University Of Manouba

IHEC Carthage - ESC Manouba

MASTER DEGREE

IDIAG

Decisional Computer Science and Artificial Intelligence

EVIDENTIAL SKYLINE OPERATOR

SAYDA ELMI

Supervised by:

Pr Boutheina Ben Yaghlane	IHEC Carthage LARODEC Tunis	Tunisie
Pr Allel Hadjali	ENSMA poitiers, LIAS	France
Dr Mohamed Anis Bach Tobji	ISG, LARODEC Tunis	Tunisie

LARODEC Tunisie/LIAS France

Contents

1 Introduction **9**

1.1 Research Context 9

1.2 Problem 9

1.3 Contributions 10

1.4 Report Structure 10

2 Basic Concepts **11**

2.1 Introduction 11

2.2 Skyline Operator 11

2.2.1 Definition 11

2.2.2 Example 12

2.2.3 SQL Extensions 12

2.2.4 Skyline Algorithms 13

2.3 Evidence Theory 15

2.3.1 General Concepts 15

2.3.2 Extended definitions of belief function 17

2.4 Evidential Database 19

2.5 Conclusion 21

3 Skyline Operator in the Literature 22

3.1 Introduction 22

3.2 Skyline Operator on Certain Data 22

3.3 Skyline Operator on Uncertain Data 23

3.3.1 Skyline Operator On Probabilistic Data 23

3.3.2 Skyline Operator On Possibilistic Data 26

3.3.3 Skyline Operator On Stochastic Data 27

3.4 Skyline Operator on Web Service 28

3.5 Skyline Models Summary 30

3.6 Comparative Study 31

3.7 Conclusion 32

4 Skyline Operator On Uncertain Data Modeled by Evidential Theory 35

4.1 Introduction 35

4.2 Evidential Skyline using Comparison between Attributes 35

4.2.1 Evidential Dominance 35

4.2.2 Evidential Skyline 38

4.3 Evidential Skyline Using Possible Worlds 40

4.3.1 Conjunctive Rule of Combination 40

4.3.2 Line Body of an Evidential Object 40

4.3.3 Evidential Skyline using Possible Worlds 42

4.4 Conclusion 44

5 Computing the b-Dominant Skyline 45

5.1 Introduction 45

5.2 Naive Method 45

5.2.1 Comparison of attribute value 45

5.2.2 Computing the degree of belief to be in the skyline 46

5.2.3 Experiments and performance 46

5.3 Performed Method 48
5.3.1 Algorithm 48
5.3.2 Performance and scalability 51
5.3.3 Experimental Setup 52
5.3.4 Size of the Evidential Skyline 52
5.3.5 Performance and Scalability 53
5.4 Conclusion 54

6 Conclusion **56**

List of Figures

2.1 Skyline Of Hotels 13

2.2 BNL Algorithm: Step 1 14

2.3 BNL Algorithm: Step 2 15

2.4 BNL Algorithm: Step 3 15

3.1 A Set of Players 23

3.2 Certain Points 24

3.3 Uncertain Points 25

3.4 Fuzzy representation in terms of $x - y$ 29

5.1 Threshold Effect (b). 47

5.2 Evidential Dimensions Number Effects. 48

5.3 Size on the evidential skyline. 53

5.4 Size on the evidential skyline. 54

5.5 Elapsed time to compute the evidential skyline. 55

List of Tables

2.1 An extension of Hotels 12

2.2 Evidential database *EDB*. 20

3.1 The summary of notations. 30

3.2 Uncertain Skyline Models. 33

3.3 Comparative Table. 34

4.1 Evidential data example. 36

4.2 An extension of relation Hotels 37

4.3 Dominance belief degrees. 38

4.4 An EDB having three Attributes 40

5.1 Evidential data example. 49

5.2 Parameters and Examined Values. 52

List of Algorithms

1 Basic Belief Assignment Comparison BBAC 46

2 Evidential Skyline Computation ESC: Basic Algorithm BA..... 47

3 Two Phase Algorithm TPA..... 50

4 b−dominates (o_i, o_j, b)..... 51

Acknowledgements

After writing this report, i am convinced that the master is far from being a lonely task. Indeed, I would never have achieved this work without the support of many people whose generosity, good humor and interest shown towards my research enabled me to progress in this work.

This Research work was made possible thanks to the support and advice of many persons. I would like to thank everyone who contributed to this endeavour.

Sincere gratitude goes to Professor **Boutheina Ben Yaghlane**, who generously gave the time to participate in meetings and provide comments, advice and support.

I would like to thank Doctor **Mohamed Anis Bach Tobji** for sharing his expertise and time. He generously participated in development of this work.

I am also grateful to Professor **Allel Hadjali**, for his generosity, good humor and interest shown towards my research that enormously helped me to progress in this work.

Chapter 1

Introduction

1.1 Research Context

This summary presents the research work, as part of a research project of Master in the research laboratory LIAS ENSMA - Poitiers. The theme of Master consists on modeling and computing skyline operator for evidential databases.

1.2 Problem

Uncertainty is inherent in several applications such as environment surveillance, market analysis, sensor networks, data integration and decision-making, to name just a few. Due to the exploding number of information stored and shared over Internet, and the introduction of new technologies to capture and transit data, uncertain data analysis is an important issue in many applications. To deal with uncertain values of database attributes, several models were proposed. One of the most powerful approaches used to model uncertainty is the evidence theory. Databases that manage such type of data are called evidential databases.
The present decade has seen a revival of interest in preference queries which aim at determining the points/tuples/objects from a database which are not dominated by any other. A point dominates another point if it is as good or better in all dimensions and better in at least one dimension. To the best of our knowledge, there is no previous studies that introduced skyline queries on uncertain data where uncertainty is modeled by evidential theory.

1.3 Contributions

In the context of this research, we propose to extend evidential databases by an advanced analysis such as the skyline operator. A skyline operation aims at filtering out a set of interesting tuples, i.e., tuples that are not dominated by any other tuples. Thus, we tackle the above-mentioned challenges with the following major contributions.

- Given two objects of an evidential database, we calculate the belief that each object dominates the other. Based on this dominance relationship, we propose the notion of b-dominant skyline, which comprises the objects that are not dominated with some belief threshold b.

- We develop a suitable algorithm based on an efficient comparison method between two objects for computing efficiently the b-dominant skyline.

- We perform an extensive experimental evaluation to demonstrate the flexibility of the proposed evidential skyline and the scalability of our algorithm.

1.4 Report Structure

The rest of the report is organized as follows. In chapter 2, we propose, on the one hand, the notions of Pareto Order and Skyline preferences, and on the other hand, the basic notions of evidential theory and the evidential database model. Chapter 3 consists on a remainder of previous works about classic and uncertain skyline. In chapter 4, we propose two models used to deal with both skyline queries and evidential databases. Chapter 5 is devoted to the evidential skyline computation, a systematic performance study is reported there. Finally, chapter 6 recalls the main contributions and outlines some perspectives for future research.

Chapter 2

Basic Concepts

2.1 Introduction

In this chapter, it is supposed to present the main concepts of our reserch project. We first present the basic notions about skyline operator, then, a reminder about the evidence theory and the evidential database will be presented and detailed with examples.

2.2 Skyline Operator

2.2.1 Definition

The idea of Skyline queries has interested different communities. It is worth noticing that preferences can have relevance to many types of attributes in order to return a set of best possible items combining multiple preference criteria. Preferences can be expressed quantitatively by a scoring function [1], or qualitatively [7], [4] which is the more general approach. Many definitions have been proposed in the literature, but all of them are based on the use of Pareto order. Borzsonyi et al. [8] have introduced skyline preferences that aim at filtering out a set of interesting tuples from a large set of data. A tuple is interesting if it is not dominated by any other tuple in all dimensions. Let say that a tuple t_1 dominates another tuple t_2 if t_1 is at least as good as t_2 in all dimensions and better than t_1 in at least one dimension.

Definition 2.1. *Let $\{P_1, P_2, ..., P_n\}$ be a set of the atomic preferences. We denote by $t \succ_{P_i} t'$ (resp. $t \succeq_{P_i} t'$) the statement "tuple t satisfies preference P_i better than (resp. at least as good as) tuple t' ". Using Pareto order, a tuple t dominates another tuple t' iff*

$$\forall i \in \{1..n\}, t \succeq_{P_i} t' \ and \ \exists k \in \{1..n\}, t \succ_{P_k} t'$$

This operator returns the tuples that are not dominated by any other tuple. In other words, it returns the more preferred tuples regarding at least one preference. We propose an example which uses the language Preference SQL [8].

2.2.2 Example

Suppose you are going on holidays to Hawaii and you are looking for a hotel satisfying both preferences: minimum of price, and minimum of distance. Finding cheap hotel and close to beach are our goal but unfortunately, hotels near the beach are expensive, that is why we need an operator that return best tuples matching user preferences for any number of attributes.
Knowing that we have the following table schema : Hotels (id, name-Hotel, price, distance, city, stars).

Hotel	**name**	**price**	**distance**	**City**	**stars**
A_1	Algonquin	50	0.8	Hawaii	***
A_2	Hilton	100	1.0	Hawaii	****
A_3	Peninsula	52	0.44	Hawaii	***
A_4	Hilton	150	0.9	Hawaii	***

Table 2.1: An extension of Hotels

2.2.3 SQL Extensions

The language preference SQL is as follow:

Select * From Hotels **Where** City = 'Hawaii'
SKYLINE OF price MIN, distance MIN;

Skyline clause selects all interesting tuples, not dominated by other tuples. Criteria can be min/max/diff. Hotel A_1 (Price =50, distance =0.8mile) dominates Hotel A_2 (Price=100, distance=1.0mile) $\Rightarrow$ Hotel A_1 is in the Skyline. As a result, a such figure (figure 2.1) can

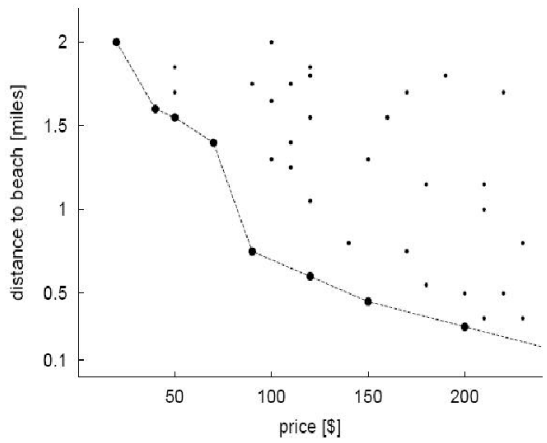


Figure 2.1: Skyline Of Hotels

graphically represent hotels being in the skyline.

Can we write an SQL query without using Skyline operator?

Answer: Yes, but cumbersome, expensive to evaluate, huge result set.

The following standard SQL query is equivalent to previous example but without using the Skyline operator:

SELECT *
FROM Hotels h
WHERE h.city = Hawaii **AND** NOT EXISTS(
SELECT *
FROM Hotels h1
WHERE h1.city = Hawaii
AND h1.distance $\leq$ h.distance
AND h1.price $\leq$ h.price
AND (h1.distance $<$ h.distance OR h1.price $<$ h.price));

2.2.4 Skyline Algorithms

In the literature, many algorithms were proposed to compute skyline from a database, we present some of these algorithms:

Divide and Conquer Algorithm

- The first step is to get a median value. Then, it divides tuples into 2 partitions P1 and P2.
- The second step is to compute the Skylines S1 and S2 of P1 and P2. This is done by recursively applying the whole algorithm to P1 and P2. The recursive partitioning

stops if a partition contains few tuples.

- Finally, it consists to merge S1 and S2 to compute the overall Skyline. This means eliminating all the tuples of S2 which are dominated by tuples of S1, i.e, no tuples of S1 can be dominated by tuples of S2 as tuples in S1 have a better d1 value.

BNL Algorithm

It is the Block Nested Loop Algorithm. BNL works particularly well if the Skyline is small. For example, if we have these information about a set of hotels;

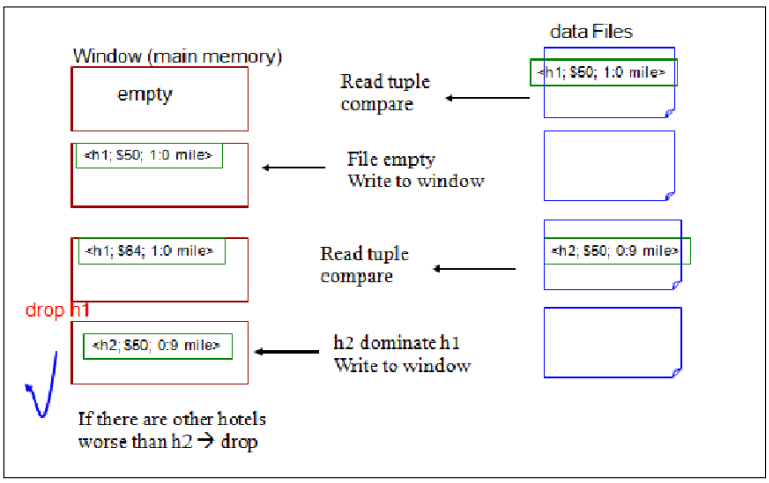


Figure 2.2: BNL Algorithm: Step 1

$$\langle h_1, 50\$, 1.0mile\rangle, \langle h_2, 50\$, 1.0mile\rangle$$
$$\langle h_3, 50\$, 1.0mile\rangle, \langle h_4, 50\$, 1.0mile\rangle$$
$$\langle h_5, 50\$, 1.0mile\rangle$$

Then, we can summarize its steps in the following figures:
Figure (2.2): Compare each tuple with another, dominating tuple/object is placed in a window in main memory.
Figure (2.3): Compare objects in the data files with selected objects in the window.
Figure (2.4): If it has no space, it writes in the temporary file, finally, the BNL algorithm compares objects in temporary file with objects in window.

Maximal Vector Computation in Large Data Sets

Find vectors that is not dominated by any of the vectors from the set and objects are represented by vectors (or points) in k-dimension space.

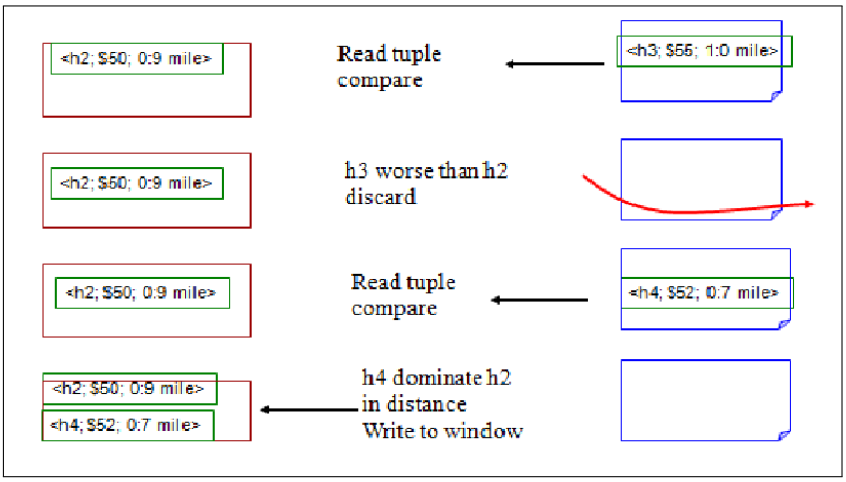


Figure 2.3: BNL Algorithm: Step 2

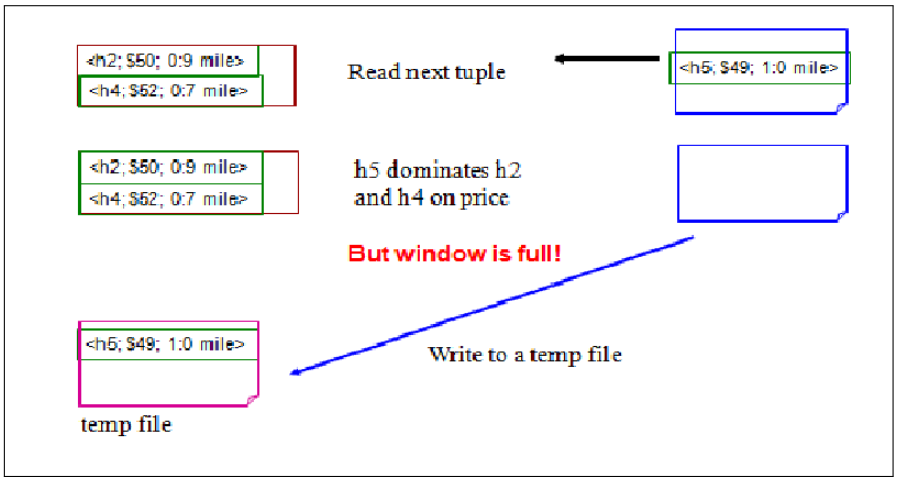


Figure 2.4: BNL Algorithm: Step 3

2.3 Evidence Theory

2.3.1 General Concepts

In order to access to the most accurate and reliable information, we need to represent and reason with uncertain, imprecise and incomplete information. In this context, Shafer [23] introduced in 1976 the mathematical theory of evidence which is a subjective evaluation using to characterize the truth of a proposition. The theory of evidence, also known

in the literature as the "Theory of belief functions" and "Theory of Dempster", is a generalization of the Bayesian theory of subjective probability [11]. The theory represents a set of propositional hypotheses by a frame of discernment.

Definition 2.2. *(Frames of Discernment). A frame of discernment, usually denoted as* Θ *where* $\Theta = \{h_1, ..., h_n\}$ *contains mutually exclusive and exhaustive propositional hypotheses, one and only one of which is true.*

Definition 2.3. *(Mass Function). A function, m:* $2^{\Theta} \rightarrow$*[0, 1], is called a basic probability assignment on a frame* Θ *if it satisfies the following two conditions:*

$$m(\phi) = 0 \qquad \textit{and} \qquad \sum_{A \subseteq \Theta} m(A) = 1. \tag{2.1}$$

where ϕ *is an empty set and* A *is any subset of* Θ*.*

The focal elements are the elements of A whose mass $m_{\Omega}(A)$ is not zero. Mass $m_{\Omega}(A)$ represents the part of degree of belief placed exactly on the A which can be different from a singleton hypothesis.

Example 2.1. *For the next presidential election, we wish to make a forecast on the winning candidate through one or more surveys. Let* Ω *be the frame of discernment where* $\Omega = \{G_1, G_2, D_1, D_2, D_3\}$*. Left Candidates are designed by:* $G_1 \cup G_2$ *and Right Candidates are designed by:* $D_1 \cup D_2 \cup D_3$*. A survey conducted in the street gives the following result :*

$$m(G_1) = 0{,}2 \quad m(G_2) = 0{,}05 \quad m(G_1 \cup G_2) = 0{,}1$$
$$m(D_1) = 0{,}3 \quad m(D_1 \cup D_2 \cup D_3) = 0{,}1$$
$$m(\Omega) = 0{,}25$$

Definition 2.4. *(Belief function). For every subset A of* Θ*, bel(A) is defined as the sum of the masses assigned to every subset B of A:*

$$bel(A) = \sum_{B \subseteq A} m(B) \tag{2.2}$$

Example 2.2. *Lets compute the belief function of some hypothesis applying equation (2.2):*

- *bel* $(G_1) = m(G_1) = 0.2$,
- *bel* $(G_1 \cup G_2) = m(G_1 \cup G_2) + m(G_1) + m(G_2) = 0.35$

Definition 2.5. *(Plausibility function). For every subset A of* Θ*, pl(A) is defined as the sum of the masses assigned to every subset B of* Θ *that intersects A:*

$$pl(A) = \sum_{B \cap A \neq \phi} m(B) \tag{2.3}$$

Example 2.3. *Lets now compute the plausibility function of the hypothesis:*

- $pl(G_1) = m(G_1) + m(G_1 \cup G_2) + m(\Omega) = 0,55.$
- $pl(G_2) = m(G_2) + m(G_1 \cup G_2) + m(\Omega) = 0,40.$
- $pl(D_1) = m(D_1) + m(D_1 \cup D_2 \cup D_3) + m(\Omega) = 0,65.$
- $pl(D_2) = m(D_1 \cup D_2 \cup D_3) + m(\Omega) = 0,35.$
- $pl(D_3) = m(D_1 \cup D_2 \cup D_3) + m(\Omega) = 0,35.$
- $pl(D_1 \cup D_2 \cup D_3) = 1 - bel(G_1 \cup G_2) = 0,65.$
- $pl(G_1 \cup G_2) = m(G_1) + m(G_2) + m(G_1 \cup G_2) + m(\Omega) = 0,6.$

As shown in this example, plausibility function measures the intensity with which we do not doubt the proposition D_1 , that is to say intensity with which D_1 is plausible. It is necessary to note that the plausibility function is the dual function of the belief function, it gives the maximum trend of a given hypothesis.

It is worth noticing that the evidential theory is very powerful in terms of resolving many and different problems dealing with uncertain data. Let us enumerate some works using evidential theory; Hong et al. [15] have used the evidential approach to query interface matching on the deep web. Add to that, evidential theory has also been used to extract frequent items so-called Data Mining [22].

2.3.2 Extended definitions of belief function

In probability theory, we can easily compute the probabilities of comparisons of two independent probability distributions. For example, we can work out the probability that one of the associated random variables is less than or equal to the other. But in standard D-S theory the definitions of the bel and pls functions do not handle comparisons like this. When we use mass functions to represent uncertain and imprecise information instead of probability distributions, it is necessary to extend the definition of bel and pls functions which can handle the comparison of two independent basic probability assignment.
Let m_X and m_Y be a focal elements, we consider x, y two random variables which are independent, and, $m_x, m_y : 2^\theta \rightarrow [0, 1]$, respectively.
Then

Definition 2.6. *(Equality bel(X=Y)) Let X, Y two random variables whose mass functions are* $m_x, m_y : 2^{\Theta} \longrightarrow [0,1]$*, respectively. Then, for* $A, B \subseteq \Theta$*,*

$$bel(X = Y) = \sum_{|A|=1} m_X(A) m_Y(A) = \sum_{a \in \Theta} m_X(\{a\}) m_Y(\{a\}). \tag{2.4}$$

Example 2.4. *(Equality bel(X=Y)) Suppose we have uncertain and imprecise information about the blood types for Jim and his wife Kim, we need to compute the proposition that their blood types are equal:*

- $bloodType_{Jim} = \langle \{A\}, 0.5 \rangle, \langle \{A, B\}, 0.5 \rangle$
- $bloodType_{Kim} = \langle \{A\}, 0.3 \rangle, \langle \{B\}, 0.3 \rangle, \langle \{A, B\}, 0.4 \rangle$
- $bel(X = Y) = 0.5 * 0.3 = 0.15$

Definition 2.7. *(Non Equality bel(X < Y)) Let X, Y two random variables whose mass functions are* $m_X, m_Y : 2^{\Theta} \longrightarrow [0,1]$*, respectively. Then, for* $A, B \subseteq \Theta$*,*

$$bel(X < Y) = \sum_{A \subseteq \Theta} (m_X(A) \sum_{B \subseteq \Theta, A <^{\forall} B} m_Y(B))), \tag{2.5}$$

Where $A <^{\forall} B$ *means that* $a < b$ *for all* $a \in A$*,* $b \in B$*.*

Definition 2.8. *(Non Equality bel(X ≤ Y)) Let X, Y two random variables whose mass functions are* $m_X, m_Y : 2^{\Theta} \longrightarrow [0,1]$*, respectively. Then, for* $A, B \subseteq \Theta$*,*

$$bel(X \leq Y) = \sum_{A \subseteq \Theta} (m_X(A) \sum_{B \subseteq \Theta, A \leq^{\forall} B} m_Y(B))). \tag{2.6}$$

Example 2.5. *(Non Equality bel(X < Y)) We have the following information about the cholesterol level ranges* $\{A, B\}$ *of Jim and Kim, where* $A > B$*. For two independent random variables X,Y whose mass functions are* $m_X, m_Y : 2^{\Theta} \longrightarrow [0,1]$*, respectively.*

- $CHO_{Jim} = \langle \{B\}, 0.3 \rangle, \langle \{A, B\}, 0.7 \rangle$
- $CHO_{Kim} = \langle \{A\}, 1.0 \rangle$

Let X and Y denote the random variables for the CHOs of Jim and Kim, respectively. Then, bel(X ≤ Y) = 1, but bel(X < Y) = 0.3 and bel(X = Y) = O.

This relation also applies to the other inequality comparison cases, and to the pls function as well as bel. With appropriate definitions, we can easily prove that the definitions of bel(X∇Y) and pls(X∇Y), where $\nabla \in (=, \neq, <, \leq)$, satisfy the following properties:

- $0 \leq$ bel(X ∇ Y) $\leq$ pls(X ∇ Y) ≤ 1;
- bel(X ∇ Y) + bel($\neg$(X ∇ Y)) ≤ 1;
- pls(X ∇ Y) + pls($\neg$ (X ∇ Y)) ≥ 1;
- If all the focal elements of m_X, m_Y are singletons, then bel(X ∇ Y) is Bayesian i.e., bel(X ∇ Y) = pls(X ∇ Y) = Pr(X ∇ Y).

General Properties

- $bel(\emptyset) = 0, \quad bel(\Omega) = 1.$
- $bel(A) + bel(\overline{A}) \leq 1.$
- $bel(A_1 \cup A_2 \cup ... \cup A_n) \geq \sum_i bel(A_i) - \sum_{i<j} bel(A_i \cap A_j) + ... + (-1)^{n+1} bel(A_1 \cap ... \cap A_n).$

2.4 Evidential Database

As said above, several real-world applications deal with imperfect data. The evidential database provides a solid framework to model and manage such type of information. Many publications [18], [26], [2] looked into modeling uncertainty in database systems. Evidential databases allow representing missing, uncertain or imprecise information thanks to the Dempster-Shafer theory of evidence. An evidential database, denoted by ***EDB***, is a database having n records and p attributes where each column has a domain Θ_i. Cell of line i ($1 \leq i \leq n$) and column j ($1 \leq j \leq p$) contains a normalized Basic probability Assignment called Mass function.

Definition 2.9. *For any attribute a_j of a data object t_i, let D_k denote the domain attribute maps into (i.e the frame of discernment Θ), and let m_{ij} represent the mass function for attribute a_j of data object t_j. We have then,*

$$t_i.a_j = \{\langle d, m_{ij}(d)\rangle | d \subseteq D_k, m_{ij}(d) > 0\}$$

$$m_{ij} : 2^{\Theta} \rightarrow [0,1] \; with \begin{cases} m_{ij}(\phi) = 0 \\ \sum\limits_{A \subseteq \Theta} m_{ij}(A) = 1 \end{cases} \tag{2.7}$$

Transaction	**Attribute A**	**Attribute B**
T_1	$\langle\{a_1\}, 0.25\rangle\langle\{a_3\}, 0.75\rangle$	$\langle\{b_1, b_2\}, 0.7\rangle$, $\langle b_2, 0.3\rangle$
T_2	$\langle a_2, 0.5\rangle$, $\langle\theta_A, 0.5\rangle$	$\langle\theta_B, 1\rangle$

Table 2.2: Evidential database *EDB*.

We obtain such database, see table 2.2, by collecting different experts opinions. First, experts are grouped into schools of thought, then opinions are aggregated using the cautious conjunction operator within groups and the non interactive disjunction across [13]. As shown in Table 2.2, in an evidential database, an item corresponds to a focal element. An itemset corresponds to a conjunction of focal elements. Table 1 is an example of an evidential database having two evidential attributes.

- The domain $\Omega_A = \{a_1, a_2, ..., a_n\}$ is the frame of discernment of the evidential attribute A,
- The domain $\Omega_B = \{b_1, b_2, ..., b_n\}$ is the frame of discernment of the second evidential attribute B.

For the first tuple T_1, we have:

- $m_B(b_1 \cup b_2) = 0.7$,
- $m_B(b_2)= 0.3$

That indicates we believe the attribute value is either b_2 with probability (mass function) 0.3 or one of the pair $\{b_1, b_2\}$ with probability (mass function) 0.7 satisfying the limit condition $\sum_{A\subseteq\Theta} m_{ij}(A) = 1$. We do not know how credible each element of the pair $\{b_1, b_2\}$ is, but we allocate the mass 0.7 to the set.
Note the kinds of imperfections that could be represented in evidential databases. For example, the value of the attribute A in the first tuple is probabilistic since it is a bba with singleton focal elements. This bba corresponds to a certain belief function.
The value of the attribute B in the first line could be seen as possibilistic with:

- $\Pi(b_1) = 0.7$
- $\Pi(b_2) = 1$.

This bba is about a consonant belief function where focal elements are nested. Such type of bba is converted to a possibility distribution if we consider that the possibility function Π in a possibility distribution meets the plausibility function pl in the evidence theory [23].
For the second tuple T_2, m_θ represents an appointment of belief to "ignorance":

- $m_A(\theta) = 0.5$ represents a partial ignorance of the A attribute value, and in the case of attribute B,
- $m_B(\theta) = 1$ represents the total ignorance of its value.

Moreover, perfect information and several other imperfections are modeled through the Dempster-Shafer theory and its special belief functions, well presented in [23]. Consequently, the operators developed on evidential databases are generic and so functional on probabilistic, possibilistic and naturally perfect databases.

2.5 Conclusion

In this chapter, we first presented the skyline operator, then, we detailed the concept of uncertain data modeled by Dempster-Shafer theory.
In the following chapter, we will be interested in the related works that were discussed in the literature.

Chapter 3

Skyline Operator in the Literature

3.1 Introduction

Skyline queries has been introduced in classic databases as well as uncertain databases such as probabilistic and possibilistic databases where data are represented by probabilistic and possibilistic distribution, respectively. In this chapter, we present some previous works of skyline operator that deal with both classic and uncertain databases.

3.2 Skyline Operator on Certain Data

The skyline of a set of d-dimensional points contains the points that are not dominated by any other point on all dimensions. Research on skyline in the context of certain data has been interested essentially in proposing a new computation methods. Skyline computation has recently received considerable attention in the database community.
For the classic database, where data are certain, there has been an increasing interest in algorithms that can quickly return the first skyline points without having to read the entire data file.
Starting with the pioneering work of Borzsonyi [8], who provided foundations for a pareto based preference model for database systems. Chomicki [10] focused on a preference algebra for an embedding of preference formulas into a relational setting.
Also, Skyline queries have been used for computing the skyline from web services based on quality of services (QoS) [4]. Many previous publications proposed algorithms for computing skylines, namely block nested table, divide and conquer [8], bitmap, index [25] and nearest neighbor [17]. Zadeh et al. introduced the concept of spatial skyline

queries (SSQ) in [24]. The best known skyline algorithm that can answer the SSQ is the Branch-and-Bound Skyline algorithm (BBS) [20].

3.3 Skyline Operator on Uncertain Data

The research towards the problem of multiple criteria optimisation over uncertain objects has a long history [28, 21, 27].

3.3.1 Skyline Operator On Probabilistic Data

Probabilistic Skyline Utility

Many previous studies showed that skyline analysis is very useful in multi−criteria decision making applications. We hava the following motivating example [21]: we consider analyzing players using the following multiple technical statistics criteria:

- The number of assists,
- The number of rebounds.

Let us use this criteria, both the larger the better, to examine the players and to find the perfect one who can achieve the best performance in all aspects. Unfortunately, such a player does not exist. The skyline analysis here is meaningful since it discloses the tradeoff between the merits of multiple aspects.
A player U is in the skyline if there exists no another player V such that V is better than U in one aspect, and is not worst than U in all other aspects. In figure 3.1, we plot a few

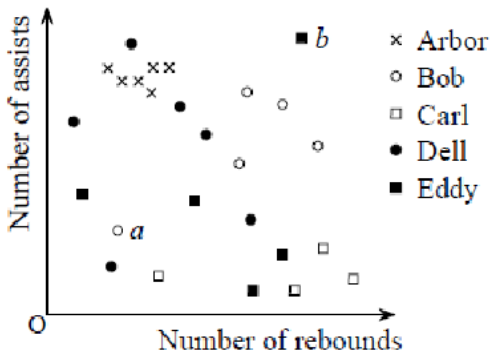


Figure 3.1: A Set of Players

games of 5 synthesis players to illustrate several important issues.
In this example, performances in different games may vary differently. In Figure 3.1, Arbor's performances are quite consistent while Eddy's performances are quite diverse. Although Eddy's performance in one game (point b) is better than Arbor's performances in all games in both criteria/ preferences. **Our goal is to decide which player is the best.**

Probabilistic Skyline Model

By default, we consider points in an n-dimensional numeric space $D = (D_1, ..., D_n)$. We assume that, on D, smaller values are more preferable.

Definition 3.1. *Let U and V be two uncertain objects, and f and f´ be the corresponding probability density functions, respectively. Let* $U = \{u_1, ..., u_{l_1}\}$ *and* $V = \{v_n, ..., v_{l_2}\}$ *be two uncertain objects and their instances. Then, the probability that V dominates U is:*

$$Pr[V \prec U] = \int_{u \in D} f(u)(\int_{v \prec u} f'(v)dv)du \tag{3.1}$$

$$Pr[V \prec U] = \frac{1}{l_1 l_2} \sum_{i=1}^{l_1} | \{v_j \in V \mid v_j \prec u_i\} \tag{3.2}$$

Example 3.1. *First, we want to show what is the difference between certain objects and uncertain objects.*

Figure (3.2) graphically represent certain objects. We can easily note that a certain

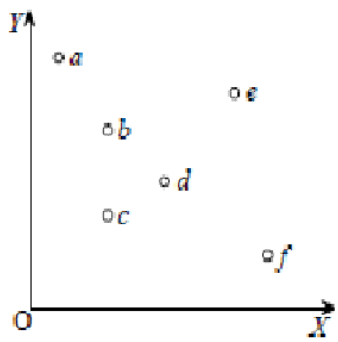


Figure 3.2: Certain Points

object is represented by only one point, however, in figure (3.3), an uncertain object can be represented by more than one point. Applying equation (3.2) n figure (3.3), we want to calculate the probability that A dominates C:
Pr(A $\prec$ *C)* $= \frac{1}{3}$ ** (C has 3 instances)* $\frac{1}{4}$ ** (A has 4 instances) **

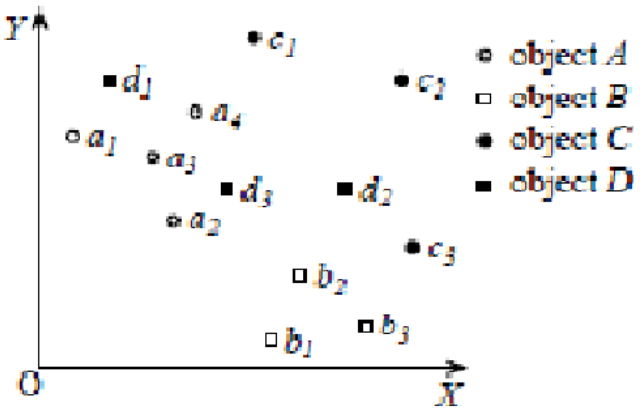


Figure 3.3: Uncertain Points

(4 (c_1 is dominated by every instance of A) + 4 (c_2 is dominated by every instance of A) + 0 (c_3 is not dominated by any instance of A))
$= \frac{2}{3}$

Definition 3.2. *(skyline probability of U) Let $U = \{u_1, ..., u_{l_1}\}$ be an uncertain object and its instances, the probability that U is in the skyline is:*

$$Pr(U) = \frac{1}{l} \sum_{i=1}^{l} \prod_{V \neq U} (1 - \frac{\mid \{v \in V \mid v \prec u_i \mid\}}{\mid V \mid} \tag{3.3}$$

Definition 3.3. *(skyline probability of instance $u \in U$) Let $U = \{u_1, ..., u_{l_1}\}$ be an uncertain object and its instances, Pr(u) is the probability that u is not dominated by any other objects, i.e., u is in the skyline is:*

$$Pr(u) = \prod_{V \neq U} (1 - \frac{\mid \{v \in V \mid v \prec u \mid\}}{\mid V \mid} \tag{3.4}$$

As an other definition of skyline probability of U, equation (3.3) can be written differently as:

$$Pr(U) = \frac{1}{l} \sum_{u \in U} Pr(u) \tag{3.5}$$

Proposed Algorithms

For uncertain data modeled by probabilistic distribution, Pei et al [21] tackled the problem of computing probabilistic skylines on large uncertain data sets and proposed two efficient algorithms:

- The bottom-up algorithm: computes the skyline probabilities of some selected instances of uncertain objects, and uses those instances to prune other instances and uncertain objects effectively.
- The top-down algorithm: recursively partitions the instances of uncertain objects into subsets, and prunes subsets and objects aggressively.

This idea has been ameliorated in [27] by adding the maybe confidence measure that indicates the existence probability of a given object.
Fung et al [12] proposed an efficient algorithm being able to compare objects that are skyline points using the probabilistic skyline model. Add to that, thanks to [12], we are able now to answer to the following question: Which of the objects are the K nearest neighbors to a given object according to their skyline probabilities?
Moreover, Jiang et al. [16] proposed a novel probabilistic skyline model where an uncertain object may take a probability to be in the skyline, and a p-skyline contains all objects whose skyline probabilities are at least p ($0 < p \leq 1$).
Also, a bounding-pruning-refining framework and three algorithms were proposed too.
In [28], Zhang et al. studied the problem of efficient processing of continuous skyline queries over sliding windows on uncertain data elements regarding given probability thresholds.

3.3.2 Skyline Operator On Possibilistic Data

In contrast with probability theory, when using possibility theory, we expect the following advantages:

- Possibility theory offers a qualitative model for uncertainty.
- In contrast with probability theory, the sum of the possibility degrees is different to 1.

For uncertain attribute values that are represented by possibility distribution, Bosc et al [6], proposed to compute the possibility that a given object/tuple is not dominated by any other object. In this framework, Skyline queries aim at computing the extent to which any object from a given relation is possibly/ certainly not dominated by any other object from that relation.
In addition, preferences can be represented in a possibilistic logic way using symbolic weights [14].

Definition 3.4. *Let $(A_1, ..., A_n)$ be the schema of the relation queried. Let $(Ar_1, ..., Ar_p)$ be the attributes concerned by a preference in the query and res the result of the query. We denote by $\succ_{Ar_k}$ the preference relation defined over the domain of attribute Ar_k.*
A given tuple t_i is dominated by another precise tuple t'_j, denoted by $t_i \prec t'_j$ iff:

$$\forall k \in \{1, ..., p\}, t'_j.Ar_k \succcurlyeq_{Ar_k} t_i.Ar_k \text{ and } \exists q \in \{1, ..., p\}, t'_j.Ar_q \succ_{Ar_q} t_i.Ar_q$$

Let $\Pi(t)$ denotes the degree of possibility that a tuple t from res be non-dominated by any other tuple t' from res. For each interpretation Π_i/t_i of t, we compute the possibility that for every tuple $t' \neq t$, there exists an interpretation t'_j of t' which does not dominate t_i. The final degree $\Pi(t)$ is the maximum of these degrees, computed over all the interpretations of t. This leads to:

$$\Pi(t) = max_{\Pi_i/t_i \in int(t)} \Pi(\Pi_i/t_i) \tag{3.6}$$

Where int(t) denotes the set of interpretations of t and

$$\Pi(\Pi_i/t_i) = min(\Pi_i, min_{t' \in res \setminus \{t\}} \Pi(t_i \not\prec t')) \tag{3.7}$$

To compute the possibility dominance between two given objects t and t', we should first compute the possibility degree that each interpretation of t is dominated/ not dominated by t'

$$\Pi(t_i \prec t') = \begin{cases} 0, if\{\Pi_j/t'_j \in int(t') \mid t_i \prec t'_j = \emptyset, \\ max_{\Pi/t'_j \in int(t') | t_i \prec t'_j} \Pi_j, otherwise \end{cases} \tag{3.8}$$

Example 3.2. *Let us consider a relation of schema (make, category), the preferences user are as follow:* $(VW \succ Ford \succ Opel)$ *and* $(SUV \succ roadster \succ others)$ *and the objects:*

- $t_1 = \langle \{1/Opel, 0.8/VW\}, roadster \rangle$
- $t_2 = \langle Ford, \{1/SUV, 0.7/VW\}, Sedan\} \rangle$
- $t_3 = \langle \{1/VW, 0.6/Opel\}, roadster \rangle$

For computing $\Pi(t_1)$*, we have to identify the interpretations of* t_1*:*

- $t_{11} = 1/\langle Opel, roadster \rangle$ *and* $t_{12} = 0.8/\langle VW, roadster \rangle$

Starting with t_{11} *we have:* $\Pi(t_1 1 \not\prec t_2) = 0.7$ *and* $\Pi(t_1 1 \not\prec t_3) = 0.6$ *Then, for* t_{12} *we have:* $\Pi(t_1 2 \not\prec t_2) = 1$ *and* $\Pi(t_1 2 \not\prec t_3) = 1$*, thus:*

$$\Pi(t_1) = max(min(1, min(0.7, 0.6)), min(0.8, min(1, 1))) = 0.8$$

3.3.3 Skyline Operator On Stochastic Data

In the light of utility of applications with uncertain data, the probabilistic skyline model is proposed to retrieve uncertain objects based on skyline probabilities. Nevertheless, skyline probabilities cannot capture the preferences of monotonic utility functions.
A novel skyline operator, namely stochastic skyline, provides the minimum set of candidates for the optimal solutions over all possible monotonic multiplicative utility functions [19].

Definition 3.5. *(Skyline) Given a class F (family of utility (scoring) functions where $F = \{\prod_{i=1}^{d} f_i(x_i)\}$, an uncertain object (random variable) U stochastically dominates V regarding F, denoted by $U \prec_F V$ if and only if*

$$E[f(U)] \geq E[f(V)] \text{ for each } f \in F$$

with f is a given utility function and E is the optimal solution.

Definition 3.6. *(Stochastic Dominance) Given two uncertain objects U and V, U stochastically dominates V, denoted by $U \prec_{sd} V$ if*

$$U.cdf(x) \geq V.cdf(x) \forall x \in \mathbb{R}^d + \text{ and } \exists y \in \mathbb{R}^d + \text{such that } U.cdf(y) > V.cdf(y)$$

The probability mass $U.cdf(x)$ of U is the sum of the probabilities of the instances in $\mathbb{R} = ((0, ..., 0), x)$ where $(0, 0, ..., 0)$ is the origin in $\mathbb{R}^d$.

Definition 3.7. *(Stochastic Skyline) Given a set of uncertain objects $\mathbf{D}$, an object $U \in \mathbf{D}$ is a stochastic skyline object if there is no object $V \in \mathbf{D}$ such that $V \prec_{sd} U$*

3.4 Skyline Operator on Web Service

Data Web services composition has emerged as a flexible solution to answer users' complex queries. User preferences are a key factor that can be used to rank candidate services/compositions and retain only the best ones.
In [3], the authors proposed a novel approach to compute the skyline from data service compositions based on user preferences. A model user preferences was introduced using fuzzy sets.
In [5], data services are modeled by fuzzy approach to be able t define, the fuzzy dominance between two points.

Definition 3.8. *(The fuzzy dominance) Given two d-dimensional points u and v, we define the fuzzy dominance to express the extent to which u dominates v as:*

$$deg(u \succ v) = \frac{\sum_{i=1}^{d} \mu \gg (u, v)}{d} \tag{3.9}$$

Where $\mu \gg$ is a membership function of the fuzzy relation $\gg$ that expresses the extent to which u is more or less greater than v. $\mu \gg$ can be defined as follows:

$$\mu \gg (x, y) = \begin{cases} 0, if x - y \leq \xi \\ 1, if x - y \geq \lambda + \xi \\ \frac{x-y-\xi}{\lambda} \quad otherwise \end{cases} \tag{3.10}$$

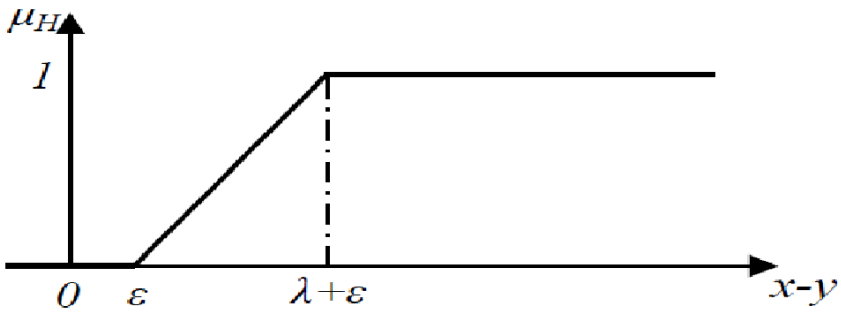


Figure 3.4: Fuzzy representation in terms of $x - y$

- if $x - y$ is less than λ then x is not at all strongly greater than y.
- if $x - y$ is larger than $\lambda + \xi$ then x is all much greater than y.
- if $x - y$ is between ξ and $\lambda + \xi$ then x is much greater than y to some extent.

3.5 Skyline Models Summary

In this section, we summarize with a table 3.2 which includes the main models already proposed in the literature that modeled the Skyline Operator in different types od data representing uncertainty. We can note the skyline from possibilistic data, the skyline from probabilistic data, and skyline from incomplete data.

Notation	**Definition**
U, V, P, Q	Uncertain objects
u, v	Instances of uncertain objects
l_1, l_2	The number of instances of U, V, respectively
$\mathbb{R}^d$	d-dimensional real space
f(x)	The continuous object
DB	The whole database
DB'	DB - f(x)
int(t)	interpretation of the tuple/object

Table 3.1: The summary of notations.

Table 3.2 presents in an easy way, the definitions of the different proposed Skyline models. Also, it details, for each uncertain model, the dominance definition between uncertain objects in the one hand, and the degree to be in the skyline in the other hand. The degree can be a probability, if data are modeled by the probabilistic distribution, or, a possibility degree if data are represented by the possibilistic distribution.
For reference, Table 3.1 is given as a summary of notations used in table 3.2.

3.6 Comparative Study

Table 3.3 presents a comparative study between the different Skyline proposed in the literature. Many publications proposed some extensions or criteria:

- **Top-k Skyline**: We denote by D the data set of uncertain objects. Let S be the result of the Skyline Query (i.e., the set of objects that are not dominated by any other objects). The top$-$k Skyline consists on ranking objects in S.
  Only Objects modeled by probabilistic distribution have been ranked.
- **Transitivity**: Given two uncertain objects x and y, $x \succ_P y$ denotes that x dominates y defined on a given preference P, the dominance relationship is transitive:

  - if $x \succ_P y$
  - and $y \succ_P z$
  - then $x \succ_P z$

  The dominance relationship is transitive in both data modeled by probabilistic distribution and possibilistic distribution.
- **Possible Worlds**: For the uncertain objects, with the extension of an object, many worlds can be drawn. This is called interpretations or extensions. Then, a dominance relationship between worlds of objects must be defined as done in [7] for possibilistic data.
- **Categorical Attribute**: There is two types of attributes: quantitative attributes and qualitative (categorical) ones. Only authors in [7], treated qualitative attribute in a possibilistic database.

To the best of our knowledge, as shown in table 3.3, there is no previous works that deals with Skyline from Eviential Database.

3.7 Conclusion

In this chapter, we presented the skyline operator from certain and uncertain data. Data can be modeled by different distributions such as probabilistic distribution and possibilistic one that was detailed in this chapter.
As evidential database (EDB) comes to generalize both probabilistic and possibilistic databases, we can say that EDB has a powerful mechanism to deal with database uncertainty.
That is why, we consider that modeling skyline from evidential data is really a challenging work.

Uncertain Models		Dominance between Objects	Degree to be in S
Uncertainty Modeled by	**Discrete Probability Model**	$Pr[V \prec U] =$ $\frac{1}{l_1 l_2} \sum_{i=1}^{l_1} \mid \{v_j \in V \mid v_j \prec u_i\}$	$Pr(U) =$ $\frac{1}{l} \sum_{u \in U} Pr(u)$
Probabilistic Distribution	**Continuous Probability Model**	$P(f_i(\overrightarrow{x_i}) \prec f_j(\overrightarrow{x_j})) =$ $\int \int f_i(\overrightarrow{x_i}).f_j(\overrightarrow{x_j}).$ $\begin{cases} 1, \overrightarrow{x_i} \prec \overrightarrow{x_j} \\ 0, \overrightarrow{x_i} \not\prec \overrightarrow{x_j} \end{cases} (d\overrightarrow{x_i} d\overrightarrow{x_j})$	$\mathrm{P}_S(f(x)) =$ $\int_{\mathbb{R}^d} f(x) \prod_{g(y) \in DB\prime}$ $\left(1 - \int_{\mathbb{R}^d} g(y). \begin{cases} 1, y \prec x \\ 0, otherwise \end{cases} dy\right) dx$
Incomplete Data		$P \succ Q$ if $\exists u_i \mid$ P.u_i and Qu_i are known and $P.u_i \succ Q.u_i$ and $\forall j \neq i$, $P.u_j$ and $Q.u_j$ are unknown, or $P.u_j \not\succ Q.u_j$	$S = \{P \mid \nexists\, Q/Q \succ P\}$
Possibilistic Distribution		$\Pi(t_i \prec t') =$ $\begin{cases} 0, if \\ \{\Pi_j/t'_j \in int(t') \mid t_i \prec t'_j = \emptyset, \\ max_{\Pi/t'_j \in int(t') \mid t_i \prec t'_j} \Pi_j, \\ otherwise \end{cases}$	$\Pi(t) =$ $max_{\Pi_i/t_i \in int(t)} \Pi(\Pi_i/t_i)$

Table 3.2: Uncertain Skyline Models.

Uncertain Skyline	Top-k Skyline	Transitivity	Possible Worlds	Categorical Attribute
Probabilistic Data	+	+	-	-
Possibilistic Data	-	+	+	+
Evidential Data	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 3.3: Comparative Table.

Chapter 4

Skyline Operator On Uncertain Data Modeled by Evidential Theory

4.1 Introduction

In this chapter, we tackle the problem of modeling skyline queries in the context of uncertain data modeled by the evidence theory. We represent first the definition of evidential skyline mainly based on the notion of dominance relationship namely evidential dominance whose purpose is the computation of the belief degree that an evidential object dominates another evidential object.
In this chapter, we present two proposed models; the first one was implemented in the context of the master degree, the second model is considered as a future work for our research team.

4.2 Evidential Skyline using Comparison between Attributes

4.2.1 Evidential Dominance

In this section, we extend the dominance relationship to evidential data, in order to be able to define the notion of evidential skyline presented in the following section.

Belief Basic Assignment Comparison

Consider in Table 4.1 a set of weight-loss products, defined over two attributes; weight loss per month and repayment (if the user is not satisfied). Each product may have one or more focal elements w.r.t. each attribute.
For example, the weight loss per month of product o_1 comprises two focal elements $\langle\{15, 16, 18\}, 0.1\rangle$ and $\langle\{19, 20\}, 0.9\rangle$. That is, we believe that the attribute value is either 15, 16 or 18 with mass function 0.1 or one of the values 19 or 20 with mass 0.9. However, we do not know how credible each single element is. To make the idea more clear, we use other examples throughout the rest of the paper.

Table 4.1: Evidential data example.

Product	Weight loss per month (kilograms)	Repayment (%)
o_1	$\langle\{15, 16, 18\}, 0.1\rangle$, $\langle\{19, 20\}, 0.9\rangle$	$\langle 90, 0.3\rangle$, $\langle\{90, 100\}, 0.7\rangle$
o_2	$\langle 7, 0.7\rangle$ $\langle\{8, 9\}, 0.3\rangle$	$\langle\{70, 80\}, 0.8\rangle$, $\langle 80, 0.3\rangle$
o_3	$\langle\{1, 4\}, 0.1\rangle$, $\langle 5, 0.9\rangle$	$\langle\{70, 80\}, 0.7\rangle$, $\langle 100, 0.3\rangle$
o_4	$\langle 10, 0.2\rangle$, $\langle 12, 0.2\rangle$, $\langle\{13, 14\}, 0.6\rangle$	$\langle 100, 1\rangle$
o_5	$\langle\{12, 13, 14\}, 0.2\rangle$, $\langle 17, 0.4\rangle$, $\langle 19, 0.4\rangle$	$\langle\{20, 30\}, 0.6\rangle$, $\langle 30, 0.4\rangle$

Let us start with the definition of the comparison between two belief basic assignment (bba) which is based on the following comparison model:
Given a set of objects $O = \{o_1, o_2, \ldots, o_n\}$ defined on a set of attributes $A = \{a_1, a_2, \ldots, a_d\}$, with $o_i.a_k$ denoting the belief basic assignment of object o_i w.r.t. attribute a_k; for example[1],
$o_1.wl = \{\langle\{15, 16, 18\}, 0.1\rangle\}, \langle\{19, 20\}, 0.9\rangle$ and $o_1.r = \{\langle 90, 0.5\rangle, \langle\{90, 100\}, 0.5\rangle\}$.
The belief that an object o_i is better than or equal to another object o_j w.r.t. an attribute a_k is given by [2]:

$$bel(o_i.a_k \geq o_j.a_k) = \sum_{A \subseteq Dom(a_k)} (m_{ik}(A) \sum_{B \subseteq Dom(a_k), A \geq^{\forall} B} m_{jk}(B)) \tag{4.1}$$

Where $A \geq^{\forall} B$ stands for $a \geq b$ for all $(a, b) \in A \times B$.
For instances, in our example, we have, $bel(o_1.wl \geq o_3.wl) = 0.3 \cdot (0.1 + 0.9) + 0.7 \cdot (0.1 + 0.9) = 1$, and $bel(o_1.r \geq o_3.r) = 0.7 \cdot 0.7 + 0.3 \cdot 0.7 = 0.7$.
Example
Suppose we have the following information about the price hotels of Regency and Hilton:

- $Price_{Hilton} = X_1 = \langle 50, 0.4\rangle, \langle\{50, 45\}, 0.6\rangle$,

[1]For short, we use wl and r to denote the weight loss per month and the repayment attributes, respectively

- $Price_{Regency} = X_2 = \langle 40, 0.75\rangle, \langle\{40, 45\}, 0.25\rangle$.
- $X_1 \prec_{Price} X_2$ means that hotel having Y as a basic belief assignment satisfies more the condition MIN PRICE.

We have then $bel(X_1 \prec_{Price} X_2) = 0.4 * (0.75 + 0.25) + 0.6 * 0.75 = 0.85$

Evidential Objects Dominance

Let us now extend the dominance relationship to evidential data. Given two objects o_i and o_j in $\mathcal{O} : o_i \neq o_j$, the belief that o_i dominates o_j is given by:

$$bel(o_i \succ o_j) = \prod_{a_k \in \mathcal{A}} bel(o_i.a_k \geq o_j.a_k) \tag{4.2}$$

For example, since $bel(o_1.wl \geq o_3.wl) = 1$ and $bel(o_1.r \geq o_3.r) = 0.7$, the belief that o_1 dominates o_3 is $bel(o_1 \succ o_3) = 1 \cdot 0.7 = 0.7$.

Example

T_i dominates T_j denoted by $T_i \succ T_j$ with a degree of belief equal to $bel(T_i \succ T_j)$.

n	**Name Hotel**	**Price**	**Stars Number**	**Country**
T_1	Marmara	$\langle 20/0.3\rangle$ $\langle\{25, 30\}/0.7\rangle$	$\langle 3/0.36\rangle$ $\langle\{3, 4\}/0.64\rangle$	Tahiti
T_2	Sheraton	$\langle 40/0.55\rangle$ $\langle\{25, 35\}/0.45\rangle$	$\langle 4/0.65\rangle$ $\langle\{3, 4\}/0.35\rangle$	Tahiti
T_3	Moorea Ia Ora	$\langle 55/0.7\rangle$ $\langle\{60, 65, 70\}/0.3\rangle$	$\langle 2/0.56\rangle$ $\langle\{4, 5\}/0.44\rangle$	Tahiti
T_4	Manava	$\langle\{15, 19\}/0.42\rangle$ $\langle 31/0.58\rangle$	$\langle 4/0.38\rangle\langle 2/0.32\rangle$ $\langle\{2, 3\}/0.3\rangle$	Tahiti
T_5	Taharuu	$\langle 30/0.18\rangle\langle 35/0.62\rangle$ $\langle\{36, 40\}/0.2\rangle$	$\langle 2/0.85\rangle$ $\langle\{2, 3\}/0.15\rangle$	Tahiti

Table 4.2: An extension of relation Hotels

$X_i \succ_{dim} X_j$ means that X_i dominates X_j. For instance, for the price attribute (respectively, stars number) $X_i \succ_{price} X_j = X_i <_{price} X_j$ (respectively, $X_i \succ_{StarsNumber} X_j = X_i >_{StarsNumber} X_j$).

Let table 4.2 be an extension of relation Hotels. Let us compute the degree of belief that T_i dominates T_j where i, j $\in$ [1..3] / i $\neq$ j, satisfying both preferences MIN Price and Max Stars Number. $X_i \succ_{Price} X_j$ denotes that X_i is preferred to X_j on price attribute and $Y_i \succ_{StarsNumber} Y_j$ denotes that Y_i is preferred to Y_j on Stars Number attribute. We have then:

- $bel(X_1 \succ_{Price} X_2) = 0,55 * 1 + 0.45 * 1 = 1$, and,
- $bel(Y_1 >_{StarsNumber} Y_2) = 0$,
- $bel(T_1 \succ T_2) = 1 * 0 = 0$.
  Now we compute the degree of belief that T_1 is dominated by T_2:
- $bel(X_2 \succ_{Price} X_1) = 0$,
- $bel(Y_2 \succ_{StarsNumber} Y_1) = 0.36 * 1 + 0.64 * 0.65 = 0.776$,
- $bel(T_2 \succ T_1) = 0 * 0.776 = 0$.
  Let us compute the degree of belief that T_1 dominates T_3:
- $bel(X_1 \succ_{Price} X_3) = 1$,
- $bel(Y_1 \succ_{StarsNumber} Y_3) = 0.56 * (0.36 + 0.64) + 0.64 * 0 = 0.56$,
- $bel(T_1 \succ T_3) = 1 * 0.56 = 0.56$.

As shown in this example, to obtain the belief degree that an object T_i dominates an other object T_i, it is necessary to compute the belief degree that the basic belief assignment (bba) of T_i is upper than the T_i bba according to the same dimension/ preference.
Now, we have a detailed idea for how can we compute the dominance relationship between evidential tuples/ objects.
Let Table 4.3 be the result of the Dominance belief degrees computation of our principle example in Table 4.1, it shows the dominance belief that each object in lines dominates another object in columns.
Observe this definition (equation 4.2) reduces to the dominance relationship on certain

Objects	o_1	o_2	o_3	o_4	o_5
o_1	–	1	0.7	0.3	0.92
o_2	0	–	0	0	0
o_3	0	0	–	0	0
o_4	0	1	1	–	0
o_5	0	0	0	0	–

Table 4.3: Dominance belief degrees.

objects, i.e., the belief is 1 if $o_i \succ o_j$ and 0 otherwise. This is because the condition $\geq$ is sufficient to an object o_i do dominate another object $o_j \neq o_i$.

4.2.2 Evidential Skyline

b-Dominance

Based on the evidential dominance relationship, we define the notion of b-dominance as follows.

Definition 4.1. *(b-dominance)*
Given two objects $o_i, o_j \in \mathcal{O}$ and a belief threshold b, o_i b-dominates o_j, denoted by $o_i \succ_b o_j$ if and only if $bel(o_i \succ o_j) \geq b$.

For instance, o_1 0.9-dominates both o_2 and o_4. However, it does not 0.9-dominate o_4 since $bel(o_1 \succ o_4) = 0.3 < 0.9$.

b-Dominance Skyline

We can now use this b-dominance relationship to define the notion of evidential skyline. Intuitively, an object is in the evidential skyline if it is not dominated with some threshold. Thus, we define the notion of b-dominant skyline as follows.

Definition 4.2. *(b-dominant skyline)*
The skyline of O, denoted by b-Sky_O, comprise those objects in O that are not b-dominated by any other object, i.e.,

$$b - Sky_O = \{o_i \in O \mid \nexists\, o_j \in O, o_j \succ_b o_i\}. \tag{4.3}$$

For example, the 0.4-dominant skyline comprises objects o_1 and o_4, since they are not 0.4-dominated by any other object, while the 0.2 contains only o_1 as o_4 is 0.2 dominated by o_1. From, this observation, we illustrate a key property of the b-dominant skyline.

Theorem 4.1. *Given two belief thresholds b and b', if b < b' then the b-dominant skyline is a subset of the b'-dominant skyline, i.e., $b < b' \Rightarrow b - Sky_O \subseteq b' - Sky_O$.*

Proof. Assume that there exists an object o_i such that $o_i \in b\text{-}Sky_O$ and $o_i \notin b'\text{-}Sky_O$. Since $o_i \notin b'\text{-}Sky_O$, there must exists another object, say o_j, that b-dominates o_i. Thus, $bel(o_j \succ o_i) > b'$. But, $b < b'$. Therefore, $bel(o_j \succ o_i) > b$. Hence, $o_j \succ_b o_i$, which leads to a contradiction as $o_i \in b\text{-}Sky_O$. □

Theorem 4.1 indicates that the size of the b-dominant skyline is smaller than that of the b'-dominant skyline if $b < b'$. Roughly speaking, from Theorem 4.1, we can see that users have the flexibility to control the size of the retrieved evidential skyline objects by varying the belief threshold.

As mentioned above, we defined the model allowing to conclude if a given object is in the skyline with a given threshold introduced by the user or not, knowing that we manipulate uncertain data modeled by a basic belief assignment.

After publishing the model that has already been presented, we worked on another model that will be the subject of another publication. You will find in the following sections, the model based on the explosion of possible worlds using the Conjunctive Rule of Combination related to the evidence theory.

4.3 Evidential Skyline Using Possible Worlds

4.3.1 Conjunctive Rule of Combination

The Conjunctive Rule of Combination (CRC) allows us to interpret the possible worlds of a basic belief assignment (bba), and to assign a mass function for every interpretation. In this section, we will detail the CRC utility with an example.

4.3.2 Line Body of an Evidential Object

Let $\mathcal{A} = \{A_1, A_2, ..., A_n\}$ be a set of attributes. We denote by Ω the frame of discernment of an object extensions where $\Omega = \Omega_{A_1} \cap \Omega_{A_2} \cap ... \cap \Omega_{A_n}$, and o_i denote the possible worlds of $\mathcal{O}$ with $\sum m(o_i) = 1$.
A line body of an evidential object is a set of interpretations or extensions o_i of a bba that has Ω as a frame of discernment where $\sum m(o_i) = 1$.
Let the following table be an example of an Evidential Database that contains five objects and three attributes, A, B and C, applying the conjunctive rule of combination, we interpret the line body of every object:

Object	**Attribute A**	**Attribute B**	**Attribute C**
O_1	$\langle\{50, 60\}, 0.7\rangle$, $\langle\{65\}, 0.3\rangle$	$\langle\{1, 2, 3\}, 0.9\rangle$, $\langle\{2\}, 0.1\rangle$	$\langle\{A, B\}, 0.9\rangle$, $\langle\{B\}, 0.1\rangle$
O_2	$\langle\{70, 80\}, 0.4\rangle$, $\langle\{50, 55\}, 0.6\rangle$	$\langle\{4, 5\}, 0.2\rangle$, $\langle\{5\}, 0.5\rangle, \langle\{6\}, 0.3\rangle$	$\langle\{A, C\}, 0.5\rangle$, $\langle\{C\}, 0.5\rangle$
O_3	$\langle\{60, 65\}, 0.35\rangle$, $\langle\{70\}, 0.65\rangle$	$\langle\{7\}, 0.8\rangle$, $\langle\{5, 6\}, 0.2\rangle$	$\langle\{B, C\}, 0.1\rangle$, $\langle\{A\}, 0.9\rangle$
O_4	$\langle\{80, 90\}, 0.5\rangle$, $\langle\{100\}, 0.5\rangle$	$\langle\{8, 9\}, 1\rangle$	$\langle\{A, B\}, 0.6\rangle$, $\langle\{B\}, 0.4\rangle$
O_5	$\langle\{100\}, 1\rangle$	$\langle\{5, 6\}, 0.8\rangle$, $\langle\{3, 4\}, 0.2\rangle$	$\langle\{A\}, 1\rangle$

Table 4.4: An EDB having three Attributes

Line Body of O_1

- $o_1 = \langle\{50, 60\}, \{1, 2, 3\}, \{A, B\}, 0.7 * 0.9 * 0.9 = 0.567\rangle$
- $o_2 = \langle\{50, 60\}, \{1, 2, 3\}, \{B\}, 0.7 * 0.9 * 0.1 = 0.063\rangle$
- $o_3 = \langle\{50, 60\}, \{2\}, \{A, B\}, 0.7 * 0.1 * 0.9 = 0.063\rangle$
- $o_4 = \langle\{50, 60\}, \{2\}, \{B\}, 0.7 * 0.1 * 0.1 = 0.007\rangle$
- $o_5 = \langle\{65\}, \{1, 2, 3\}, \{A, B\}, 0.3 * 0.9 * 0.9 = 0.243\rangle$
- $o_6 = \langle\{65\}, \{1, 2, 3\}, \{B\}, 0.3 * 0.9 * 0.1 = 0.027\rangle$
- $o_7 = \langle\{65\}, \{2\}, \{A, B\}, 0.3 * 0.1 * 0.9 = 0.027\rangle$
- $o_8 = \langle\{65\}, \{2\}, \{B\}, 0.3 * 0.1 * 0.1 = 0.003\rangle$

Line Body of O_2

- $o_1 = \langle\{70, 80\}, \{4, 5\}, \{A, C\}, 0.04\rangle$
- $o_2 = \langle\{70, 80\}, \{4, 5\}, \{C\}, 0.04\rangle$
- $o_3 = \langle\{70, 80\}, \{5\}, \{A, C\}, 0.1\rangle$
- $o_4 = \langle\{70, 80\}, \{5\}, \{C\}, 0.1\rangle$
- $o_5 = \langle\{70, 80\}, \{6\}, \{A, C\}, 0.06\rangle$
- $o_6 = \langle\{70, 80\}, \{6\}, \{C\}, 0.06\rangle$
- $o_7 = \langle\{50, 55\}, \{4, 5\}, \{A, C\}, 0.06\rangle$
- $o_8 = \langle\{50, 55\}, \{4, 5\}, \{C\}, 0.06\rangle$
- $o_9 = \langle\{50, 55\}, \{5\}, \{A, C\}, 0.15\rangle$
- $o_{10} = \langle\{50, 55\}, \{5\}, \{C\}, 0.15\rangle$
- $o_{11} = \langle\{50, 55\}, \{6\}, \{A, C\}, 0.09\rangle$
- $o_{12} = \langle\{50, 55\}, \{6\}, \{C\}, 0.09\rangle$

Line Body of O_3

- $o_1 = \langle\{60, 65\}, \{7\}, \{B, C\}, 0.028\rangle$
- $o_2 = \langle\{60, 65\}, \{7\}, \{A\}, 0.252\rangle$
- $o_3 = \langle\{60, 65\}, \{5, 6\}, \{B, C\}, 0.007\rangle$
- $o_4 = \langle\{60, 65\}, \{5, 6\}, \{A\}, 0.063\rangle$
- $o_5 = \langle\{70\}, \{7\}, \{B, C\}, 0.052\rangle$
- $o_6 = \langle\{70\}, \{7\}, \{A\}, 0.468\rangle$
- $o_7 = \langle\{70\}, \{5, 6\}, \{B, C\}, 0.013\rangle$
- $o_8 = \langle\{70\}, \{5, 6\}, \{A\}, 0.117\rangle$

Line Body of O_4

- $o_1 = \langle\{80, 90\}, \{8, 9\}, \{A, B\}, 0.3\rangle$
- $o_2 = \langle\{80, 90\}, \{8, 9\}, \{B\}, 0.2\rangle$
- $o_3 = \langle\{100\}, \{8, 9\}, \{A, B\}, 0.3\rangle$
- $o_4 = \langle\{100\}, \{8, 9\}, \{B\}, 0.2\rangle$

Line Body of O_5

- $o_1 = \langle\{100\}, \{5, 6\}, \{A\}, 0.8\rangle$
- $o_2 = \langle\{100\}, \{3, 4\}, \{A\}, 0.2\rangle$

4.3.3 Evidential Skyline using Possible Worlds

Skyline Belief of an instance u of an object U

Let U and V be two evidential objects where $U \neq V$, and u be an **instance/ extension/ possible world/ interpretation** of an uncertain object U modelled by a bba.
Let $bel(u)$ be the belief that u is not dominated by any other object V, i.e, u is in the Skyline.

$$\forall u \in U, \qquad bel(u) = \frac{1}{|V|} m_u(A). \sum_{\substack{v \in V, \forall V \neq U, \\ A, B \in \Omega, B \leq A}} m_v(B) \tag{4.4}$$

Skyline Belief of an object U

$bel(U)$ is the belief that U is not dominated by any other object V, i.e, U is in the Skyline.

$$bel(U) = \frac{1}{|V|} \sum_{i=1}^{|U|} m_u(A) \sum_{\substack{v \in V, \forall V \neq U, \\ A, B \in \Omega, B \leq A}} m_v(B) \tag{4.5}$$

$$bel(U) = \sum_{i=1}^{|U|} bel(u_i) \tag{4.6}$$

Example 4.1. *Let's compute the belief that the object O_4 be in the skyline*

- $bel(o_{4_1}) = \frac{1}{4} * 0.3 * (\underbrace{1} + \underbrace{1} + \underbrace{0,028 + 0,007 + 0,052 + 0,013}) = 0,157$

- $bel(o_{4_2}) = \frac{1}{4} * 0.2 * (\underbrace{0,063 + 0,007 + 0,027 + 0,003} +$
  $\underbrace{0,04 + 0,1 + 0,06 + 0,06 + 0,15 + 0,09} +$
  $\underbrace{0,028 + 0,007 + 0,052 + 0,013}) = 0,035$

Let's compute the belief that the O_4 instances be in the skyline:

- $bel(o_{4_3}) = \frac{1}{4} * 0.3 * (\underbrace{1} + \underbrace{1} + \underbrace{0,028 + 0,007 + 0,052 + 0,013} + \underbrace{1}) = 0,232$

- $bel(o_{4_4}) = \frac{1}{4} * 0.2 * (\underbrace{0,063 + 0,007 + 0,027 + 0,003}$
  $+ \underbrace{0,04 + 0,1 + 0,06 + 0,06 + 0,15 + 0,09} +$
  $\underbrace{0,028 + 0,007 + 0,052 + 0,013}) = 0,035$

The belief that O_4 be in the skyline: bel(O_4) = 0,157 + 0,035 + 0,232 + 0,035 = 0,459
Belief Degrees of the other objects: in the same way, we compute the belief degrees of the rest of objects:

- $bel(O_1) = 0,$
- $bel(O_2) = 0,$
- $bel(O_3) = 0.295,$
- $bel(O_4) = 0.459,$
- $bel(O_5) = 0.430.$

As a result, we can rank the objects presented in the example according to the belief skyline computation: $O_4 \succ O_5 \succ O_3 \succ O_1 \succeq O_2$
Intuitively, an object is in the evidential skyline if it is not dominated with some threshold b. Then, we define the notion of *b*- skyline as follows.

Definition 4.3. *(b-skyline)*
Let D be the evidential database. The skyline of D, denoted by b-Sky_D, comprise those objects in D that have a belief skyline greater than the threshold b:

$$Sky(b) = \{\ U \in D \mid \quad bel(U) \geq b\} \tag{4.7}$$

Example 4.2. *(The b−Skyline)*

- *If* $b = 0.2, sky(0.2) = \{O_3, O_4, O_4\}$
- *If* $b = 0.3, sky(0.3) = \{O_4, O_5\}$

4.4 Conclusion

As mentioned above, we defined two models allowing to conclude if a given object is in the skyline with a given threshold introduced by the user or not.
The first model was published and accepted, we worked on the second model implementation as a future work. In the next chapter, we will present the main details of the first model implementation.

Chapter 5

Computing the b-Dominant Skyline

5.1 Introduction

This chapter mainly contains two parts, the first part consist on detailing the naive method for the skyline computation, we present here some experiments. After that, we will introduce the performed method and fortunately, our experimental evaluation demonstrated the scalability of our algorithm.

5.2 Naive Method

In this section, we first describe a general algorithm for computing the degree of belief that a tuple/object from an evidential database, be in the skyline. We then describe the algorithms that allow to compare evidential attribute values knowing that the number of dimensions to be compared depends on user preferences number. We formally state the pseudo code of the approach, denoted as Skyline Computation for Evidential data shown in naive algorithm 1. We develop some critical implementation details here. Let O denotes the evidential database EDB and A the set of attributes/preferences.

5.2.1 Comparison of attribute value

Let O represent the evidential database. Comparison function is applied only to the attributes which are related to user preferences. We first compare those attribute value of

two given objects/tuples o_i and $o_j, \forall o_i,\ o_j \in O$. Each attribute is treated differently, it depends on the preference nature (i,e MAX, MIN), let $o_i.a_k$ denote the basic belief assignment for object o_i defined on the attribute/dimension a_k. A BBA is composed of a set of focal elements. A focal element is composed by a set of hypothesis in Θ (the frame of discernment) and a mass function attributed for the hypothesis.
As shown in naive algorithm BBAC, we scan all focal elements for every BBA, $FocalElement_{o_j} \succ_{a_k} FocalElement_{o_i}$ denotes that an hypothesis from o_j is preferred to an hypothesis from o_i.
The evidential attribute value comparison algorithm is an implementation of the mathematical definition mentioned in equation (4.1) to compare two basic belief assignment (BBA) proposed by Bell [2]. BBAC algorithm returns the degree of belief that the bba $o_i.a_k$ is preferred to another bba $o_j.a_k$.

Algorithm 1 Basic Belief Assignment Comparison BBAC $(o_i, o_j \in O)$

```
som ⟵ 0, bel ⟵ 0, BEL ⟵ 0
foreach FocalElement_{o_i} in o_i.a_k do
    foreach FocalElement_{o_j} in o_j.a_k do
        if FocalElement_{o_j} ≻_{a_k} FocalElement_{o_j} then
            som ⟵ som + FocalElement_{o_j}.mass;
    bel ⟵ som * FocalElement_{o_i}.mass
    BEL ⟵ BEL + bel
return BEL
```

5.2.2 Computing the degree of belief to be in the skyline

To decide if an object o_i in O is in the skyline or not, we need to compute the belief degree that o_i is not dominated by any other object o_j in O. As shown in ESC algorithm, for an object o_i, the algorithm scans the evidential database to conclude if there exist an other object o_j that dominates o_i respecting a given threshold introduced by the user.
If bel $(o_i \geq o_j) <$ b then $o_i \nexists$ Ev−Sky, i.e, the evidential skyline. Obviously, the data complexity is in O(n^2) where n denotes the size of the EDB (number of objects in O).

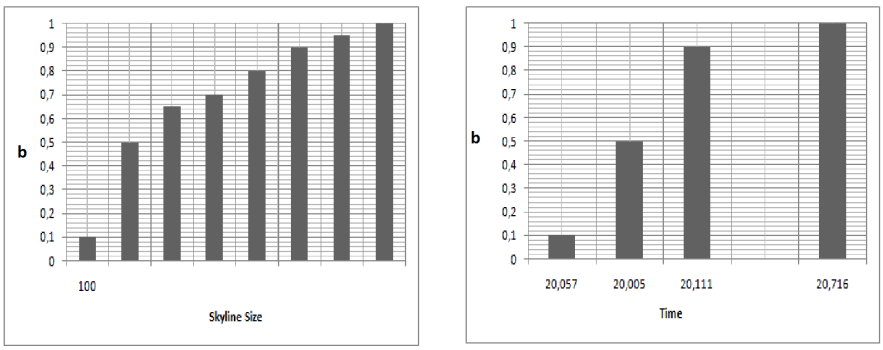
5.2.3 Experiments and performance

In this step, we began with introducing a real values of objects, this experiment is performed on a limited number of objects due to the difficulty of the task, but since we need to evaluate more objects, a synthetic database will be useful to evaluate more objects. That will be presented in the next section.
For the real object values, Fig.5.1a shows the size variation (i.e., the number of objects returned) of the skyline for different values of the threshold b.

Algorithm 2 Evidential Skyline Computation ESC: Basic Algorithm BA

Input: Objects $\mathcal{O}$; belief threshold b;
Output: Evidential skyline *Ev-Sky*;

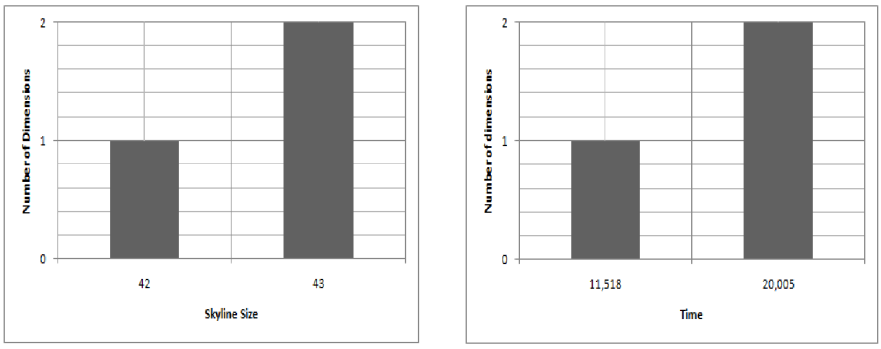
```
foreach Oi ∈ O do
    Sky ⟵ False , Ev-Sky ⟵ ∅

    foreach Oj ∈ O, Oj ≠ Oi do
        bel ⟵ 1;

        foreach a_k ∈ A do
            bel ⟵ bel * BBAC (Oi, Oj);

        if bel ≥ b then
            Sky ⟵ True else exit;
    if Sky = True then
        Ev-Sky ⟵ Oi;
return Ev-Sky
```



(a) Skyline Size. (b) Time.

Figure 5.1: Threshold Effect (b).

Fig.5.1b shows that the size on the evidential skyline increases with higher b since when b increases more objects have chances not to be dominated.
Fig.5.2 shows that the size on the evidential skyline and the elapsed time increase with higher number of attributes/preferences/dimensions d since when d increases more objects have chances not to be dominated.
As shown, the naive algorithm results in a high computational cost. Thus, it is crucial to optimize the dominance checks to improve the performance of the algorithm. That is why, a performed method has been necessary to minimize the number of dominance checks and



(a) Skyline Size. (b) Time.

Figure 5.2: Evidential Dimensions Number Effects.

optimize our algorithm.

5.3 Performed Method

In this section, we first propose an appropriate algorithm to compute the evidential skyline, minimizing the dominance checks.
We then devise an efficient method for optimizing the dominance checks.

5.3.1 Algorithm

A straightforward algorithm to compute the b-dominant skyline is to compare each object o_i against the other objects.
If o_i is not b-dominated, then it belongs to the evidential skyline. However, this approach results in a high computational cost (see Experimental Evaluation) as it needs to compare each object with every others.
Also, consider the objects depicted in Table 5.1. We have:

- $bel(o_x \succ o_y) = 0.3$,
- $bel(o_y \succ o_z) = 0.4$,
- $bel(o_x \succ o_z) = 0.0245$.

Observe that o_x 0.3-dominates o_y and o_y 0.3-dominates o_z, but o_x does not 0.3-dominate o_z. Thus, **the b-dominance relationship is not transitive.** Therefore, an object

cannot be eliminated even if it is b-dominated since it will be useful for eliminating other objects.

Table 5.1: Evidential data example.

Product	Weight loss per month (kilograms)	Repayment (%)
o_x	$\langle\{10, 11\}, 0.5\rangle$, $\langle\{19, 20\}, 0.5\rangle$	$\langle 70, 0.3\rangle$, $\langle\{80, 90\}, 0.7\rangle$
o_y	$\langle\{17, 18\}, 1\rangle$	$\langle\{60, 70\}, 0.6\rangle$, $\langle 100, 0.4\rangle$
o_z	$\langle\{9, 12\}, 0.7\rangle$, $\langle\{15, 16\}, 0.3\rangle$	$\langle 80, 0.1\rangle$, $\langle\{90, 100\}, 0.9\rangle$

From this property, we propose a two phase algorithm (see Algorithm 3) that follows the principle of the two scan algorithm [9].

TPA, computes the evidential skyline through two phases:
In the first phase (lines 2–13), a set of candidate objects $b\text{-}Sky_{\mathcal{O}}$ is selected by comparing each object o_i in $\mathcal{O}$ with those selected in $b\text{-}Sky_{\mathcal{O}}$. If an object o_j in $b\text{-}Sky_{\mathcal{O}}$ is b-dominated by o_i, then o_j is removed from the set of candidate objects since it is not part of the evidential skyline.
At the end of the comparison of o_i with objects of $b\text{-}Sky_{\mathcal{O}}$, if o_i is not b-dominated by any object then, it is added to $b\text{-}Sky_{\mathcal{O}}$ as a candidate object. After the first phase, $b\text{-}Sky_{\mathcal{O}}$ comprises a set of objects that may be part of the b-dominated skyline.

To avoid the situation illustrated by the example in Table 5.1, a second phase is needed (lines 14–17). To determine if an object o_i in $b\text{-}Sky_{\mathcal{O}}$ is indeed in the b-dominant skyline, it is sufficient to compare o_i with those in $\mathcal{O} \setminus (b\text{-}Sky_{\mathcal{O}} \cup undom(o_i) \cup \{o_i\}$ that occurs earlier than o_i since the other ones have been already compared against o_i, where $undom(o_i)$ is the set of objects that occurs before o_i and that do not b-dominate o_j. This set is computed in the first phase in order to reduce the dominance checks in the second phase.

Applying TPA to our example (see Table 4.3), with $b = 0.4$:

- Both objects o_1 and o_4 will be inserted into $b\text{-}Sky_{\mathcal{O}}$ in the first phase,
- Then, o_1 and o_4 are not 0.4-dominated,
- Thus, after the second phase o_1 and o_4 will be returned.

However, applying TPA to the objects in Table 5.1, with $b = 0.3$, both o_x and o_z will be inserted in $b\text{-}Sky_{\mathcal{O}}$ in the first phase, but o_z will be eliminated by o_y in the second phase. Thus the algorithm returns only o_x.

Even if, TPA minimizes the number of dominance checks. It also may result in a high computational cost. In particular, when the average number of focal elements per object

Algorithm 3 Two phase algorithm TPA

Input: Objects $\mathcal{O}$; belief threshold b;
Output: Evidential skyline $b\text{-}Sky_{\mathcal{O}}$;

```
begin
    foreach o_i in O do
        isSkyline ← true  foreach o_j in b-Sky_O do
            if isSkyline then
                if o_j ≻_b o_i then
                    isSkyline ← false
                else
                    undom(o_i) ← undom(o_i) ∪ {o_j}
            if o_i ≻_b o_j then
                remove o_j from b-Sky_O
        if isSkyline then
            insert o_i in b-Sky_O
    foreach o_i in b-Sky_O do
        foreach o_j in O \ (b-Sky_O ∪ undom(o_i) ∪ {o_i}), pos(o_j) < pos(o_i) do
            if o_j ≻_b o_i then
                remove o_i from b-Sky_O
    return b-Sky_O;
```

is large. Thus, it is crucial to optimize the dominance checks to improve the performance of TPA.
In the following, we devise an efficient method that overcomes this problem using the minimum and the maximum values of each object w.r.t. each attribute. Given an object o_i, we denote by $o_i.a_k^-$ and by $o_i.a_k^+$ respectively the minimum value and the maximum value of o_i on attribute a_k.
For example, if we return to table (4.1), we have:

- $o_1.wl = \langle\{15, 16, 18\}, 0.1\rangle$, $\langle\{19, 20\}, 0.9\rangle$
- then, $o_1.wl^- = 15$ and $o_1.wl^+ = 20$.

Next, we delve into some useful properties that help us to improve the dominance checks. Given two objects $o_i, o_j \in \mathcal{O}$, and an attribute $a_k \in \mathcal{A}$ it is easy to check the following properties.

property 5.1. *if* $o_i.a_k^+ < o_j.a_k^-$ *then* $bel(o_i \succ o_j) = 0$ *since* $bel(o_i.a_k \geq o_j.a_k) = 0$.

property 5.2. *if* $o_i.a_k^- > o_j.a_k^+$ *then* $bel(o_i \succ o_j) = \prod_{a_\ell \in \mathcal{A} \setminus \{a_k\}} bel(o_i.a_\ell \geq o_j.a_\ell)$ *since* $bel(o_i.a_k \geq o_j.a_k) = 1$.

Algorithm 4 b-dominates(o_i, o_j, b)

foreach a_k *in* $\mathcal{A}$ **do**
  if $o_i.a_k^+ < o_j.a_k^-$ **then**
    return false

$bel \leftarrow 1$ **foreach** a_k *in* $\mathcal{A}$ **do**
  if $o_i.a_k^- \leq o_j.a_k+$ **then**
    $bel \leftarrow bel \times bel(o_i.a_k \geq o_j.a_k)$ **if** $bel < b$ **then**
      return false

return true

To determine if an object o_i b-dominates another object o_j Property 5.1 and Property 5.2 shows that it is not necessary to iterate the focal elements of all attributes. Based on these properties, we propose an efficient method (Algorithm 4) that checks if a given object o_i b-dominates or not another object o_j.

The details of b-dominates function are as follows.
For each attribute $a_k \in \mathcal{A}$, $o_i.a_k^+$ is compared against $o_j.a_k^-$. If there is any attribute a_k for which $o_i.a_k^+ < o_j.a_k^-$ holds then return false (loop in line 1); since o_i cannot b-dominate o_j according to Property 5.1.
Otherwise, the belief that o_i dominates o_j, bel, is computed by considering only the attributes where $o_i.a_k^- \leq o_j.a_k+$ (loop in line 5); since if $o_i.a_k^- > o_j.a_k^+$ then $bel(o_i.a_k \geq o_j.a_k) = 1$ and thus it does not affect the result; see Property 5.2.
The method returns "false" as soon as bel is less than the threshold b; since $bel \times bel(o_i.a_k \geq o_j.a_k)$ (line 7) will be less than b for the rest of iterations.
Finally, if bel is greater than b, the function returns true.

For example, comparing object o_1 against object $o2$ in our example. The method can quickly identify that $bel(o_1 \succ o_2) = 1$ and returns true, without iterating the focal element of each attribute. Similarly, to compare o_2 against o_1, it directly returns false.

5.3.2 Performance and scalability

In this section, we present an extensive experimental evaluation of our approach. More specifically, we focus on two issues:
(i) the size of the evidential skyline; and
(ii) the scalability of our proposed techniques for computing the evidential skyline.
For comparison purposes, we also implemented a baseline algorithm referred to as BA (baseline algorithm). In addition, to show the benefits resulting from the use of the dominance check function, we also implemented TPA without this function; we refer to this algorithm as BTPA (basic TPA).

5.3.3 Experimental Setup

Due to the limited availability of real evidential databases, existing approaches use synthetically generated datasets for their evaluation; we also follow this direction.
The generation of the sets of evidential data is controlled by the parameters in Table 5.2, which lists the parameters under investigation, their examined and default values. In each experimental setup, we investigate the effect of one parameter, while we set the remaining ones to their default values.

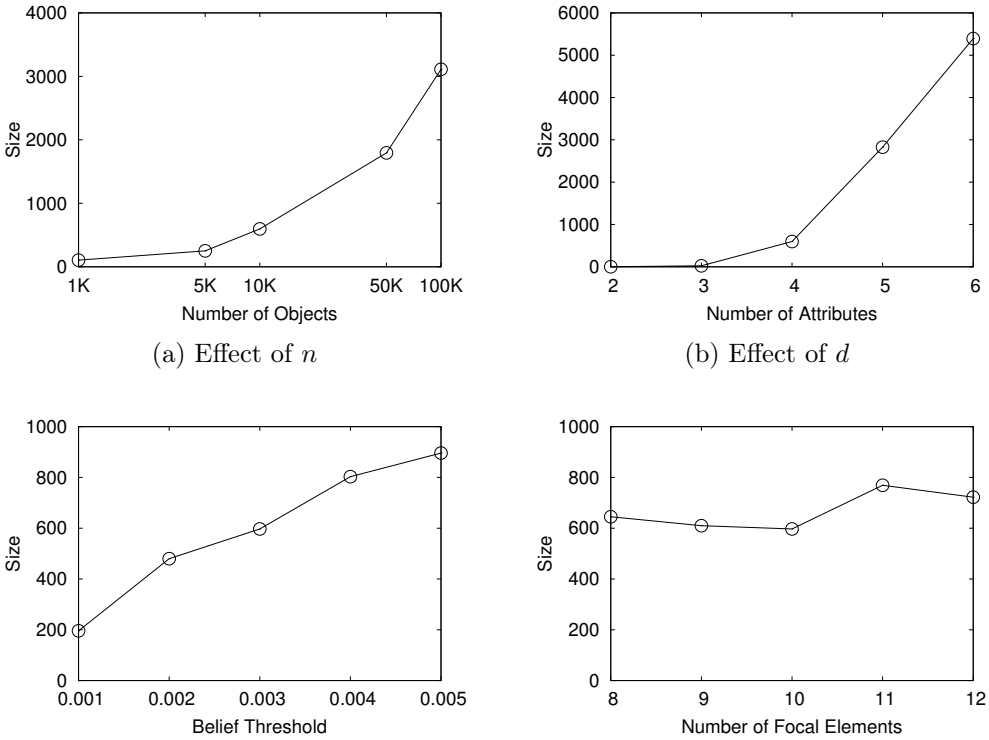
The data generator and the algorithms, i.e., BA, BTPA and TPA were, implemented in Java, and all experiments were conducted on a 2.3 GHz Intel Core i5 processor, with 8GB of RAM, running Mac OS X.

Parameter	Symbol	Values	Default
Number of objects	n	1K, 5K, 10K, 50K, 100K	10K
Number of attributes	d	2, 3, 4, 5, 6	4
belief threshold	b	0.001, 0.002, 0.003, 0.004, 0.005	0.003
Number of focal elements/attribute	f	8, 9, 10, 11, 12	10

Table 5.2: Parameters and Examined Values.

5.3.4 Size of the Evidential Skyline

Fig. 5.4 shows the size (i.e., the number of objects returned) of the b-dominant skyline w.r.t. n, d, b and f.
Fig. 5.4a shows that the size on the evidential skyline increases with higher n since when n increases more objects have chances not to be dominated.
As shown in Fig. 5.4b the cardinality of the evidential skyline increases significantly with the increase of d. This is because with the increase of d an object has better opportunity to be not dominated in all attributes.
Fig. 5.4c shows that the size of the evidential skyline increases with the increase of the b since the b-dominant skyline contains the b'-dominant skyline if $b > b'$; see Theorem 4.1 – recall that from this property users have the flexibility to control the size of the returned objects.
In contrast to n, d, and b, f has no apparent effect on the size of the evidential skyline as shown in Fig. 5.4d varying f some objects have better chances not to be dominated, while other have better chances to be dominated.



(a) Effect of n (b) Effect of d

(c) Effect of b (d) Effect of f

Figure 5.3: Size on the evidential skyline.

5.3.5 Performance and Scalability

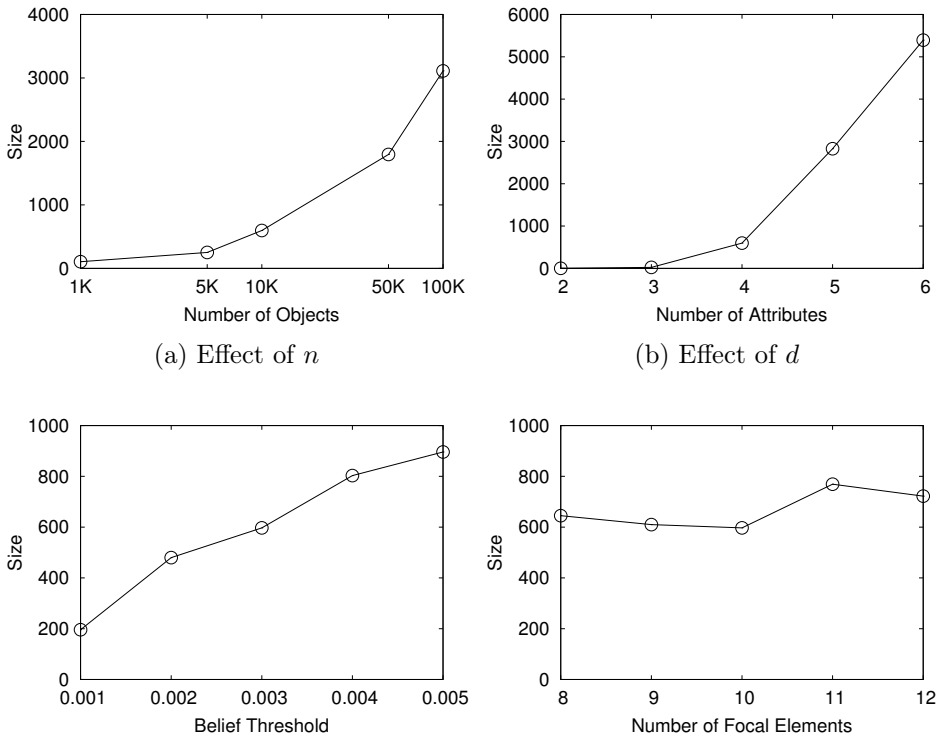
Fig. 5.5 depicts the execution time of the implemented algorithms w.r.t. n, d, b and f. Overall, TPA performs BA and BTPA. More specifically, TPA is faster than BTPA, which in turn is faster than BA.

As expected, Fig. 5.5a shows that the performance of the algorithms deteriorates with the increase of n. Observe that TPA is one order of magnitude faster BA and BTPA since it can quickly identifies if an object is dominated or not.

As shown in Fig. 5.5b BTPA does not scale with d. This is because when d increases the size of the evidential skyline becomes larger, thus a large number of objects will be selected to the second phase. Hence, BTPA performs a large number of dominance checks with a basic function. Even if TPA performs the same number of dominance checks than BTPA, TPA is efficient than BTPA since it can detect immediately whether an object dominates or not another.

As shown in Fig. 5.5c, BA is not affected by b as it computes the belief dominance between all objects. However, TPA and BTPA increases slightly because the size of the evidential skyline increases with the increase of b, thus, less objects will be eliminated.

Fig. 5.5d shows that the execution time of the algorithm slightly with the increase of f as the size of the skyline is not very affected by f, each algorithm performs practically the



(a) Effect of n (b) Effect of d

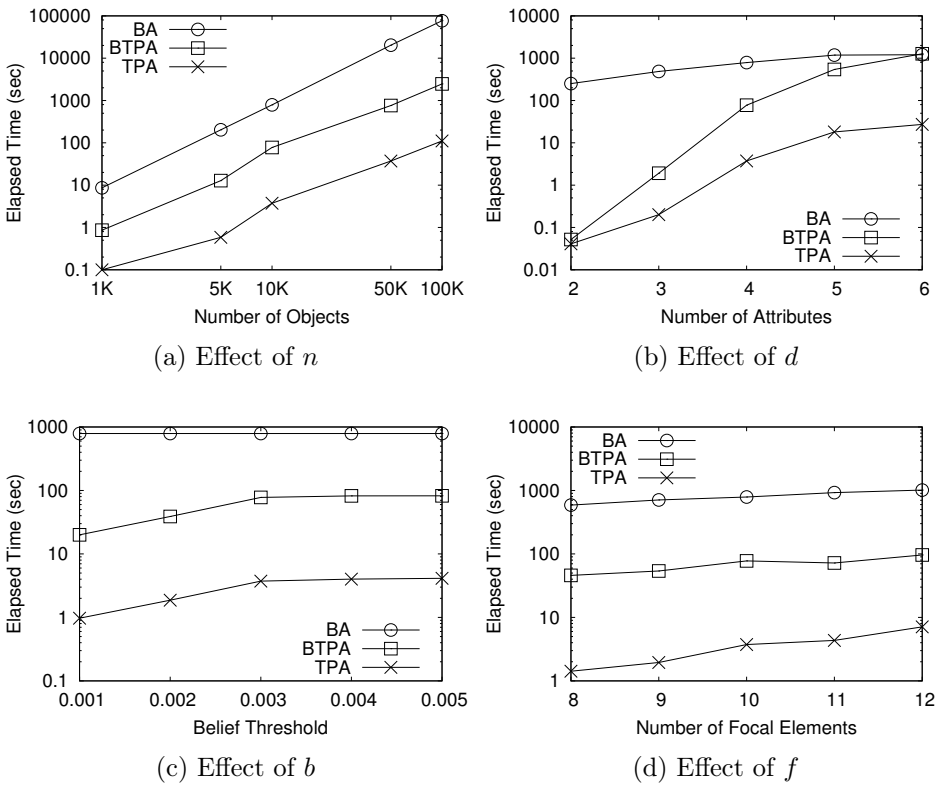
(c) Effect of b (d) Effect of f

Figure 5.4: Size on the evidential skyline.

same number of dominance check.
Still, observe that, TPA is more than one order of magnitude faster than BA and BTPA.

5.4 Conclusion

In this section, we presented a general algorithm for computing the degree of belief that a tuple from an evidential database, be d-dominated by another tuple where b is a given threshold introduced by the user. We then described the algorithms that allow to compare evidential attribute values (BBA) in two ways, naively and then efficiently with a performed method. We also developed some critical implementation details.



(a) Effect of n

(b) Effect of d

(c) Effect of b

(d) Effect of f

Figure 5.5: Elapsed time to compute the evidential skyline.

Chapter 6

Conclusion

This research project has presented a novel approach to modelling a new kind of skyline operator that enables the querying of evidential database.

This work differs from existing approaches in that it automatically translates the skyline modelling problem into uncertain data modeled by the evidence theory with the introducing of a given threshold. We can enumerate several benefits to this method that really open many perspectives as future work.

The use of a such kind of skyline queries that convert the classic preferences problem into an uncertain context allows criteria to be formalized in uncertain database and specifically evidential ones. More importantly, it also enables query answers to be effectively more real because the real world is full of uncertainty especially, that information in the web is collected from many and different sources.

The extension of the evidential database with skyline operator also permits the incorporation of the imperfection and the uncertainty in the decision making problems. In this framework, we developed an efficient algorithm for computing this skyline and Our experimental evaluation demonstrated the flexibility of the proposed evidential skyline and the scalability of our algorithm.

As future work, there are some experimental and theoretical issues that need to be addressed, however. On the experimental side, although our algorithm gives good results, we have to improve so that the execution time decreases for processing large amounts of data.

On the theoretical side, an interesting future direction is to implement the second model proposed above, that uses the possible worlds, and to develop new techniques for ranking the objects retrieved by the evidential skyline.

Bibliography

[1] Charu C. Aggarwal and Philip S. Yu. A survey of uncertain data algorithms and applications. *IEEE Trans. Knowl. Data Eng.*, 21(5):609–623, 2009.

[2] David A. Bell, J. W. Guan, and Suk Kyoon Lee. Generalized union and project operations for pooling uncertain and imprecise information. *Data Knowl. Eng.*, 18(2):89–117, 1996.

[3] Karim Benouaret, Djamal Benslimane, and Allel Hadjali. A fuzzy framework for selecting top-k web service compositions. *SIGAPP Appl. Comput. Rev.*, 11(3):32–40, August 2011.

[4] Karim Benouaret, Djamal Benslimane, and Allel HadjAli. Selecting skyline web services from uncertain qos. In *IEEE SCC*, pages 523–530, 2012.

[5] Karim Benouaret, Djamal Benslimane, Allel Hadjali, Mahmoud Barhamgi, Zakaria Maamar, and Quan Z. sheng. Web Service Compositions with Fuzzy Preferences: A Graded Dominance Relationship Based Approach. *ACM Transactions on Internet Technology*, June 2014.

[6] Patrick Bosc, Allel HadjAli, and Olivier Pivert. On possibilistic skyline queries. In *FQAS*, pages 412–423, 2011.

[7] Patrick Bosc and Olivier Pivert. Modeling and querying uncertain relational databases: a survey of approaches based on the possible worlds semantics. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 18(5):565–603, 2010.

[8] Stephan Börzsönyi, Donald Kossmann, and Konrad Stocker. The skyline operator. pages 421–430, 2001.

[9] Chee Yong Chan, H. V. Jagadish, Kian-Lee Tan, Anthony K. H. Tung, and Zhenjie Zhang. Finding k-dominant skylines in high dimensional space. In *SIGMOD Conference*, pages 503–514, 2006.

[10] Jan Chomicki, Parke Godfrey, Jarek Gryz, and Dongming Liang. Skyline with presorting. In *ICDE*, pages 717–719, 2003.

[11] A. P. Dempster. A generalization of bayesian inference. *Journal of the Royal Statistical Society*, 30(B):205–247, 1968.

[12] Gabriel Pui Cheong Fung, Wei Lu, and Xiaoyong Du. Dominant and k nearest probabilistic skylines. *Lecture Notes in Computer Science*, 5463:263–277, 2009.

[13] Minh Ha-Duong. Hierarchical fusion of expert opinions in the transferable belief model, application to climate sensitivity. *Int. J. Approx. Reasoning*, 49(3):555–574, 2008.

[14] Allel Hadjali, Souhila Kaci, and Henri Prade. Database preferences queries – a possibilistic logic approach with symbolic priorities. *Foundations of Information and Knowledge Systems*, 4932:291–310, 2008.

[15] Jun Hong, Zhongtian He, and David A. Bell. An evidential approach to query interface matching on the deep web. *Information Systems archive*, 35(2):140–148, 2010.

[16] Bin Jiang, Jian Pei, Xuemin Lin, and Yidong Yuan. Probabilistic skylines on uncertain data: model and bounding-pruning-refining methods. *Journal of Intelligent Information Systems*, 38(1):1–39, 2012.

[17] Donald Kossmann, Frank Ramsak, and Steffen Rost. Shooting stars in the sky: an online algorithm for skyline queries. pages 275 – 286, 2002.

[18] Suk Kyoon Lee. An extended relational database model for uncertain and imprecise information. In *VLDB*, pages 211–220, 1992.

[19] Xuemin Lin, Ying Zhang, Wenjie Zhang, and Cheema M.A. Stochastic skyline operator. *Data Engineering (ICDE), IEEE*, pages 721 – 732, 2011.

[20] Dimitris Papadias, Greg Fu, Jp Morgan Chase, and Bernhard Seeger. Progressive skyline computation in database systems. *ACM Trans. Database Syst*, 30:2005, 2005.

[21] Jian Pei, Bin Jiang, Xuemin Lin, and Yidong Yuan. Probabilistic skylines on uncertain data. In *VLDB*, pages 15–26, 2007.

[22] Ahmed Samet, Eric Lefèvre, and Sadok Ben Yahia. Mining frequent itemsets in evidential database. *Advances in Intelligent Systems and Computing*, 245:377–388, 2014.

[23] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, 1976.

[24] Mehdi Sharifzadeh and Cyrus Shahabi. The spatial skyline queries. In *VLDB Proceedings of the 32nd international conference on Very large data*, pages 751–762, 2006.

[25] Kian-Lee Tan, Pin-Kwang Eng, and Beng Chin Ooi. Efficient progressive skyline computation. pages 301–310, 2001.

[26] Mohamed Anis Bach Tobji, Boutheina Ben Yaghlane, and Khaled Mellouli. A new algorithm for mining frequent itemsets from evidential databases. In *IPMU*, pages 1535–1542, 2008.

[27] Hyountaek Yong, Jongwuk Lee, Jinha Kim, and Seung won Hwang. Skyline ranking for uncertain databases. *Information Systems*, 2014.

[28] Wenjie Zhang, Xuemin Lin, Ying Zhang, Wei Wang, and Jeffrey Xu Yu. Probabilistic skyline operator over sliding windows. In *ICDE*, pages 1060–1071, 2009.