CS101

D_2'

Student	Course
Bill	CS101
Bill	CS102

D_3'

Student	Course
Bill	CS102

D_4'

Student	Course
Bill	CS102
Bill	CS100

The probability of the remaining four possible worlds in the integration is obtained by distributing the sum (0.8) according to the pairwise product of probabilities of underlying possible worlds.

$$P(D_1 \wedge D_1') = P(D_1 \mid D_1') * P(D_1') = P(D_1) / [P(D_1) + P(D_2)] \ * P(D_1') = 0.13125$$

$$P(D_1 \wedge D_2') = P(D_1 \mid D_2') * P(D_2') = P(D_1) / [P(D_1) + P(D_2)] \ * P(D_2') = 0.16875$$

$$P(D_2 \wedge D_1') = P(D_2 \mid D_1') * P(D_1') = P(D_2) / [P(D_1) + P(D_2)] \ * P(D_1') = 0.21875$$

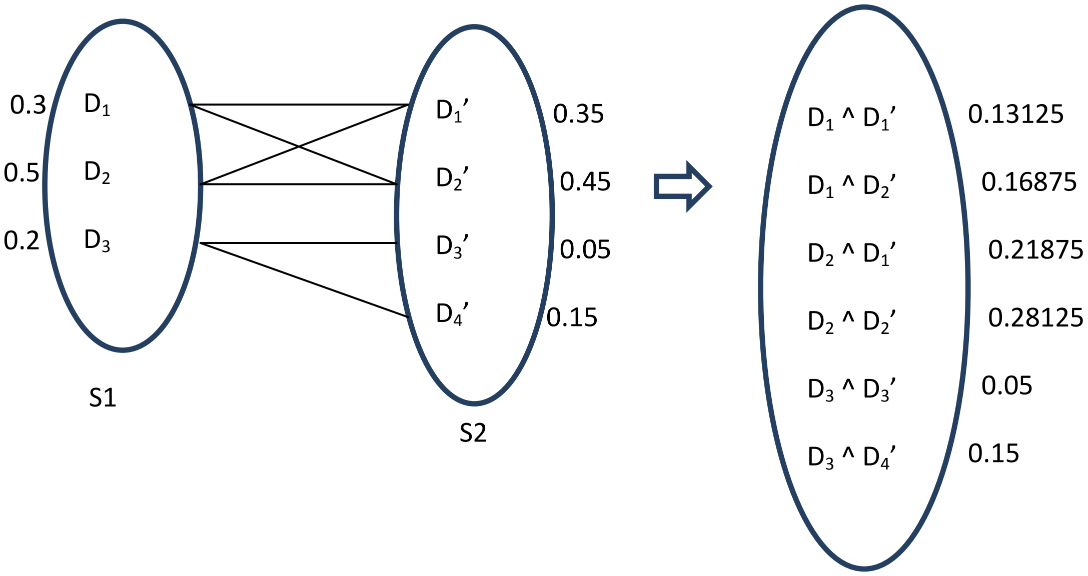
$$P(D_2 \wedge D_2') = P(D_2 \mid D_2') * P(D_2') = P(D_2) / [P(D_1) + P(D_2)] \ * P(D_2') = 0.28125$$

Since (D_2 ^ D_2') has the highest probability upon integration, the possible world *(Bill registered for CS101, CS103 and CS102)* is the most likely solution of integration.

2.3 Data Integration using the Probabilistic Relation Model without probabilities

Moving on from the Possible Worlds model, the work of [2] represents uncertain information from two sources using the *probabilistic relations* (pr-relation) model and integrates it. It introduces the *extended probabilistic relation* (epr-relation) model for representing the integration. An epr-relation is a pr-relation plus a set of event constraints that restrict the set of valid truth assignments, and hence, the set of possible worlds of an epr-relation. The ensuing section describes the integration algorithm in detail.

Figure 1. Result of Integrating with the Corresponding Probabilities



2.3.1 Integration Algorithm for Sources with epr-relations

Let r_1 and r_2 be pr-relations for sources S1 and S2, respectively. Let R = R' U {E} be the schema of r_1 and r_2, where R' is the set of regular attributes and E is the special event attribute. Let T_1 be the set of regular tuples (tuple-set) of r_1, that is, $T_1 = \{t (R') \mid t \in r_1\}$. Similarly, let T_2 be the tuple-set of r_2.

The result of integration of r_1 and r_2 is an epr-relation r obtained as follows.

- The Schema of r is the same as that of r_1 and r_2, namely, R = R' U {E}.
- The set of regular tuples (tuple-set) of r is $T = T_1 \cup T_2$.
- For a tuple $t \in T$, its Event attribute value in r is obtained as follows
  - If $t \in T - T_2$ (t is in only r_1) then copy the corresponding E value from r_1.
  - If $t \in T - T_1$ (t is in only r_2) then copy the corresponding E value from r_2.
  - If $t \in T_1 \cap T_2$ (t is in both r_1 and r_2) then copy the corresponding E value from either r_1 or r_2.

- Add the following event constraints to r: For each t ϵ $T_1 \cap T_2$, add the constraint $w_1 \equiv w_2$, where w_1 is the value of Event attribute for t in r_1 and w_2 is the value of Event attribute for t in r_2. We use $w_1 \equiv w_2$ to represent the formula $(w_1 \rightarrow w_2) \wedge (w_2 \rightarrow w_1)$.

We apply this algorithm to the example outlined in Section 2.1.

Table 6. pr-relation for Source S1 and S2

S1

Student	**Course**	**Event Attribute (E)**
Bill	CS100	x
Bill	CS101	¬x

S2

Student	**Course**	**Event Attribute (E)**
Bill	CS101	y
Bill	CS102	¬y

Table 7. Integrated epr-relation

Student	**Course**	**Event Attribute (E)**
Bill	CS100	x
Bill	CS101	¬x
Bill	CS102	¬y
	¬x ≡ y	

The event constraint in Table 7 (**¬x ≡ y**) implies that only the cases (x = true, y = false) and (x = false, y = true) are valid. All other cases are invalid. We already know that the truth table with x and y event variables gives us all the possible worlds. Eliminating all cases where (x ≡ y) labeling them as invalid assignments leaves us with the remaining valid possible worlds as: *(Bill registered for CS101)*, or *(Bill registered for both CS100 and CS102)*.

2.4 Motivation for This Work

The work of [2] has provided a compact representation for efficient integration of uncertain data. However, it does not take the probabilities that accompany such data into account. Integration methods using the possible worlds model that take the probabilities into account and calculate the probabilities of the result of integration are already available, but they are highly inefficient and impractical for the purposes of storing and processing large amounts of uncertain data. Therefore, we attempt to solve this problem of determining probabilities using the compact representation of pr-relation and the work of [2] provides a good starting point for our work.

In the next section, we will look at how possible worlds models with known associated probabilities can be represented using the compact representation of pr-relations by applying the conversion algorithm shown in the work of [13]. We will use this and the work of [2] as a starting point for our work which is presented in the ensuing chapters.

2.5 The Conversion Algorithm

This algorithm works in two steps. In the first step, the probabilities of the event variables are determined. Next, the corresponding pr-relations are formed.

2.5.1 Determining the Probabilities of the Event Variables

Let the possible worlds be D_1, D_2 . . . $D_{n.}$ Let the probability of the Possible Worlds P (D_i) = d_i, i = 1, 2 . . . n such that $\sum_i^n (d_i) = 1$. Let the event variables be x_1, x_2,.. x_{n-1}. Let the probabilities of the event variables P (x_i) = p_i.

We consider the 2^{n-1} truth assignments from the truth table for the event variables x_1, x_2 . . . x_{n-1}. We use a specific assignment of possible worlds D_1, D_2 . . . D_n to the truth assignments that facilitates the computation of probabilities P (x_i) = x_i, i = 1, 2 . . . n-1 in terms of possible worlds probabilities P (D_i) = d_i, i = 1, 2 . . . n.

- Assign D_1 to all combinations where x_1 = true (there are 2^{n-2} such combinations). This result in $p_1 = d_1$.
- Assign D_2 to all combinations where x_1 = false and x_2 = true (there are 2^{n-3} such combinations). This results in (1- p_1) $p_2 = d_2$. Hence, we obtain $p_2 = d_2 / (1 - d_1)$.

Note that d_1, d_2, and $(1 - d_1)$ are positive, and $d_1 + d_2 \leq 1$. So, we have $0 \leq p_2 \leq 1$.

- Assign D_3 to all combinations where x_1 = false, x_2 = false and x_3 = true (there are 2^{n-4} such combinations). This results in $(1- p_1)\ (1- p_2)\ p_3 = d_3$. Hence, we obtain $p_3 = d_3 / (1 - d_1 - d_2)$. Note that d_1, d_2, d_3 and $(1 - d_1 - d_2)$ are positive, and $d_1 + d_2 + d_3 \leq 1$. So, we have $0 \leq p_2 \leq 1$.
- Continuing in this manner, we get the general term $p_i = d_i / (1 – d_1 – d_2 - . . . – d_{i-1})$ or, equivalently $p_i = d_i / (d_n + d_{n-1} + d_{n-2} + . . . + d_i)$. Clearly, $0 \leq p_i \leq 1$.

  The n^{th} possible world, D_n, is assigned to the combination x_1 = false . . . x_{n-1} = false. We should obtain $(1- p_1)\ (1- p_2)\ . . .\ (1- p_{n-1}) = d_n$. We can verify this by noticing that, from the equation for p_i above,

$$(1 - p_i) = \frac{(d_n + d_{n-1} + d_{n-2} + \dots + d_{i+1})}{(d_n + d_{n-1} + d_{n-2} + \dots + d_i)}$$

  which yields,

$$\begin{aligned} (1 – p_1) (1 – p_2) \dots (1 – p_{n-1}) &= \frac{d_n}{(d_n + d_{n-1} + d_{n-2} + \dots + d_1)}. \\ &= d_n \end{aligned}$$

Thus, we have now obtained the probabilities for the event variables. We next proceed to form the pr-relations.

2.5.2 Forming the pr-relations

We can obtain the pr-relations, r, corresponding to the Possible Worlds D_1, D_2 . . . D_n. The pr-relations r contains the set of tuples D_1 U D_2 . . . U D_n.

- With each world D_i, i = 2, 3, . . . n-1 we associate the Boolean expression

$$f_i = \neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge. . . \wedge\neg x_{i+1} \wedge x_i$$

We provide the expression for D_1 as $f_1 = x_1$ for i=1, and the expression for the n^th term D_n as $f_n = \neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge. . . \wedge \neg x_{n-1}$. The value of the event attribute for a tuple 't' in r is obtained as $V_{t\epsilon Di}f_i$. In other words, the value of Event attribute (E) for t is the disjunction of the expressions associated with the possible worlds that contain t.

We present an example to demonstrate the working of the Conversion Algorithm. Consider the information about Bill's course registrations during Fall 2013, which is stored in an uncertain database. Let the set of all given tuples in the database be as shown in Table 8.

Table 8. Description of the Tuples

Tuple	Student	Course
t_1	Bill	CS101
t_2	Bill	CS103
t_3	Bill	CS102
t_4	Bill	CS100
t_5	Bill	CS104
t_6	Bill	CS105

Let Source S1 consist of three possible worlds, t_4, t_1t_3 and t_1t_2, with probabilities 0.08, 0.32 and 0.6 respectively. Let Source S2 consist of two possible worlds, t_1 and t_4t_5, with probabilities 0.4 and 0.6 respectively. Given this information, we now proceed to apply the Conversion Algorithm. We first list out the three possible worlds for Source S1 in Table 9. We also list out the two possible worlds for Source S2 in Table 10. We need two event variables since there are three possible worlds in Source S1, say x_1 and x_2. We next perform the following truth assignments:

- Assign the first possible world t_4 to all combinations where the truth value of x_1 is *True*. The corresponding probability for the event variable x_1 is $P(x_1) = P(t_4) = 0.6$

- Assign the second possible world t_1t_3 where the truth value of x_1 is *True* and x_2 is *False*. The corresponding probability for the event variable x_2 is,

$$P(x_2) = P(t_1t_3)/ [1 - P(x_1)] = 0.32/0.4 = 0.8$$

- Assign the last possible world t_1t_2 to the remaining row where both x_1 and x_2 are *False*. The truth assignments for the event variables x_1 and x_2 for source S1 are shown in Table 11.

Table 9. Possible worlds for Source S1

D_1, $P(D_1) = 0.08$

Student	Course
Bill	CS100

D_2, $P(D_2) = 0.32$

Student	Course
Bill	CS101
Bill	CS102

D_3, $P(D_3)=0.6$

Student	Course
Bill	CS101
Bill	CS103

Based on the truth assignments in Table 11, we now form the corresponding pr-relations in the following manner.

$$t_4 = x_1$$

$$t_1, t_3 = \neg x_1 x_2$$

$$t_1, t_2 = \neg x_1 \neg x_2$$

Hence simplifying the expression for t_1 gives $t_1 = \neg x_1 x_2 V \neg x_1 \neg x_2 = \neg x_1$

Table 10. Possible Worlds for Source S2

D_1', P(D_1') = 0.4

Student	Course
Bill	CS101

D_2', P(D_2') = 0.6

Student	Course
Bill	CS100
Bill	CS104

Table 11. Truth Assignments for Source S1

x_1	x_2	Possible World
0	0	t_1t_2
0	1	t_1t_3
1	0	t_4
1	1	t_4

We now proceed along similar lines for Source S2. Since we have two possible worlds, we need only one event variable, say y_1. We next perform the following truth assignments.

- Assign the first possible world t_1 to all combinations where the truth value of y_1 is *True*. The corresponding probability for the event variable y_1 is P (y_1) = P (t_1) = 0.4
- Assign the second possible world t_4t_5 where the truth value of y_1 is *False*.

Therefore, the truth assignments for Source S2 are as shown in Table 12. Based on Table 12, we form the corresponding pr-relations in the following manner:

$$t_4, t_5 = \neg\, y_1$$

$$t_1 = y_1$$

Table 12. Possible Worlds for Source S2

y_1	Possible World
0	t_4t_5
1	t_1

Thus the probabilistic relation and the probabilities of the event variables are as follows: $P(x_1) = 0.6$, $P(x_2) = 0.8$, $P(y_1) = 0.4$. At this point, we have successfully obtained the corresponding compact representation for the given data with probabilities ready to be applied for data integration. We represent the uncertain information in the form of pr-relations for Source S1 and Source S2 as shown in Table 13.

Table 13. pr-relations for Source S1

Source S1, $P(x_1) = 0.6$, $P(x_2) = 0.8$

Tuple	Student	Course	Event Attribute
t_1	Bill	CS101	$\neg x_1$
t_2	Bill	CS103	$\neg x_1 \neg x_2$
t_3	Bill	CS102	$\neg x_1 x_2$
t_4	Bill	CS100	x_1

Table 14. pr-relations for Source S2

Source S2, $P(y_1) = 0.4$

Tuple	Student	Course	Event Attribute
t_1	Bill	CS101	y_1
t_4	Bill	CS100	$\neg y_1$
t_5	Bill	CS104	$\neg y_1$

Clearly, Table 13 and Table 14 provide a more compact representation for this problem compared to Table 9 and Table 10 respectively.

In summary, we use this conversion algorithm to arrive at a compact representation of the uncertain data with the associated probabilities to determine the probabilities of the result of integration using our method as presented in the ensuing chapters.

CHAPTER III

PROBABILITIES FOR UNCERTAIN DATA INTEGRATION

In this chapter, we seek to introduce the necessary ideas that help us to successfully compute the probabilities associated with the integrated results over uncertain databases. We introduce the set of notations that we will use throughout the rest of this work. We also introduce the idea of an Event Variable Formula (EVF) that will form the basis for computing the probabilities of the integrated result.

3.1 Computing Probabilities in Data Integration

Given two sources, let us assume that we have the possible worlds with known associated probabilities available to us. Say, we have obtained the corresponding pr-relations with the probabilities using the conversion algorithm. At this point, we present two methods of obtaining probabilities of the possible worlds in the integration.

3.1.1 Using the pr-relations

- We start with the pr-relations and construct the EVFs for every possible world of both sources.
- We integrate the two sources to obtain the resulting possible worlds.
- For each of these possible worlds, we combine the relevant EVFs to obtain the EVF for the possible worlds in the integration.
- We finally obtain the probability of the possible worlds in the integration in terms of the probabilities of the event variables of the EVF.

3.1.2 Using the epr-relation

- We start with the epr-relation and obtain the corresponding possible worlds in integration.
- We build EVFs for each of these possible worlds thus obtained.
- We finally obtain the probability of the possible worlds in the integration in terms of the probabilities of the event variables of the EVF.

We show that the EVFs and the probabilities obtained for every possible world in the integration is the same, irrespective of whether we use the pr-relations or the epr-relation.

3.2 Introducing Notations

- A pr-relation shall be denoted by r
- The schema for the database shall be denoted by R
- The schema of r is R U {E}, where E is the E*vent* Attribute
- The set of event variables for r shall be denoted by V
- A tuple in r has the form t@f, where t is the tuple and f is the corresponding Event Attribute
- r has n tuples t_1@f_1, . . . , t_n@f_n. We assume $t_i \neq t_j$ for all $i \neq j$
- If t@f ϵ r, we also say the regular tuple t is in r. The set of regular tuples of a pr-relation or epr-relation r is denoted by T(r). T(r) is also the tuple-set of the uncertain relation r.
- Possible Worlds of r, PW(r), is the set of relations $r_1, \ldots, r_k$ on schema R. The possible worlds contain regular tuples, $r_i \subseteq T(r)$, i = 1, . . . , k.
- We have two sources, represented by pr-relations r and s
- r and s have the same schema: R U {E}
- The set of event variables for s is W. We assume $V \cap W = \emptyset$

- s has m tuples, $u_1@g_1, \dots, u_m@g_m$
- Possible Worlds of s, PW(s), is the set of relations $s_1, \dots, s_l$ on schema R
- The integration of r and s is denoted by r Ũ s. We overload the notation and use r_i Ũ s_j to denote the result of integrating possible worlds $r_i \in PW(r)$ and $s_j \in PW(s)$. If r_i and s_j are compatible, then r_i Ũ s_j = r_i U s_j . If r_i and s_j are not compatible, they cannot be integrated and r_i Ũ s_j is nil.
- Without loss of generality, we assume r and s have p common (regular) tuples, namely, $t_k = u_k$ for k = 1. . . p. If p = 0, the two pr-relations do not have any common tuples.

3.3 Event Variable Formula (EVF)

An Event Variable Formula (EVF) is a logic based *formula* obtained as a result of performing Boolean operations over multiple event variables. Since the probabilities of the event variables are already known, the EVF can be used to obtain the probabilities of the integrated results. The following section provides a formal definition of the EVF in the scenario where it can be used to integrate from two pr-relations and epr-relation.

3.3.1 EVF Corresponding to a pr-relation

Building an EVF corresponding to a pr-relation is a two-step process as highlighted in Section 3.1.1. In the ensuing sections, we delve deeper into it.

3.3.1.1 EVF Corresponding to a possible world of the pr-relation

Let r be the pr-relation on the schema R U {E} and $t_1@f_1 \dots t_n@f_n$ be the tuples of r. Consider a relation $r_i \subseteq \{t_1 \dots t_n\}$ in the possible world of r, $r_i \in PW(r)$. We define the EVF φ_i corresponding to the possible world r_i as,

$$\varphi_i = \wedge_{tk \in ri} f_k \wedge_{tk \notin ri} \neg f_k$$

Since $T(r) = \{t_1 ... t_n\}$, we can write the EVF of $r_i \in PW(r)$ in the form,

$$\varphi_i = \wedge_{tk\in ri} f_k \wedge_{tk\in(T(r)- ri)} \neg f_k$$

Observation 1: For a relation $r_i \notin PW(r)$, the EVF $\varphi_i = \wedge_{tk\in ri} f_k \wedge_{tk\in(T(r)- ri)} \neg f_k$ is false

Proof: If $r_i \notin PW(r)$ then no truth assignment v to event variables V exists for which (1) all f_j are true for $t_j \in r_i$, and (2) all f_j are false for $t_j \in (T(r) - r_i)$. Hence, φ_i is not satisfiable (it is false for all truth assignments to event variables V).

Corollary 1: For a pr-relation r, $r_i \in PW(r)$ if and only if φ_i is satisfiable.

3.3.1.2 EVF Corresponding to Integration of pr-relations from Two Sources

Let r and s on the schema R U {E} be the two pr-relations in the integration. V and W are the set of event variables for r and s respectively, and $V \cap W = \phi$. Let $r_i \in PW(r)$ and $s_j \in PW(s)$ with EVFs φ_i and ψ_j, respectively. If $r_i \in PW(r)$ and $s_j \in PW(s)$ are compatible, then the EVF for the possible world in the integration of r and s represented by ξ is obtained by conjuncting the EVFs for r_i and s_j respectively.

$$\xi = \varphi_i \wedge \psi_j$$

where, $\varphi_i = \wedge_{tk\in ri} f_k \wedge_{tk\in(T(r) - ri)} \neg f_k$ and

$$\psi_j = \wedge_{tk\in sj} g_k \wedge_{tk\in(T(s) - sj)} \neg g_k$$

3.3.2 EVF Corresponding to an epr-relation

For an epr-relation r with set of tuples $t_1@f_1 \ldots t_n@f_n$ and event constraints $c_1, c_2 \ldots c_n$ we can define EVFs as defined in Section 3.3.1 for a relation $r_i \subseteq \{t_1 \ldots t_n\}$ that is in the possible world of r as follows,

$$\varphi_i = \wedge_{tk \epsilon ri} f_k \wedge_{tk \epsilon (T(r) - ri)} \neg f_k$$

<u>Observation 2:</u> For an epr-relation r and a relation $r_i \subseteq \{t_1 \ldots t_n\}$ that is not in the possible world of r, the EVF $\varphi_i = \wedge_{tk \epsilon ri} f_k \wedge_{tk \epsilon (T(r) - ri)} \neg f_k \wedge (c_1 \wedge c_2 \wedge \ldots c_n)$ is *false*, where $c_1, c_2 \ldots c_n$ are event constraints of r. Thus φ_i is *true* for valid truth assignments to event variables in V that yield the possible world r_i and *false* for all other valid truth assignments. Hence the event constraints needs to be satisfied to construct a valid EVF.

<u>Observation 3:</u> For an epr-relation r and $r_i \subseteq T(r)$ with the EVF φ_i, we observe that:

- $r_i \epsilon PW(r)$, if at least one valid truth assignment exists that makes φ_i true.
- If all valid truth assignments make φ_i false, then $r_i \notin PW(r)$.

<u>Corollary 2:</u> For an epr-relation r, $r_i \epsilon PW(r)$ iff $\varphi_i \wedge C$ is satisfiable, where $C = c_1 \wedge c_2 \wedge \ldots c_n$ is the conjunction of all event constraints of r.

<u>Theorem 1:</u> Given two pr-relations r and s on the schema R U {E}, let q be the epr-relation that represents the integration of r and s i.e $q = r \tilde{U} s$. Let $r_i \epsilon PW(r)$ and $s_j \epsilon PW(s)$ with EVFs φ_i and ψ_j, respectively. Let $\xi = \varphi_i \wedge \psi_j$, and μ be a truth assignment to variables in V and W. Then, if r_i and s_j are compatible, and if ξ is true under μ, then μ is a valid truth assignment. That is, all event

constraints of q are satisfied under μ. This means $\xi \rightarrow C$, where C is the conjunction of event constraints of q.

Proof: Assume, without loss of generality, that r and s have p common regular tuples $t_k = u_k$, k = 1 . . . p. Alternatively, $T(r) \cap T(s) = \{t_1 \ldots t_p\} = \{u_1 \ldots u_p\}$. Then $q = r \tilde{U} s$ has p event constraints $f_k \equiv g_k$, k = 1 . . . p. Since r_i and s_j are compatible, then by Lemma in [2], there is no regular tuple $t_k \in T(r) \cap T(s)$ such that $t_k \in r_i$ and $t_k \notin s_j$ or vice versa. Further, since ξ is true, then φ_i and ψ_j, are true. It follows that for all $t_k \in T(r) \cap T(s)$, either:

(1) $t_k \in r_i$ and $t_k \in s_j$ and both f_k and g_k are true under truth assignment μ, or

(2) $t_k \notin r_i$ and $t_k \notin s_j$ and f_k and g_k are both false under truth assignment μ.

Hence, the event constraints $f_k \equiv g_k$, k = 1 . . . p are satisfied under μ.

<u>Theorem 2:</u> Let r, s, r_i, s_j, φ_i, ψ_j, ξ, and μ be as defined in Theorem 1. Then ξ is the EVF associated with possible world $q_{ij} = r_i \tilde{U} s_j$ of epr-relation $q = r \tilde{U} s$.

Proof: Let $r = \{t_1@f_1 \ldots t_n@f_n\}$ and $s = \{u_1@g_1 \ldots u_m@g_m\}$, with p common (regular) tuples, $t_k = u_k$, k = 1 . . . p. Then, one possible set of tuples for $q = r \tilde{U} s$ is,

$$q = \{t_1@f_1 \ldots t_n@f_n\} \text{ U } \{u_{p+1}@g_{p+1} \ldots u_m@g_m\}$$

Consider the truth assignment μ to event variables V U W. By Theorem 1, if ξ is true under μ, then μ is legal. Further, if ξ is true under μ, then so are φ_i and ψ_j. Hence, f_k is true for all tuples $t_k \in r_i$, and it is false for all tuples $t_k \in (T(r)- r_i)$. Similarly, g_k is true for all tuples $u_k \in s_j$, and it is false for all tuples $u_k \in (T(s)- s_j)$. Consider a tuple $v@h \in q = r \tilde{U} s$. ($v@h$ is either a $t_k@f_k$ or a $u_k@g_k$). In this case h is true under μ iff $v \in r_i \text{ U } s_j$. It follows that ξ is true for all valid truth assignments that yield the possible world q_{ij}, and, hence, ξ is the EVF for q_{ij}.

3.4 Computing the Probabilities from the EVF

Once the EVFs of the possible worlds in integration are obtained, we can compute their corresponding probabilities using the probabilities of the event variables in the EVF. The event variables are assumed independent, except those related through event constraints.

- If the event variables are independent, the probability of the integrated possible world is simply the product of the probabilities of the individual event variables in the EVF. For example the probability associated with EVF $(b\wedge\neg c)$ is $P(b\wedge\neg c) = P(b) * (1 - P(c))$.
- On the other hand, if the event variables are not independent, conditional probability is used to determine the resulting probability. For example, an event constraint of the form $a \equiv d$ enforces $P(a) = P(d)$. The probability associated with EVF $(a\wedge d)$ is $P(a\wedge d) = P(a|d) * P(d)$. In this case, $P(a|d) = P(d|a) = 1$ and hence $P(a\wedge d) = P(a) = P(d)$.

3.5 Example for Determining the Probabilities

We present an example to demonstrate the calculation of the probabilities using the two methods explained in Section 3.1.1 and 3.1.2. Consider two sources represented compactly using the pr-relations along with the probabilities of the event variables as shown in Table 15. Consider the integrated epr-relation shown in Table 16. Let the set of all given tuples in the database be as shown in Table 8.

3.5.1 EVF using the pr-relation

We build the EVF of the possible worlds in the integration using the method described in Section 3.1.1.

We first obtain the Possible Worlds in S1 using the truth value of x. This gives $r_1 = \{\phi\}$, $r_2 = \{t_1\}$. The EVF corresponding to the possible worlds of S1 are $\neg x$ and x respectively. The

Possible Worlds in S2 are $s_1 = \{\phi\}$, $s_2 = \{t_3\}$, $s_3 = \{t_1, t_2\}$, and $s_4 = \{t_1, t_2, t_3\}$. The EVF corresponding to the possible worlds of S2 are $\neg y \wedge \neg z$, $\neg y \wedge z$, $y \wedge \neg z$, and $y \wedge z$, respectively.

The compatible pairs of possible worlds from r and s are (r_1, s_1), (r_1, s_2), (r_2, s_3), and (r_2, s_4). The EVFs for these possible worlds in the integration are $\neg x \wedge \neg y \wedge \neg z$, $\neg x \wedge \neg y \wedge z$, $x \wedge y \wedge \neg z$, and $x \wedge y \wedge z$, respectively.

Table 15. pr-relations for Sources S1 and S2

Source S1, P(x) = 0.8

Tuples	Event Attribute
t_1	x

Source S2, P(y) = 0.8, P(z) = 0.2

Tuples	Event Attribute
t_2	y
t_3	z
t_1	y

Table 16. epr relation for the Result of Integration

Tuples	Event Attribute
t_1	x
t_2	y
t_3	z

x ≡ y

3.5.2 EVF using the epr-relation

We build the EVF of the possible worlds in the integration using the method described in Section 3.1.2.

We first obtain the Possible Worlds of the integrated epr-relation using the truth values of x, y, z that eliminate invalid cases based on the event constraints. This gives $\{\phi\}$, $\{t_3\}$, $\{t_1, t_2\}$, and $\{t_1, t_2, t_3\}$. In the next step, the EVF corresponding to these possible worlds are obtained. These are $\neg x \wedge \neg y \wedge \neg z$, $\neg x \wedge \neg y \wedge z$, $x \wedge y \wedge \neg z$, and $x \wedge y \wedge z$, respectively.

3.5.3 Calculating the Probabilities from the EVF

The probabilities are calculated using the method described in Section 3.4. In this case, event variables x and y are dependent. The possible worlds in integration and the corresponding EVFs obtained in both 3.5.1 and 0 for this example have matched exactly.

$P(\phi) = P(\neg x \wedge \neg y \wedge \neg z) = P(\neg x|\neg y) * P(\neg y) *P(\neg z) = P(\neg y) *P(\neg z) = 0.16$

$P(t_3) = P(\neg x \wedge \neg y \wedge z) = P(\neg x|\neg y) * P(\neg y) *P(z) = P(\neg y) *P(z) = 0.04$

$P(t_1t_2) = P(x \wedge y \wedge \neg z) = P(x|y) * P(y) *P(\neg z) = P(y) *P(\neg z) = 0.64$

$P(t_1t_2t_3) = P(x \wedge y \wedge z) = P(x|y) * P(y) *P(z) = P(y) *P(z) = 0.16$

Thus, we get matching probabilities from both methods in this case. Using the two *methods we introduced,* we have obtained the probabilities of the possible worlds in integration. Since the possible world *{t_1, t_2}* has the highest probability upon integration, the possible world *(Bill registered for CS101 and CS103)* is the most likely solution of integration.

CHAPTER IV

FORMAL RESULTS

We have now constructed the EVF for the possible worlds in the integration starting from both pr- and epr-relations. We have also presented how we can compute the probabilities using the EVF. In this chapter, we verify that the EVF generated by using both the methods are equivalent and the resulting probabilities are also equal.

4.1 Results

Our verification methodology involves comparing the EVFs generated by the pr-relations and the epr-relations.

Firstly, we start with an integrated epr-relation and decompose it into its corresponding pr-relations. However, the process of decomposition may not necessarily always lead to a unique pair of pr-relations. Hence, we compute the EVF for each such pair and check if these formulae match exactly. If yes, we also compare it against the EVF generated by the epr-relation. Matching formulae will always ensure matching probabilities as well.

Since we have restricted ourselves to integrating data from only two sources in this work, an integrated epr-relation may not be obtainable by the integration of two pr-relations. Hence, we use the *sufficient* conditions to check beforehand if it can even be broken down into exactly two pr-relations. Section 4.2 presents these conditions in detail.

Secondly, we start with two pr-relations and obtain the integrated epr-relation. We simply ensure that the event variables in both the EVFs are equivalent along with the same truth values. The probabilities generated by both the EVFs should therefore also match exactly.

4.2 Sufficient Conditions for an epr-relation to be Integrated

Given an epr-relation q, we say q is integrated if a pair of non-empty pr-relations r and s exists such that q = r Ũ s. The integration algorithm in [2] is non-deterministic and can produce 2^p different epr-relations, where p refers to the number of common tuples.

<u>Theorem 3:</u> An epr-relation q is integrated if a partition (V, W) of event variables of q exists such that,

1) For each tuple t@f ϵ q, all event variables appearing in f are in V or all are in W.
2) For each event constraint f ≡ g of q, all event variables appearing in f are in V and all event variables appearing in g are in W, or vice versa.
3) For each event constraint f ≡ g of q, there is a unique (regular) tuple t such that t@f ϵ q or t@g ϵ q.

Proof: We show that if conditions of Theorem 3 hold, then the *Decomposition Algorithm* described in Section 4.2.1 can be used to produce pr-relations r and s such that q ≡ r Ũ s. We assume that Step 3(a) and Step 3(b) of this algorithm partition the tuples of q onto pr-relations r and s. By condition (1) of Theorem 3, this partition will be well-defined. Then, Step 3(c) of the algorithm adds more tuples to r and/or s to complete the construction. Thus, q is now decomposed into pr-relations r and s.

Next, we show that given an epr-relation q, pr-relations r and s produced by the *Decomposition Algorithm* should satisfy r Ũ s ≡ q. Assume r Ũ s = q'. We first verify that q' has the same set of event constraints as q. For each constraint f ≡ g in q, by Conditions (2) and (3) of Theorem 3, there is a unique tuple t@f or t@g in q, and, by Step 3(b) of the *Decomposition Algorithm*, in r or s. Without loss of generality, assume t@f ϵ r. Step 3(c) of the *Decomposition Algorithm* adds t@g to s. Then the integration algorithm in the work of [2] generates f ≡ s for q' = r Ũ s.

Finally, we show that q’ has the same (or equivalent) set of tuples as q. By the *Decomposition Algorithm*, for all t@f ϵ q, either t@f ϵ r or t@f ϵ s. Then, by the integration algorithm of [2], either t@f ϵ q’ or t@g ϵ q’ for some g such that f ≡ g. It, thus, follows that set of tuples of q’ and q are equivalent.

4.2.1 The Decomposition Algorithm

Given an epr-relation q, this algorithm determines whether q satisfies the sufficient conditions of Theorem 3, and if it does, generates partitions (V, W) of event variables of q and obtains all pairs of two pr-relations (r,s). We introduce the idea of an Extra-Set which is the set of Event Attributes in q that are not specifically associated with partition V or W. The number of different pairs of pr-relations depends on the size of Extra-Set. Specifically, it will be $2^{|\text{Extra-Set}|}$. The algorithm works in three steps.

In the first step, it checks if conditions 2 and 3 of Theorem 3 are satisfied and identifies the event variables in event constraints that should appear separately (in V and W). In the second step, it checks if condition 1 of Theorem 3 is satisfied and identifies event variables in Event Attributes that should appear separately (in V and W). Finally, in the third step it obtains all pairs of two pr-relations (r,s) based on the partition (V, W) generated in the first 2 steps.

Algorithm:

Input: epr-relation q with tuples {v_1@h_1. . . v_l@h_l} and constraints $f_i = g_i$, i = 1 . . .p. Let ev(q) be the set of event variables appearing in q (in h_i’s, f_i’s, g_i’s).

Initialization: Initialize the partition sets V and W to Null. Initialize the set Extra-Set to Null.

Step 1: In this step, the event constraints occurring on either sides of ≡ symbol needs to be partitioned into V and W.

The first part of the if condition checks if condition 2 of Theorem 3 is satisfied, by verifying if all event variables in f are in V (or W) and all event variables in g are in W (or V) and therefore not in both. The second part of the if condition checks condition 3 of Theorem 3.

The else part adds the event variables in the constraints to V or W in the following way. It checks if a subset of f or g is already present in V or W, in which case it adds the remaining event variables in that Event Attribute to the same partition. Otherwise, all event variables in f are added to V and all event variables in g are added to W. The pseudo code for Step 1 is,

```
For each event constraint $f \equiv g$ of q:
        If(($f \subseteq V$ (or W) and $g \subseteq V$ (or W)) or !($\exists$ unique tuple t such that t@f $\epsilon$ q or t@f $\epsilon$ q))
                Exit()
        Else
                If $f \subseteq V$ (or W), add f - V (or f – W) to V (or W) and g to W (or V)
                Else if $g \subseteq V$ (or W), add g - V (or g – W) to V (or W) and f to W (or V)
                Else add f to V (or W) and g to W (or V)
End
```

Step 2: In the second step, the remaining event variables in the Event Attributes are partitioned based on existing event variables in V and W.

The if condition checks if a subset of event variables in that Event Attribute is in V and also in W, in which case the partition is not possible.

The else part adds the event variables in the Event Attribute in the following way. It checks if a subset of the Event Attribute is already present in V or W, in which case it adds the remaining event variables to the same partition V or W. Otherwise, it adds the Event Attribute to the Extra-Set. The pseudo code for Step 2 is,

```
For each ev (q):
        If(ev (q) ⊆ V and ev(q) ⊆ W)
                Exit()
        Else
                If ev (q) ⊆ V, add (ev (q) – V) to V
                Else If ev (q) ⊆ W, add (ev (q) – W) to W
                Else add (ev (q) - Extra-Set) to Extra-Set
End
```

Step 3: In the third step, all pairs of two pr-relations (r,s) is generated in the following way.

Let ev (f) represent the set of event variables of Event Attribute f.

a) Partition tuples of q as follows: Let r = {t@f є q | ev(f) ⊆ V} and s = {t@f є q | ev(f) ⊆ W}.

b) Add each element of the Extra-Set to r or s in $2^{|\text{Extra-Set}|}$ ways.

c) For each constraint f ≡ g of q, if t@f є r (or t@f є s), then add t@g to s (or to r), or if t@g є r (or t@g є s), then add t@f to s (or to r).

4.2.2 Examples

To demonstrate the *Decomposition Algorithm*, we consider the following example. Let the set of all given tuples in the database be as shown in Table 8.

Firstly, the initialization V = ɸ, W = ɸ, Extra-Set = ɸ is done.

Step 1: Here, Condition 3 is satisfied since $t_1 = x$. x is added to V and y is added to W. At the end of Step 1, V = {x} and W = {y}. The partition satisfies Condition 2.

Step 2: In this step, z is added to V and w is added to W. {vk} is added to Extra-Set. At the end of Step 2, V = {x,z} W = {y,w} Extra-Set = {{vk}}

Step 3: The decomposition of q results in two pairs of (r,s) as shown in Table 18 and Table 19.

Table 17. epr-relation

Tuples	Event Attribute (E)
t_1	x
t_2	$\neg x$
t_3	$\neg y$
t_4	xz
t_5	wy
t_6	vk

$x \equiv y$

Table 18. pr-relation r, s for Sources S1 and S2 (Pair #1)

Tuples	Event Attribute (E)
t_1	x
t_2	$\neg x$
t_4	xz
t_6	vk

Tuples	Event Attribute (E)
t_1	y
t_3	$\neg y$
t_5	wy

Table 19. pr-relation r, s for Sources S1 and S2 (Pair #2)

Tuples	Event Attribute (E)
t_1	x
t_2	$\neg x$
t_4	xz

Tuples	Event Attribute (E)
t_1	y
t_3	$\neg y$
t_5	wy
t_6	vk

Consider yet another example in Table 20 for which Decomposition is not possible. Decomposition may not be possible if the event variables in the Event Attributes cannot be partitioned into (V,W).

Table 20. epr-relation

Tuples	Event Attribute (E)
t_1	xy
t_2	x
t_3	$\neg y$

$x \equiv y$

Firstly the initialization, $V = \phi$, $W = \phi$, Extra-Set = ϕ is done.

Step 1: Here, Condition 3 is satisfied since $t_2 = x$. At the end of Step 1, $V = \{x\}$, $W = \{y\}$. The partition satisfies Condition 2.

Step 2: In this step, partition of Event Attributes is not possible. This is because, the first Event attribute {xy} implies that event variables x and y should be in the same partition, but x and y have been already separated into different partitions in Step 1. Thus there is a contradiction and hence decomposition is not possible.

4.3 Theorem 4

Given an extended probabilistic relation (epr-relation) q which is obtainable by integrating two pr-relations, let S be the set of all pairs (r,s) of two probabilistic relations (pr-relation) whose integration results in q. We prove that all pairs (r,s) in S give exactly the same EVF for the integrated possible worlds.

Proof: Without loss of generality, assume $r = \{t_1@f_1 . . . t_n@f_n\}$ and $s = \{u_1@g_1, . . . , u_m@g_m\}$ be an (r,s) pair obtained by *Decomposition Algorithm* for an epr-relation q. Also assume, without loss of generality that $t_i = u_i$ for $i = 1. . . p$. Note that $T(q) = \{t_1, . . . , t_n\} \cup \{u_1, . . . , u_m\}$.

Let $a_k = \{a_1, a_2 . . a_e . . a_l\}$ be a possible world obtained by integrating the possible worlds of two pr-relations. As a special case, a_k can be the empty relation ($a_k = \phi$) when $l = 0$. Let $\{a_e\}$ be a member of the Extra Set that can freely move to any source. The tuples in this possible world is a subset of the tuples in q, $a_k \subseteq T(q)$.

There can be several pairs (r,s) that can lead to the integrated possible world $\{a_1, a_2 . . a_e . . a_k\}$. Let us consider 2 possibilities here,

- $r_i = \{a_1, a_2 . . a_e . . a_t\} \in PW(r)$ and $s_j = \{a_{t+1} . . . a_z\} \in PW(s)$, or
- $r'_i = \{a_1, a_2 . . . a_t\} \in PW(r')$ and $s'_j = \{a_{t+1} . . a_e . . a_z\} \in PW(s')$

The EVF for r_i and s_j and their integration is,

$$\varphi_i = \Lambda_{a_k \epsilon r_i} f_k \, \Lambda_{a_k \epsilon (T(r) - r_i)} \neg f_k \qquad \text{and} \qquad \psi_j = \Lambda_{a_k \epsilon s_j} g_k \, \Lambda_{a_k \epsilon (T(s) - s_j)} \neg g_k$$

$$\xi = \varphi_i \Lambda \psi_j = \Lambda_{a_k \epsilon r_i} f_k \, \Lambda_{a_k \epsilon (T(r) - r_i)} \neg f_k \, \Lambda_{a_k \epsilon s_j} g_k \, \Lambda_{a_k \epsilon (T(s) - s_j)} \neg g_k$$

Similarly, the EVF for r'_i and s'_j and their integration is,

$$\varphi'_i = \Lambda_{a_k \epsilon r'_i} f'_k \, \Lambda_{a_k \epsilon (T(r') - r'_i)} \neg f'_k, \qquad \text{and}$$

$$\psi'_j = \Lambda_{a_k \epsilon s'_j} g'_k \, \Lambda_{a_k \epsilon (T(s') - s'_j)} \neg g'_k$$

$$\xi' = \varphi'_i \Lambda \psi'_j = \Lambda_{a_k \epsilon r_i} f'_k \, \Lambda_{a_k \epsilon (T(r) - r_i)} \neg f_k \, \Lambda_{a_k \epsilon s_j} g'_k \, \Lambda_{a_k \epsilon (T(s) - s_j)} \neg g'_k$$

The *Decomposition* A*lgorithm* gives the partition of the event variables with two pieces of information. Firstly, the event variables that should be in fixed sources and secondly, the Extra-Set of event variables that can be in any source. That is

if a_k ϵ Extra Set,

a_k ϵ r → Possibility #1

a_k ϵ s → Possibility #2

else,

a_k ϵ r or a_k ϵ s (fixed for every possibility)

Consider the component, $\Lambda_{a_k \epsilon r_i} f_k \, \Lambda_{a_k \epsilon s_j} g_k$ in r_i and s_j, and compare it with the component $\Lambda_{a_k \epsilon r'_i} f'_k \, \Lambda_{a_k \epsilon s'_j} g'_k$ in r'_i and s'_j. For $a_k = a_e$ if a_e ϵ r_i, $f_k = f_e$ or if a_e ϵ s'_j, $g'_k = g_e$. For all the other tuples where $a_k \neq a_e$, the EVFs will be the same. It follows that the two components have exactly the same terms for $a_e \epsilon r$ and for $a_e \epsilon s'$. A similar argument shows that the second components of the expressions (with $\neg f_k$ and $\neg g_k$) are also equal.

In the same way, the elements of the Extra Set can be moved one by one from one source to the other giving rise to different possibilities. However, this will not change the EVF for the integrated possible worlds. The EVF remains the same for every possible world including an invalid one. For this case, all possibilities result in a false value. Let us consider an example that demonstrates the results of this Theorem as shown below.

4.3.1 Example

Consider the epr-relation q of Table 21. Let the set of all given tuples in the database be as shown in Table 8.

Table 21. epr-relation q

Tuples	Event Attribute
t_1	a
t_2	b
t_3	¬c

a ≡ c

The possible worlds of q are $\{t_1\}$, $\{t_1,t_2\}$, $\{t_3\}$, and $\{t_2,t_3\}$ with EVF $a \wedge \neg b \wedge c$, $a \wedge b \wedge c$, $\neg a \wedge \neg b \wedge \neg c$, and $\neg a \wedge b \wedge \neg c$, respectively. Applying the *Decomposition Algorithm* on q will produce the two pr-relations shown in Table 22 and Table 23.

Table 22. pr-relations r and s

S1

Tuples	Event Attribute
t_1	a
t_2	b

S2

Tuples	Event Attribute
t_1	c
t_3	¬c

Possible worlds of r are $r_1 = \phi$, $r_2 = \{t_1\}$, $r_3 = \{t_2\}$, and $r_4 = \{t_1,t_2\}$ with EVF $\neg a \wedge \neg b$, $a \wedge \neg b$, $\neg a \wedge b$, and $a \wedge b$, respectively. Possible worlds of s are $s_1 = \{t_1\}$, and $s_2 = \{t_3\}$, with EVF c and $\neg c$, respectively. The compatible pairs of possible worlds from r and s are (r_2, s_1), (r_4, s_1), (r_1, s_2), and (r_3, s_2). The EVFs for these possible worlds in the integration are $a \wedge \neg b \wedge c$, $a \wedge b \wedge c$, $\neg a \wedge \neg b \wedge \neg c$, and $\neg a \wedge b \wedge \neg c$, respectively. Next, consider the pr-relations in the following Table.

Table 23. pr-relations r’ and s’

S1

Tuples	Event Attribute
t_1	a

S2

Tuples	Event Attribute
t_1	c
t_2	b
t_3	$\neg c$

Possible worlds of r’ are $r_1' = \{\phi\}$, and $r_2' = \{t_1\}$ with EVF $\neg a$, a, respectively. Possible worlds of s’ are $s_1' = \{t_1\}$, $s_2' = \{t_3\}$, $s_3' = \{t_1,t_2\}$, and $s_4' = \{t_2,t_3\}$, with EVF $\neg b \wedge c$, $\neg b \wedge \neg c$, $b \wedge c$ and $b \wedge \neg c$, respectively. The compatible pairs of possible worlds from r’ and s’ are (r_2', s_1'), (r_2', s_3'), (r_1', s_2'), and (r_1', s_4'). The EVFs for these possible worlds in the integration are $a \wedge \neg b \wedge c$, $a \wedge b \wedge c$, $\neg a \wedge \neg b \wedge \neg c$, and $\neg a \wedge b \wedge \neg c$, respectively.

Thus the integration of pairs of compatible possible worlds in different possibilities yields exactly the same possible worlds with the same EVF.

4.4 Theorem 5

Consider an integrated epr-relation q that satisfies the conditions of Theorem 3. Consider a possible world q_k of q. There are different ways of obtaining EVF of q_k: One possibility is to obtain it directly from q. Another possibility is to use the *Decomposition Algorithm* to obtain two sources r and s whose integration yield q, there may be multiple pairs of sources with this property. Then obtain the EVF for q_k by conjuncting the EVF of possible worlds r_i and s_j of the two sources whose integration yield q_k. We show that the EVF obtained in different ways are equivalent, with respect to the event constraints of q.

Proof: Without loss of generality, assume $r = \{t_1@f_1 \ldots t_n@f_n\}$ and $s = \{u_1@g_1 \ldots u_m@g_m\}$ be an (r,s) pair obtained by *Decomposition Algorithm* for an epr-relation q. Also assume, without loss of generality that $t_i = u_i$ for $i = 1 \ldots p$. Here $T(q) = \{t_1 \ldots t_n\} \cup \{s_1 \ldots s_m\}$. Assume $q_k = \{v_1 \ldots v_l\}$. As a special case, q_k can be empty ($q_k = \phi$) when $l = 0$. The EVF for q_k obtained from q is $\xi_k = \wedge_{vi \epsilon qk} h_i \wedge_{vi\epsilon(T(q)-qk)} \neg h_i$ where h_i is the event attribute value associated with v_i in q. That is, $v_i@h_i \ \epsilon \ q$. Alternatively, consider $r_i \ \epsilon \ PW(r)$ and $s_j \ \epsilon \ PW(s)$ such that $r_i \tilde{U} s_j = q_k$. The EVF for r_i and s_j are $\varphi_i = \wedge_{tk\epsilon ri} f_k \ \wedge_{tk\epsilon(T(r)- ri)} \neg f_k$, and $\psi_j = \wedge_{sk\epsilon sj} g_k \wedge_{sk\epsilon(T(s)- sj)} \neg g_k$

We show that $\varphi_i \wedge \psi_j$ and ξ_k are equivalent with respect to event constraints of q. Consider the component, $\wedge_{tk\epsilon ri} f_k$ and $\wedge_{sk\epsilon sj} g_k$ in $\varphi_i \wedge \psi_j$, and compare it with the component $\wedge_{vi\epsilon qk} h_i$ in ξ_k. $q_k = r_i \cup s_j$ and $h_i = f_i$ if $v_i \ \epsilon \ r_i - r_i \cap s_j$; $h_i = g_i$ if $v_i \ \epsilon \ s_j - r_i \cap s_j$; $h_i = f_i$ or $h_i = g_i$ if $v_i \ \epsilon \ r_i \cap s_j$. It follows that the two components have exactly the same terms for $v_i \ \epsilon \ r_i - r_i \cap s_j$ and for $v_i \ \epsilon \ s_j - r_i \cap s_j$. But for $v_i \ \epsilon \ r_i \cap s_j$, the first component has $f_i \wedge g_i$, while the second component only has h_i where $h_i = f_i$ or $h_i = g_i$. However, q also has the constraint $f_i \equiv g_i$ for $v_i \ \epsilon \ r_i \cap s_j$. It follows that $f_i \wedge g_i$ and h_i are equivalent for $v_i \ \epsilon \ r_i \cap s_j$. Hence, $\wedge_{tk\epsilon ri} f_k \wedge_{sk\epsilon sj} g_k$ in $\varphi_i \wedge \psi_j$, and $\wedge_{vi\epsilon qk} h_i$ in ξ_k are equivalent with respect to the event constraints of q.

A similar argument shows that the second components of the expressions (with $\neg h_i$, $\neg f_k$, and $\neg g_k$) are also equivalent. Hence, $\varphi_i \wedge \psi_j$ and ξ_k are equivalent with respect to event constraints of q.

4.4.1 Example

Consider $q_1 = \{t_1\}$ of the following epr-relation, where the set of all given tuples in the database be as shown in Table 8.

Table 24. epr-relation q

Tuples	Event Attribute
t_1	a
t_2	b

a ≡ c

$$\xi = a \wedge \neg b \rightarrow \ (1)$$

Applying the *Decomposition Algorithm* on q will produce the pr-relation (r,s) shown in Table 25. The event variable a is in pr-relation r and the event variables b and c are in pr-relation s.

Table 25. pr-relation r and s

S1

Tuples	Event Attribute
t_1	a

S2

Tuples	Event Attribute
t_1	c
t_2	b

The possible world $r_1 = \{t_1\}$ of r and possible world $s_1 = \{t_1\}$ of s combine to obtain the possible world $\{t_1\}$ in integration with EVF a and $c \wedge \neg b$ respectively.

The EVF for their integration is,

$$\varphi_i \wedge \psi_j = a \wedge \neg b \wedge c \rightarrow \ (2)$$

Equations (1) and (2) are equivalent with respect to the event constraint a ≡ c.

4.5 Theorem 6

Given an epr-relation q, let S be the set of all pairs (r,s) of two pr-relations whose integration results in q. All pairs (r,s) in S give exactly the same probabilities for the integrated possible worlds.

Proof: We saw in Theorem 4 in Section 4.3 that every pair (r,s) in S gives the same EVF for a possible world in integration. Hence the corresponding probabilities for the integrated possible worlds also match exactly. Consider the following example that demonstrates this.

4.5.1 Example

Consider the epr-relation q of Table 26. The event constraint of q has a variable d that does not appear in the tuples. The possible worlds of q are shown in Figure 2, and correspond to EVF $\neg a\wedge\neg b\wedge\neg c$, $\neg a\wedge\neg b\wedge c$, $\neg a\wedge b\wedge\neg c$, $\neg a\wedge b\wedge c$, $a\wedge\neg b\wedge\neg c$, $a\wedge\neg b\wedge c$, $a\wedge b\wedge\neg c$, and $a\wedge b\wedge c$ respectively. Given probabilities for event variables a, b, and c, we can calculate the probabilities of possible worlds. For example, probability of possible world $\{t_1, t_2\}$ associated with EVF $a\wedge b\wedge\neg c$ is P(a)P(b)(1-P(c)).

Table 26. epr-relation q

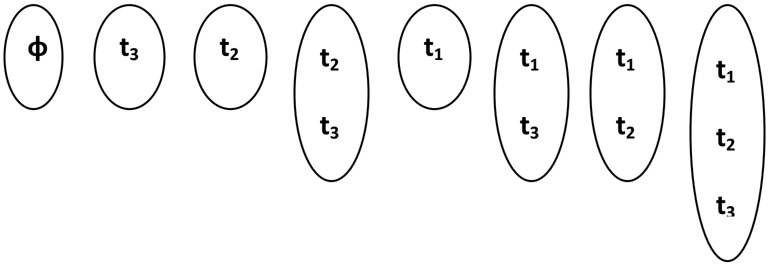
Tuples	Event Attribute
t_1	a
t_2	b
t_3	c

a ≡ d

Applying *Decomposition Algorithm* on q gives the following pair of pr-relations r and s of Table 27. Possible worlds of r are $r_1 = \{\phi\}$, $r_2 = \{t_2\}$, $r_3 = \{t_1\}$, and $r_4 = \{t_1, t_2\}$. The corresponding EVFs are $\neg a\wedge\neg b$, $\neg a\wedge b$, $a\wedge\neg b$, and $a\wedge b$, respectively. Similarly, possible worlds of s are $s_1 = \{\phi\}$, $s_2 =$

$\{t_3\}$, $s_3 = \{t_1\}$, and $s_4 = \{t_1,t_3\}$, and their corresponding EVFs are $\neg c\wedge\neg d$, $c\wedge\neg d$, $\neg c\wedge d$ and $c\wedge d$ respectively.

Figure 2. Possible worlds of q



Consistency graph of these possible worlds are shown in Figure 3. Edges connect pairs of compatible relations. For example, r_4 and s_4 can be integrated to obtain the possible world $\{t_1, t_2\}$ of q.

Table 27. pr-relations r and s

S1

Tuples	Event Attribute
t_1	a
t_2	b

S2

Tuples	Event Attribute
t_1	d
t_3	c

The EVF of r_4 U s_3 can be obtained from EVF of r_4 and s_3, namely, $(a\wedge b) \wedge (\neg c\wedge d) = a\wedge b\wedge\neg c\wedge d$. The EVF for this possible world obtained from q is $a\wedge b\wedge\neg c$. However they are equivalent due to the event constraint $a \equiv d$. The corresponding probability associated with $\{t_1, t_2\}$ is P(a)P(b)(1-P(c)).

q can also be obtained by integrating pr-relations r' and s' of Table 28. Possible worlds of r' are r_1'= $\{\phi\}$, and r_2'= $\{t_1\}$, with EVF $\neg a$ and a, respectively. Possible worlds of s' are s_1' = $\{\phi\}$, and s_2' = $\{t_1\}$, s_3' = $\{t_3\}$. s_4' = $\{t_1, t_3\}$, s_5' = $\{t_2\}$, s_6' = $\{t_1, t_2\}$, s_7' = $\{t_2, t_3\}$, and s_8' = $\{t_1, t_2, t_3\}$. Their

EVF are ¬b∧¬c∧¬d, ¬b∧¬c∧d, ¬b∧c∧¬d, ¬b∧c∧d, b∧¬c∧¬d, b∧¬c∧d, b∧c∧¬d, and b∧c∧d, respectively. r_1' is compatible with s_1', s_3', s_5'and s_7'. r_2' is compatible with s_2', s_4', s_6' and s_8'.

The possible world {t_1, t_2} of q is obtained as r_2' U s_6' in this case, with EVF a ∧ (b∧¬c∧d) = a∧b∧¬c∧d, which is exactly the same as the EVF obtained for r_4 U s_3 in the previous case, and is equivalent to a∧b∧c due to the event constraint a ≡ d. The corresponding probability associated with {t_1, t_2} is P(a)P(b)(1-P(c)). Thus the integration of pairs of compatible possible worlds yields exactly the same probabilities as from a different pair (r,s).

Figure 3. Consistency Graph

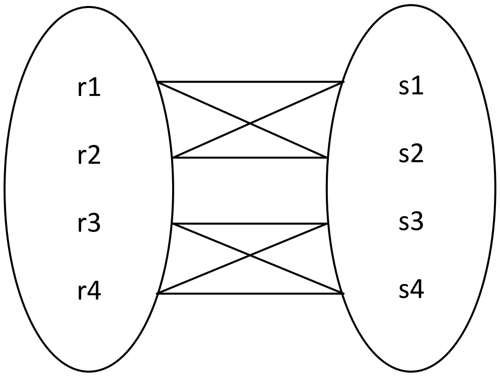


Table 28. pr-relations r' and s'

S1

Tuples	Event Attribute
t_1	a

S2

Tuples	Event Attribute
t_1	d
t_2	b
t_3	c

CHAPTER V

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this thesis we presented two methods to determine the probabilities of the possible worlds in integration and proved that they are equivalent. We introduced the Event Variable Formula (EVF) to build logical expressions corresponding to the pr- and epr-relations. We introduced the Decomposition Algorithm to determine the different ways in which the pr-relations in the source can be regenerated. We also verified that every decomposed pair of pr-relations is equivalent to any other pair and that all these pr-relations are equivalent to the epr-relation. Finally, due to the matching EVFs between the pr- and epr-relation, the probabilities were also found to match.

5.2 Future Work

With the field of uncertain data integration in its infancy, this work opens up exciting prospects for the future. Innumerable applications such as medical diagnosis based on data integrated across databases of observed symptoms and current medications, or sensor data processing systems stand to benefit directly from this work. We next outline the specific directions to further this research.

- This work provides new methods to determine the probabilities of data in integration and the theoretical background to support it. A practical implementation that can perform query processing based on the results derived from this work will be the next logical step.

- The current work limits itself to only two sources for integration. Considering more than two sources for integration opens up a new set of challenges that could benefit many more applications.

REFERENCES

[1] Sadri,F., "*On the foundations of probabilistic information integration*", CIKM '12: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, Oct 2012

[2] Borhanian,A.D., Sadri,F., *"A Compact Representation for Efficient Uncertain-Information Integration"*, IDEAS '13, Proceedings of the 17th International Database Engineering & Applications Symposium, pp.122-131

[3] Dalvi,N., Suciu,D., *"Efficient query evaluation on probabilistic databases"*, VLDB '04 Proceedings of the Thirtieth international conference on Very Large Data Bases, vol.30, pp.864-875

[4] Agrawal,P., "Incorporating uncertainty in data management and integration", Stanford University, 2012, http://ilpubs.stanford.edu:8090/1053/

[5] Hearst,A.M., "Information Integration", Trends and Controversies, IEEE Intelligent Systems, Sep/Oct 1998, http://www.cs.jyu.fi/ai/vagan/course_papers/Paper_12_III.pdf

[6] Haas,M.L, *"Beauty and the beast: The theory and practice of information integration"*, Proceedings of International Conference on Database Theory, pp.28–43, 2007

[7] Halevy,A.Y., Rajaraman,A., Ordille,J.J, "*Data integration: The teenage years"*, Proceedings of International Conference on Very Large Databases, pp.9–16, 2006

[8] Abiteboul,S., Kanellakis,C.P., Grahne,G., *"On the representation and querying of sets of possible worlds"*, Proceedings of ACM SIGMOD International Conference on Management of Data, pp.34-48, 1987

[9] Aggarwal,C.C., *"Managing and Mining Uncertain Data"*, IBM T.J.Watson Research Center, Hawthorne, NY 10532, Kluwer Academic Publishers, Boston/Dordrecht/London, http://charuaggarwal.net/utoc.pdf

[10] *"Scalable Query Processing in Probabilistic Databases"*, Department of Computer Science, University of Oxford, http://www.cs.ox.ac.uk/projects/SPROUT/

[11] Sen,P., *"Representing and Querying uncertain data",* Dissertation, Department of Computer Science, University of Maryland, College Park, 2009

[12] Agrawal,P., Sarma,A.D., Ullman,J., Widom,J., *"Foundations of Uncertain-Data Integration ",*Proceedings of the VLDB Endowment, Volume 3 Issue 1-2, pp.1080-1090, 2010

[13] Dalvi,N., Suciu,D., *"Efficient query evaluation on probabilistic databases"*, VLDB '07 Proceedings of the 30th VLDB Conference, Volume 30,pp.523-544